SIEMENS

SINUMERIK

SINUMERIK 840D sl / 828D
Munka-előkészítés

Programozási kézikönyv

Érvényes

vezérlés
SINUMERIK 840D sl / 840DE sl / 828D

CNC szoftver 4.8 SP3 verzió

08/2018
6FC5398-2BP40-6QA2

Előszó
Alapvető biztonsági utalások 1
Rugalmas NC programozás 2
Fájl- és program-kezelés 3
Védelmi tartományok: 4
Speciális út-utasítások 5
Koordináta-transzformációk (frame-k) 6
Transzformációk 7
Kinematikai láncok 8
Ütközés elkerülése kinematikai láncokkal 9
Transzformációk kinematikai láncokkal 10
Szerszámkorrekciók 11
Pályaviselkedés 12
Tengely-csatolások 13
Szinkron-akciók 14
Ingázás 15
Lyukasztás és sapkázás 16
Köszörülés 17
További funkciók 18
Saját leforgácsoló-program 19
Ciklusok külső programozása 20
Táblázatok 21
Függelék A

Jogi megjegyzések

Figyelmeztetési utasítás tervezet

A kézikönyv útmutatásokat tartalmaz, amelyeket személyes biztonsága, valamint az anyagi károk megelőzése érdekében követnie kell. A személyes biztonságához kapcsolódó útmutatásokat veszélyjelző háromszög emeli ki, Az általános anyagi károkhoz kapcsolódó útmutatásoknál nincs veszélyjelző háromszög. A veszély súlyossági fokától függően a veszélyjelző útmutatásokat a súlyostól a kevésbé súlyos veszély felé haladva a következőképpen ábrázolják.

⚠ VESZÉLY
Azt jelenti, hogy halálos baleset vagy súlyos sérülés**történik**, ha nem hozzák meg a megfelelő elővigyázatossági rendszabályokat.

⚠ FIGYELMEZTETÉS
Azt jelenti, hogy halálos baleset vagy súlyos sérülés**történhet**, ha nem hozzák meg a megfelelő elővigyázatossági rendszabályokat.

⚠ VIGYÁZAT
Azt jelenti, hogy könnyű sérülés történhet, ha nem hozzák meg a megfelelő elővigyázatossági rendszabályokat.

FIGYELEM
Azt jelenti, hogy anyagi kár történhet, ha nem hozzák meg a megfelelő elővigyázatossági rendszabályokat.

Ha a különböző súlyossági fokú veszélyből egyszerre több áll fenn, mindig a legsúlyosabb fokú veszélyhez tartozó veszélyjelző háromszöget használják. Ha veszélyjelző háromszöggel ellátott veszélyjelző útmutatás személyi sérülések lehetőségére figyelmeztet, az útmutatáshoz anyagi károk veszélyét jelző útmutatás is társítható.

Szakképzett személyzet

Az ehhez a dokumentációhoz tartozó terméket/rendszert csak az adott feladatkörre **kiképzett személyzet** kezelheti az adott feladatkörre vonatkozó dokumentáció figyelembevételével, különös tekintettel az abban foglalt biztonsági és figyelmeztető utasításokra. A kiképzett személyzet a kiképzésére és tapasztalatára alapozva képes az ezekkel a termékekkel/rendszerekkel történő munkák során a kockázatok felismerésére és a lehetséges veszélyek elkerülésére.

Siemens termékek rendeltetésszerű használata

Ennél a következőket kell követni:

⚠ FIGYELMEZTETÉS
A Siemens termékek csak a katalógusban és a hozzátartozó műszaki dokumentációban meghatározott alkalmazási esetekre használhatók. Ha idegen termékek és –egységek alkalmazására kerül sor, akkor be kell szerezni a Siemens javaslatát ill. engedélyét. A termékek kifogástalan és biztonságos üzemeltetésének előfeltétele a szakszerű szállítás, szakszerű tárolás, felállítás, összeszerelés, telepítés, üzembe helyezés, kezelés és karbantartás. A megengedett környezeti feltételeket be kell tartani. A hozzátartozó dokumentációkban szereplő utasításokat figyelembe kell venni.

Védjegyek

Az ® oltalmi jogi megjegyzéssel jelölt minden elnevezés a Siemens AG. bejegyzett védjegye. A dokumentációban használt többi elnevezés olyan védjegy lehet, amelyeknek harmadik fél részéről saját célra történő használata sértheti a tulajdonosaik jogait.

Felelősség kizárása

Megvizsgáltuk, hogy a nyomtatvány tartalma egyezik-e az ismertetett hardverrel és szoftverrel. Ennek ellenére nem zárható ki, hogy eltérések vannak közöttük, ezért a maradéktalan egyezésért nem vállalunk felelősséget. A nyomtatvány tartalmát rendszeresen átnézzük, a szükséges javításokat a soron következő kiadásokban szerepeltetjük.

Siemens AG
Division Digital Factory
Postfach 48 48
90026 NÜRNBERG
NÉMETORSZÁG

Dokumentum rendelési szám: 6FC5398-2BP40-6QA2
℗ 08/2018 A változtatások jogát fenntartjuk

Copyright © Siemens AG 1995 - 2018.
Minden jog fenntartva

Előszó

SINUMERIK dokumentáció

A SINUMERIK dokumentáció a következő kategóriába van tagolva:

- Általános dokumentáció/katalógusok
- Felhasználói dokumentáció
- Gyártói-/szerviz-dokumentáció

További információk

A következő címen (https://support.industry.siemens.com/cs/de/en/view/108464614) találhat információt a témákról:

- Dokumentáció megrendelése / Dokumentációk áttekintése
- További helyek a dokumentációk letöltéséhez
- Dokumentációk online használata (kézikönyveket/információkat megtalálni és bennük keresni)

A műszaki dokumentációval kapcsolatos kérdésekkel (pl. javaslatok, javítások) küldjenek egy e-mailt a következő címre (mailto:docu.motioncontrol@siemens.com):

mySupport/Dokumentation

A következő címen (https://support.industry.siemens.com/My/ww/en/documentation) információkat találhat arról, hogyan állíthatja össze egyénileg dokumentációját a Siemens-tartalmak alapján, és hogyan igazíthatja ezt hozzá a saját gépdokumentációjához.

Képzés

A következő címen (http://www.siemens.com/sitrain) talál információt a Siemens SITRAIN tréningjéről – a meghajtás és automatizálási technika termékeivel, rendszereivel és megoldásaival kapcsolatban.

FAQ

Frequently Asked Questions (Gyakran Ismételt Kérdések) a Service&Support oldalakon a Produkt Support (https://support.industry.siemens.com/cs/de/en/ps/faq)-nál találhatók.

SINUMERIK

A következő címen (http://www.siemens.com/sinumerik) találhat információt a SINUMERIK-ről:

Célcsoport

Ez a kiadvány a következőkhöz szól:

- programozók
- tervezők

Haszon

Ez a programozási kézikönyv képessé teszi a célcsoportot programok és szoftver-felületek tervezésére, írására, tesztelésére és a hibák megszüntetésére.

Alap terjedelem

Jelen programozási kézikönyvben az alap terjedelem funkcionalitása kerül leírásra. A gépgyártó által végzett kiegészítéseket és változtatásokat a gépgyártó dokumentálja.

A vezérlésben működhetnek további, ebben a dokumentációban nem leírt funkciók is. Ezekre a funkciókra azonban nem lehet igényt támasztani egy új szállításnál ill. szerviz esetén.

Ez a dokumentáció az áttekinthetőség miatt nem tartalmazza a termék összes típusának valamennyi részletes információját és nem veheti figyelembe az alkalmazás, az üzemeltetés és a karbantartás valamennyi elképzelhető esetét.

Műszaki támogatás

Országonkénti telefonszámok találhatók a műszaki támogatáshoz az interneten a következő címen (https://support.industry.siemens.com/sc/ww/en/sc/2090) a „Kapcsolatok” alatt.

Szerkezet és tartalom információk

Programozási kézikönyv Alapok / Munka-előkészítés

Az NC programozás leírása két kézikönyvre van felosztva:

1. **Alapok**
   Az "Alapok" programozási kézikönyv a gépkezelő szakmunkások részére készült és feltételezi a fúrási, marási és esztergálási megmunkálások megfelelő ismereteket . Egyszerű programozási példákon elmagyarázásra kerülnek a DIN 66025 szerint ismeretes utasítások és parancsok is.
2. **Munka-előkészítés**
   A "Munka-előkészítés" Programozási kézikönyv technológusok részére ismereteket ad a teljes programozási lehetőségekről. A SINUMERIK vezérlés egy speciális programozói nyelv segítségével lehetővé teszi egy komplex munkadarabprogram programozását (pl. szabad formafelületek, csatorna-koordinálás,...) és megkönnyíti a technológusok részére a bonyolultabb programozást.

A leírt NC nyelvi elemek rendelkezésre állása

Az ebben a kézikönyvben leírt NC nyelvi elelemek a SINUMERIK 840D sl esetén állnak rendelkezésre. A rendelkezésre állás a SINUMERIK 828D esetén a "Utasítások: Rendelkezésre állás SINUMERIK 828D esetén (Oldal 859)" táblázatban található.

Tartalomjegyzék

Előszó....3

1 Alapvető biztonsági utalások....17

1.1 Általános biztonsági utalások....17

1.2 Garancia és szavatosság az alkalmazási példákra....18

1.3 Industrial Security....19

2 Rugalmas NC programozás....21

2.1 Változók....21
2.1.1 Rendszeradatok....21
2.1.2 Előre definiált felhasználói változók: Számítási paraméterek....24
2.1.2.1 Csatorna-specifikus számítási paraméterek (R)....24
2.1.2.2 Globális számítási paraméterek (RG)....25
2.1.3 Előre definiált felhasználói változók: Link-változók....27
2.1.4 Felhasználói változók definíciója (DEF)....29
2.1.5 Rendszeradatok, felhasználói adatok és NC utasítások újra definiálása (FRDEF)....35
2.1.6 Tulajdonság: Inicializálási érték....38
2.1.7 Tulajdonság: Határértékek (LLI, ULI)....41
2.1.8 Tulajdonság: Fizikai egység (PHU)....43
2.1.9 Tulajdonság: Hozzáférési jog (APR, APW, APRP, APWP, APRB, APWB)....45
2.1.10 Tulajdonság: Adat-osztály (DCM, DCI, DCU) - csak SINUMERIK 828D....50
2.1.11 Definiálható és újra definiálható tulajdonságok áttekintése....51
2.1.12 Mező-változók (DEF, SET, REP) definíciója és inicializálása....52
2.1.13 Mező-változók (DEF, SET, REP) definíciója és inicializálása: További információk....56
2.1.14 Adattípusok....57
2.1.15 Egy változó előfordulását megvizsgálni (ISVAR)....58
2.1.16 Attribútum érték / adattípus olvasása (GETVARPHU, GETVARAP, GETVARLIM, GETVARDIM, GETVARDFT, GETVARTYP)....60

2.2 Közvetett programozás....66
2.2.1 Címek közvetett programozása....66
2.2.2 G utasítások közvetett programozása....68
2.2.3 Pozíció tulajdonságok (GP) közvetett programozása....69
2.2.4 Munkadarabprogram sorok közvetett programozása (EXECSTRING)....72

2.3 Számítási műveletek....73

2.4 Összehasonlító és logikai műveletek....76

2.5 Pontosság-korrekció összehasonlítási hibáknál (TRUNC)....78

2.6 Változók minimuma, maximuma és tartománya (MINVAL, MAXVAL, BOUND)....80

2.7 Műveletek prioritása....82

2.8 Lehetséges típus-konverziók....83

2.9 String műveletek....84
2.9.1 Típus-konverzió STRING-re (AXSTRING)....84

2.9.2 Típus-konverzió STRING-ből (NUMBER, ISNUMBER, AXNAME) 85
2.9.3 Stringek láncolása (<<) 86
2.9.4 Kis- /nagybetű átalakítás (TOLOWER, TOUPPER) 87
2.9.5 String hosszának meghatározása (STRLEN) 88
2.9.6 Karakter/string keresése stringben (INDEX, RINDEX, MINDEX, MATCH) 89
2.9.7 Rész-string kiválasztása (SUBSTR) 90
2.9.8 Egyes karakterek írása és olvasása....91
2.9.9 String formattálás (SPRINT)....92

2.10 Program ugrások és elágazások....101
2.10.1 Visszaugrás a programkezdetre (GOTOS)....101
2.10.2 Programugrások ugrás-jelölőkre (GOTOB, GOTOF, GOTO, GOTOC)....102
2.10.3 Program-elágazás (CASE ... OF ... DEFAULT ...)....105

2.11 Programrész ismétlése (REPEAT, REPEATB, ENDLABEL, P)....107

2.12 Vezérlő struktúrák....113
2.12.1 Feltételes utasítás és elágazás (IF, ELSE, ENDIF)....114
2.12.2 Végtelen programhurok (LOOP, ENDLOOP)....116
2.12.3 Számláló hurok (FOR ... TO ..., ENDFOR)....116
2.12.4 Programhurok feltétellel a hurok elején (WHILE, ENDWHILE)....118
2.12.5 Programhurok feltétellel a hurok végén (REPEAT, UNTIL)....118
2.12.6 Programpélda skatulyázott vezérlő-struktúrákkal 119

2.13 Koordinációs utasítások (INIT, START, WAITM, WAITMC, WAITE, SETM, CLEARM). 120

2.14 Interrupt rutinok (ASUP)....125
2.14.1 Egy interrupt-rutin funkciója....125
2.14.2 Interrupt-rutin létrehozása....126
2.14.3 Interrupt-rutint hozzárendelni és indítani (SETINT, PRIO, BLSYNC)....126
2.14.4 Egy interrupt-rutin hozzárendelését deaktiválni/újra-aktiválni (DISABLE, ENABLE)....129
2.14.5 Egy interrupt-rutin hozzárendelésének törlése (CLRINT)....129
2.14.6 Gyors leemelés a kontúrról (SETINT LIFTFAST, ALF)....130
2.14.7 Mozgásirány gyors leemelésnél a kontúrról 132
2.14.8 Mozgás-lefutások interrupt-rutinoknál....135

2.15 Tengely-csere, orsó-csere (RELEASE, GET, GETD) 137

2.16 Tengelyt egy másik csatornának átadni (AXTOCHAN)....142

2.17 Gépadatokat hatásossá tenni (NEWCONF)....144

2.18 Fájl írás (WRITE)....145

2.19 Fájl törlés (DELETE)....149

2.20 Sorok olvasása fájlban (READ)....150

2.21 Egy fájl létezését megvizsgálni (ISFILE)....152

2.22 Fájl információkat kiolvasni (FILEDATE, FILETIME, FILESIZE, FILESTAT, FILEINFO)....153

2.23 Felkerekítés (ROUNDUP)....155

2.24 Alprogram-technika....156
2.24.1 Általános....156
2.24.1.1 Alprogram....156
2.24.1.2 Alprogramnevek....157
2.24.1.3 Alprogramok egymásba skatulyázása....158
2.24.1.4 Keresési ág....159

2.24.1.5 Formális és aktuális paraméterek....159
2.24.1.6 Paraméter-átadás....160
2.24.2 Egy alprogram definíciója....162
2.24.2.1 Alprogram paraméter-átadás nélkül....162
2.24.2.2 Alprogram Call-by-Value paraméter-átadással (PROC)....162
2.24.2.3 Alprogram Call-by-Reference paraméter-átadással (PROC, VAR)....164
2.24.2.4 Modális G-funkciókat menteni (SAVE)....166
2.24.2.5 Egyes-mondat feldolgozás elnyomás (SBLOF, SBLON)....167
2.24.2.6 Aktuális mondatkijelzés elnyomása (DISPLOF, DISPLON, ACTBLOCNO)....173
2.24.2.7 Alprogramokat előkészítéssel megjelölni (PREPRO)....176
2.24.2.8 Alprogram-visszaugrás M17....176
2.24.2.9 Alprogram-visszaugrás RET....177
2.24.2.10 Paraméterezhető alprogram-visszaugrás (RET...)....178
2.24.2.11 Paraméterezhető alprogram-visszaugrás (RETB ...)....184
2.24.3 Egy alprogram felhívása....188
2.24.3.1 Alprogram hívása paraméter-átadás nélkül....188
2.24.3.2 Alprogram felhívása paraméter-átadással (EXTERN)....190
2.24.3.3 Program ismétlések száma (P)....192
2.24.3.4 Modális alprogram-hívás (MCALL)....193
2.24.3.5 Közvetett alprogram-hívás (CALL)....195
2.24.3.6 Közvetett alprogram-hívás a végrehajtandó programrész megadásával (CALL BLOCK ... TO ...)....196
2.24.3.7 Egy ISO-nyelven programozott program közvetett felhívása (ISOCALL)....197
2.24.3.8 Alprogramot ág-megadással és paraméterekkel felhívni (PCALL)....198
2.24.3.9 Keresőágat alprogram-hívásnál bővíteni (CALLPATH)....199
2.24.3.10 Külső alprogram feldolgozás (840D sl) (EXTCALL)....200
2.24.3.11 Külső alprogramot feldolgozni (828D) (EXTCALL)....203

2.25 Makró-technika (DEFINE...AS)....208

3 Fájl- és program-kezelés....211

3.1 Programtároló....211
3.1.1 Programtároló az NC-ben....211
3.1.2 Külső programtároló....213
3.1.3 Programtároló fájljainak címzése....215
3.1.4 Kereső-ág alprogramhívásnál....219
3.1.5 Az ág és a fájlnév lekérdezése....220

3.2 Munkatároló (CHANDATA, COMPLETE, INITIAL)....222

4 Védelmi tartományok:....225

4.1 Védelmi tartományok definiálni (CPROTDEF, NPROTDEF)....225

4.2 Védelmi tartományok aktiválása/deaktiválása (CPROT, NPROT)....229

4.3 Védelmi-tartomány sértés, munkatér-határolás és szoftver-végállások vizsgálata (CALCPOSI)....233

5 Speciális út-utasítások....243

5.1 Kódolt pozíciókra menni (CAC, CIC, CDC, CACP, CACN)....243

5.2 Spline interpoláció (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL)....244

5.3 Spline-kötelék (SPLINEPATH)....255

5.4 NC-mondat kompresszort be-/kikapcsolni (COMPON / COMPCURV, COMPCAD, COMPSURF, COMPOF)....257
5.5 Polinom-interpoláció (POLY, POLYPATH, PO, PL)....258
5.6 Beállítható pálya-vonatkoztatás: (SPATH, UPATH)....264
5.7 Mérés kapcsoló tapintóval (MEAS, MEAW)....266
5.8 Tengely-specifikus mérés (MEASA, MEAWA, MEAC) (opció)....269
5.9 Speciális funkciók OEM-felhasználóknak (OMA1 ... OMA5, OEMIPO1, OEMIPO2, G810 ... G829)....279
5.10 Előtolás-csökkentés sarokkésleltetéssel (FENDNORM, G62, G621)....280
5.11 Programozható mozgás vége kritériumok (FINEA, COARSEA, IPOENDA, IPOBRKA, ADISPOSA)....281

6 Koordináta-transzformációk (frame-k)....285

6.1 Koordináta-transzformáció frame változókkal....285
6.1.1 Előre definiált frame változók ($P_CHBFRAME, $P_IFRAME, $P_PFRAME, $P_ACTFRAME)....287
6.2 Érték-hozzárendelés frame-ekhez....291
6.2.1 Közvetlen érték-hozzárendelés (tengelyérték, szög, mérték)....291
6.2.2 Frame komponensek olvasása és megváltoztatása (TR, FI, RT, SC, MI)....293
6.2.3 Számolás frame-ekkel....294
6.2.4 Frame változók definíciója (DEF FRAME)....296
6.3 Durva- és finom-eltolás (CTRANS, CFINE)....297
6.4 Külső nullaponteltolás ($AA_ETRANS)....299
6.5 Valósérték beállítása a referencia állapot elvesztésével (PRESETON)....301
6.6 Valósérték beállítása a referencia állapot elvesztése nélkül (PRESETONS)....303
6.7 Frame-számítás 3 térbeli mérési pontból (MEAFRAME)....305
6.8 NCU-globális frame-k....309
6.8.1 Csatorna-specifikus frame-k ($P_CHBFR, $P_UBFR)....310
6.8.2 Csatornában hatásos frame-k....310

7 Transzformációk....315

7.1 A transzformáció fajták általános programozása....315
7.1.1 Tájolási mozgások a transzformációknál....317
7.1.2 TRAORI tájolási transzformáció áttekintése....321
7.2 Három-, négy-, öt-tengelyes transzformációk (TRAORI)....323
7.2.1 Általános összefüggések kardán szerszámfejnél....323
7.2.2 Három-, négy- és öt-tengelyes transzformáció (TRAORI)....326
7.2.3 Tájolás programozás és alaphelyzet változatok (OTIRESET)....327
7.2.4 Szerszámtájolás programozása (A..., B..., C..., LEAD, TILT)....329
7.2.5 Homlokmarás (A4, B4, C4, A5, B5, C5)....335
7.2.6 A tájolótengelyek vonatkoztatása (ORIWKS, ORIMKS)....336
7.2.7 Tájolótengelyek programozása (ORIAXES, ORIVECT, ORIEULER, ORIRPY, ORIRPY2, ORIVIRT1, ORIVIRT2)....338

7.2.8 Tájolás programozása egy kúppalást-felület mentén (ORIPLANE, ORICONCW, ORICONCCW, ORICONTO, ORICONIO)....340
7.2.9 Két érintőpont tájolás-megadása (ORICURVE, PO[XH]=, PO[YH]=, PO[ZH]=)....343

7.3 Tájolási polinomok (PO[szög], PO[koordináta])....345

7.4 Szerszámtájolás forgatásai (ORIROTA, ORIROTR, ORIROTT, ORIROTC, THETA)....347

7.5 Pálya-viszonyítású tájolás....350
7.5.1 Pályához viszonyított tájolási módok....350
7.5.2 Szerszámtájolás pályára vonatkoztatott forgatása (ORIPATH, ORIPATHS, forgásszög)....351
7.5.3 Szerszámforgatás pályára vonatkozatott interpolációja (ORIROTC, THETA)....352
7.5.4 A tájolás lefutásának simítása (ORIPATHS A8=, B8=, C8=)....354

7.6 A tájolás komprimálása (COMPON, COMPCURV, COMPCAD, COMPSURF)....356

7.7 Tájolás lefutás simítását be-/kikapcsolni (ORISON, ORISOF)....359

7.8 Kinematikus transzformációk....361
7.8.1 Homlokoldal-transzformációt bekapcsolni (TRANSMIT)....361
7.8.2 Hengerpalást-transzformációt bekapcsolni (TRACYL)....361
7.8.3 Ferdeszög-transzformációt programozható szöggel bekapcsolni (TRAANG)....364
7.8.4 Ferde beszúrás köszörűgépeken (G5, G7)....365

7.9 Láncolt transzformáció bekapcsolása (TRACON)....367

7.10 PTP-mozgás derékszögű koordinátarendszerben....369
7.10.1 Derékszögű PTP-mozgás be-/kikapcsolása (PTP, PTPG0, PTPWOC, CP)....369
7.10.2 Csukló állását megadni (STAT)....370
7.10.3 Tengelyszög előjelét megadni (TU)....374
7.10.4 Példa 1: Egy 6 tengelyes robot PTP mozgatása ROBX transzformációval....377
7.10.5 Példa 2: PTP-mozgás általános 5-tengelyes transzformációnál....378
7.10.6 Példa 3: PTPG0 és TRANSMIT....378

7.11 Peremfeltételek egy transzformáció felhívásánál....380

7.12 Transzformáció kikapcsolása (TRAFOOF)....381

8 Kinematikai láncok....383

8.1 Komponensek törlése (DELOBJ)....383

8.2 Index megállapítása névvel (NAMETOINT)....386

9 Ütközés elkerülése kinematikai láncokkal....387

9.1 Ütközés párok vizsgálata (COLLPAIR)....388

9.2 Ütközés elkerülés gép-modell újra számítását igényelni (PROTA)....389

9.3 Védelmi tartomány állapotot beállítani (PROTS)....390

9.4 Két védelmi tartomány távolságának meghatározása (PROTD)....391

10 Transzformációk kinematikai láncokkal....393

10.1 Transzformáció bekapcsolása (TRAFOON)....393

10.2 Tájolási transzformáció a gép bemérése után módosítani (CORRTRAFO)....395

11 Szerszámkorrekciók....403

11.1 Korrekció-tároló....403

11.2 Additív korrekciók....406
11.2.1 Additív korrekciókat kiválasztani (DL)....406
11.2.2 Kopási és beállítási érték megadása ($TC_SCPxy[t,d], $TC_ECPxy[t,d])....407
11.2.3 Additív korrekciókat törölni (DELDL)....408

11.3 Szerszámkorrekció - különleges kezelés....410
11.3.1 Szerszámhosszak tükrözése....411
11.3.2 Kopás előjel értékelés....412
11.3.3 Aktív megmunkálás koordinátarendszere (TOWSTD, TOWMCS, TOWWCS, TOWBCS, TOWTCS, TOWKCS)....413
11.3.4 Szerszámhossz és síkcsere....415

11.4 Online szerszámkorrekció....417
11.4.1 Polinom függvény definiálása (FCTDEF)....417
11.4.2 Online szerszámkorrekció írás, folytonosan (PUTFTOCF)....418
11.4.3 Online szerszámkorrekció írás, diszkrét (PUTFTOC)....419
11.4.4 Online-szerszámkorrekciót be-/kikapcsolni (FTOCON/FTOCOF)....420

11.5 3D-s szerszámkorrekció....421
11.5.1 3D-s szerszámsugár-korrekció aktiválása a 3D-s kerületmarásra (CUT3DC, CUT3DCD, ISD)....421
11.5.2 3D-s szerszámkorrekciót a 3D-s homlokmarásra kiválasztani (CUT3DF, CUT3DFS, CUT3DFF, CUT3DFD)....425
11.5.3 3D-s kerületi marás egy határoló felület figyelembe vételével (CUT3DCC, CUT3DCCD)....431

11.6 Szerszám-tájolás (ORIC, ORID, OSOF, OSC, OSS, OSSE, ORIS, OSD, OST)....436

11.7 Szabad D-szám megadás, vágóélszám....442
11.7.1 Szabad D-szám megadás, vágóél-szám (CE cím)....442
11.7.2 Szabad D-szám megadás: D-számot vizsgálni (CHKDNO)....442
11.7.3 Szabad D-szám megadás: D-számot átnevezni (GETDNO, SETDNO)....443
11.7.4 Szabad D-szám megadás: T-számot a megadott D-számhoz megállapítani (GETACTTD)....444
11.7.5 Szabad D-szám megadás: D-számot érvénytelenné tenni (DZERO)....444

11.8 Szerszámtartó kinematika....445

11.9 Szerszámhossz-korrekció tájolható szerszámtartókra (TCARR, TCOABS, TCOFR, TCOFRX, TCOFRY, TCOFRZ)....450

11.10 Online szerszámhossz-korrekció (TOFFON, TOFFOF)....453

11.11 Korrekcióadatok módosítása forgatható szerszámoknál....456
11.11.1 Tájolásokat kiszámítani (ORISOLH)....456
11.11.2 Korrekcióadatok módosítását forgatható szerszámoknál aktiválni (CUTMOD, CUTMODK)....464

11.12 Szerszám környezetekkel dolgozni....471
11.12.1 Szerszám-környezetet tárolni (TOOLENV)....471
11.12.2 Szerszám-környezetet törölni (DELTOOLENV)....474
11.12.3 T-, D-, és DL-számokat olvasni (GETTENV)....475
11.12.4 Információk olvasása a tárolt szerszám-környezethez ($P_TOOLENVN, ($P_TOOLENV)....476
11.12.5 Szerszámhosszakat ill. szerszámhossz-komponenseket kiolvasni (GETTCOR)....476
11.12.6 Szerszám-komponenseket változtatni (SETTCOR)....483

11.13 L1, L2, L3 szerszámhosszak hozzárendelése a koordináta-tengelyek olvasásához (LENTOAX)....496

12 Pályaviselkedés....499

12.1 Érintőleges vezérlés....499
12.1.1 Csatolást definiálni (TANG)....499
12.1.2 Közbenső mondat létrehozás bekapcsolása (TLIFT)....500
12.1.3 Csatolást bekapcsolni (TANGON)....501
12.1.4 Csatolást kikapcsolni (TANGOF)....503
12.1.5 Csatolást törölni (TANGDEL)....503

12.2 Előtolás lefutás (FNORM, FLIN, FCUB, FPO)....505

12.3 Gyorsulási viselkedés....510
12.3.1 Gyorsulási modus (BRISK, BRISKA, SOFT, SOFTA, DRIVE, DRIVEA)....510
12.3.2 Követő-tengelyek gyorsulásának befolyásolása (VELOLIMA, ACCLIMA, JERKLIMA)....512
12.3.3 Technológia-specifikus dinamika-értékek aktiválása (DYNNORM, DYNPOS, DYNROUGH, DYNSEMIFIN, DYNFINISH)....514

12.4 Mozgás elővezérléssel (FFWON, FFWOF)....516

12.5 Programozható kontúr-pontosság (CPRECON, CPRECOF)....517

12.6 Programlefutás előrefutás tárolóval (STARTFIFO, STOPFIFO, STOPRE)....519

12.7 Stop-Delay tartományt definiálni (DELAYFSTON, DELAYFSTOF)....522

12.8 Programhely tiltása SERUPRO számára (IPTRLOCK, IPTRUNLOCK)....525

12.9 Újra-rámenet a kontúrra (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL)....527

12.10 Mozgás-vezetés befolyásolása....536
12.10.1 Százalékos rándítás-korrekció (JERKLIM)....536
12.10.2 Százalékos sebesség-korrekció (VELOLIM)....537
12.10.3 Programpélda VELOLIM-ra és JERKLIM-re....539

12.11 Kontúr-/tájolás-tűrés programozása (CTOL, OTOL, ATOL)....540

12.12 Mondatváltás viselkedés aktív csatolásnál (CPBC)....544

13 Tengely-csatolások....545

13.1 Vontatás (TRAILON, TRAILOF)....545

13.2 Görbe-táblázatok (CTAB)....550
13.2.1 Görbe-táblázatot definiálni (CTABDEF, CATBEND)....550
13.2.2 Egy görbe-táblázat előfordulását megvizsgálni (CTABEXISTS)....557
13.2.3 Görbe-táblázatok törlése (CTABDEL)....557
13.2.4 Görbe-táblázatok zárolása törlésre és átírásra (CTABLOCK, CTABUNLOCK)....558
13.2.5 Görbe-táblázatok: Táblázat tulajdonságok megállapítása (CTABID, CTABISLOCK, CTABMEMTYP, CTABPERIOD)....559
13.2.6 Görbe-táblázat értékeinek olvasása (CTABTSV, CTABTEV, CTABTSP, CTABTEP, CTABSSV, CTABSEV, CTAB, CTABINV, CTABTMIN, CTABTMAX)....561
13.2.7 Görbe-táblázatok: Erőforrások használatának vizsgálata (CTABNO, CTABNOMEM, CTABFNO, CTABSEGID, CTABSEG, CTABFSEG, CTABMSEG, CTABPOLID, CTABPOL, CTABFPOL, CTABMPOL)....566

13.3 Tengely vezető érték csatolás (LEADON, LEADOF)....568

13.4 Elektronikus hajtómű (EG)....574
13.4.1 Elektronikus hajtómű definíció (EGDEF)....574
13.4.2 Elektronikus hajtóművet bekapcsolni (EGON, EGONSYN, EGONSYNE)....575
13.4.3 Elektronikus hajtómű kikapcsolása (EGOFS)....578
13.4.4 Egy elektronikus hajtómű definícióját törölni (EGDEL)....579
13.4.5 Fordulat-előtolás (G95) / Elektronikus hajtómű (FPR)....579

13.5 Szinkron-orsó....580
13.5.1 Szinkronorsó: Programozás (COUPDEF, COUPDEL, COUPON, COUPONC, COUPOF, COUPOFS, COUPRES, WAITC)....580

13.6 Generátoros csatolás (CP...)....591

13.7 Master/Slave csatolás (MASLDEF, MASLDEL, MASLON, MASLOF, MASLOFS)....598

14 Szinkron-akciók....601

14.1 Egy szinkronakció definíciója....601

15 Ingázás....603

15.1 Aszinkron ingázás (OS, OSP1, OSP2, OST1, OST2, OSCTRL, OSNSC, OSE, OSB)....603

15.2 Szinkron-akciókkal vezérelt ingázás (OSCILL)....608

16 Lyukasztás és sapkázás....615

16.1 Aktiválás / deaktiválás....615
16.1.1 Lyukasztás és sapkázás be-/kikapcsolás (SPOF, SON, PON, SONS, PONS, PDELAYON, PDELAYOF, PUNCHACC)....615

16.2 Automatikus út-felosztás....620
16.2.1 Út felosztás pályatengelyeknél....622
16.2.2 Út felosztás egyes tengelyeknél....624

17 Köszörülés....627

17.1 Köszörű-specifikus szerszám-felügyelet be-/kikapcsolása (TMON, TMOF)....627

18 További funkciók....629

18.1 Tengely funkciók (AXNAME, AX, SPI, AXTOSPI, ISAXIS, AXSTRING, MODAXVAL)....629

18.2 Átkapcsolható geometria-tengelyek (GEOAX)....632

18.3 Tengely-konténer (AXCTSWE, AXCTSWED, AXCTSWEC)....637

18.4 Várni érvényes tengelypozícióra (WAITENC)....639

18.5 Programozható paraméterkészlet-átkapcsolás (SCPARA)....641

18.6 NC-nyelv meglevő terjedelmének vizsgálata (STRINGIS)....643

18.7 Ablakot munkadarabprogramból interaktívan felhívni (MMC)....647

18.8 Program futásidő / munkadarab számláló....652
18.8.1 Program futásidő....652
18.8.2 Munkadarab számláló....655

18.9 Process DataShare - Kiadás egy külső készülékre/fájlba (EXTOPEN, WRITE, EXTCLOSE)....657

18.10 Vészjelzések (SETAL)....662

18.11 Kibővített leállítás és visszahúzás (ESR)....663
18.11.1 NC-vezérelt ESR....664
18.11.1.1 NC vezetésű visszahúzás (POLF, POLFA, POLFMASK, POLFMLIN)....664
18.11.1.2 NC vezetésű leállítás....668
18.11.2 Hajtás-alapú ESR....668
18.11.2.1 Hajtásalapú leállítás beállítása (ESRS)....668
18.11.2.2 Hajtásalapú visszahúzás beállítása (ESRS)....669

18.12 Nyersdarab definiálása (WORKPIECE)....671

18.13 Nyelvi módus átkapcsolása (G290, G291)....675

19 Saját leforgácsoló-program....677

19.1 A leforgácsolást támogató funkciók....677

19.2 Kontúr-táblázatot létrehozni (CONTPRON)....678

19.3 Kódolt kontúr-táblázatot létrehozni (CONTPRON)....684

19.4 Két kontúrelem metszéspontját megállapítani (INTERSEC)....688

19.5 Egy táblázat kontúrelemeinek mondatonkénti megtétele (EXECTAB)....690

19.6 Kör-adatok kiszámítása (CALCDAT)....691

19.7 Kontúr-feldolgozást kikapcsolni (EXECUTE)....693

20 Ciklusok külső programozása....695

20.1 Technológiai ciklusok....695
20.1.1 Bevezetés....695
20.1.2 Technológia-specifikus áttekintés....696
20.1.3 Lyuk-sor - HOLES1....698
20.1.4 Lyuk-kör - HOLES2....698
20.1.5 POCKET3 - Négyszögzseb marása....700
20.1.6 POCKET4 - Körzseb marása....703
20.1.7 SLOT1 - hosszvájat....705
20.1.8 SLOT2 - körvájat....708
20.1.9 LONGHOLE - Hosszlyuk....710
20.1.10 CYCLE60 - Gravírozás ciklus....712
20.1.11 CYCLE61 - Síkmarás....715
20.1.12 CYCLE62 - Kontúrhívás....717
20.1.13 CYCLE63 - Kontúrzsebet marni....718
20.1.14 CYCLE64 - Kontúrzsebet előfúrni....720
20.1.15 CYCLE70 - Menetmarás....721
20.1.16 CYCLE72 - Pályamarás....723
20.1.17 CYCLE76 - négyszögcsap marás....726
20.1.18 CYCLE77 - Körcsap marása....729
20.1.19 CYCLE78 - Furatmenet marás....731
20.1.20 CYCLE79 - sokszög....733
20.1.21 CYCLE81 - Fúrás, központozás....735
20.1.22 CYCLE82 - Fúrás, sík-süllyesztés....736
20.1.23 CYCLE83 - Mély-lyuk fúrás....739
20.1.24 CYCLE84 - Menetfúrás kiegyenlítő tokmány nélkül....742
20.1.25 CYCLE85 - dörzsölés....745
20.1.26 CYCLE86 - Kiesztergálás....746
20.1.27 CYCLE92 - Leszúrás....747

20.1.28 CYCLE95 - Kontúr leforgácsolás....749
20.1.29 CYCLE98 - Menetlánc....751
20.1.30 CYCLE99 - Menetesztergálás....755
20.1.31 CYCLE435 - Lehúzó koordináták beállítása....760
20.1.32 CYCLE495 - Profilozás....760
20.1.33 CYCLE800 - Billentés....762
20.1.34 CYCLE801 - Rács vagy keret....765
20.1.35 CYCLE802 - Tetszőleges pozíciók....767
20.1.36 CYCLE830 - Mélylyuk fúrása 2....769
20.1.37 CYCLE832 - High Speed Settings....775
20.1.38 CYCLE840 - Menetfúrás kiegyenlítő tokmánnyal....779
20.1.39 CYCLE899 - Nyitott vájat marása....782
20.1.40 CYCLE930 - Beszúrás....784
20.1.41 CYCLE940 - Szabadra szúrás formák....787
20.1.42 CYCLE951 - Leforgácsolás....789
20.1.43 CYCLE952 - Kontúr-szúrás....792
20.1.44 CYCLE4071 - hossz-köszörülés fogással a fordulóponton....798
20.1.45 CYCLE4072 - Hossz-köszörülés fogással a fordulóponton és megszakítás jel....799
20.1.46 CYCLE4073 - Hossz-köszörülés folyamatos fogásvétellel....803
20.1.47 CYCLE4074 - Hossz-köszörülés folyamatos fogásvétellel és megszakítás jellel....804
20.1.48 CYCLE4075 - Sík-köszörülés fogással a fordulóponton....807
20.1.49 CYCLE4077 - Sík-köszörülés fogással a fordulóponton és megszakítás jel....810
20.1.50 CYCLE4078 - Sík-köszörülés folyamatos fogásvétellel....813
20.1.51 CYCLE4079 - Sík-köszörülés közbenső fogásvétellel....815
20.1.52 GROUP_BEGIN - programblokk kezdete....817
20.1.53 GROUP_END - programblokk vége....818
20.1.54 GROUP_ADDEND - bejáratos kiegészítés vége....818
20.1.55 Peremfeltételek....818
20.1.55.1 Technológia skálázás ciklus maszkokon belül....818

20.2 Mérőciklusok....820

21 Táblázatok....821

21.1 utasítások....821

21.2 Utasítások: Rendelkezésre állás SINUMERIK 828D esetén....859
21.2.1 Esztergálás / marás vezérlés változatok....859
21.2.2 Köszörülés vezérlés változat....886

21.3 Aktuális nyelv a HMI-ben....914

A Függelék....915

A.1 Rövidítések listája....915

A.2 Dokumentáció áttekintés....924

Fogalmak....925

Index....947

Alapvető biztonsági utalások 1

1.1 Általános biztonsági utalások

⚠ FIGYELMEZTETÉS

Életveszély a biztonsági utalások és maradék kockázatok nem figyelembe vétele miatt

A megfelelő hardver dokumentációkban levő biztonsági utalások és maradék kockázatok nem figyelembe vétele súlyos sérüléses balesetet vagy halált okozhat.

- Tartsa be a hardver dokumentáció biztonsági utalásait..
- Vegye figyelembe a kockázatok megítélésénél a maradék kockázatokat.

⚠ FIGYELMEZTETÉS

Gép helytelen működése hibás vagy megváltoztatott paraméterezés következtében

A hibás vagy megváltoztatott paraméterezés miatt a gépen helytelen működés léphet fel, ami testi sérülést vagy halált okozhat.

- Védje a paraméterezést a jogosulatlan hozzáféréstől.
- A lehetséges helytelen működést megfelelő intézkedésekkel, pl. VÉSZ-ÁLLJ vagy VÉSZ-KI hárítsa el.

1.2 Garancia és szavatosság az alkalmazási példákra

Az alkalmazási példák kötelezettség nélküliek és nincs szavatolva a teljesség a konfiguráció és a kiépítés vonatkozásában és bármely előre nem látott esetben. Az alkalmazási példák nem vevő-specifikus megoldások, hanem segítséget nyújtanak tipikus feladatoknál.

Felhasználóként a leírt termékek szakszerű üzemeltetéséért Ön a felelős. Az alkalmazási példák nem mentik fel Önt a biztonságos eljárás kötelezettsége alól az alkalmazásnál, összeállításnál, üzemeltetésnél és karbantartásnál.

1.3 Industrial Security

Megjegyzés

Industrial Security

A Siemens olyan ipari biztonsági funkciókkal rendelkező termékeket és szolgáltatásokat kínál, melyek elősegítik a létesítmények, rendszerek, gépek és hálózatok biztonságos működését.

A létesítmények, rendszerek. gépek és hálózatok kiber-támadások elleni védelme érdekében egy, a technológia jelen állásának megfelelő átfogó ipari biztonsági koncepció kidolgozása (és folyamatos fejlesztése) szükséges. A Siemens termékei és megoldásai ennek a koncepciónak csak egy részét képezik.

Az ügyfél felelőssége, hogy megakadályozza a létesítmények, rendszerek, gépek és hálózatok jogosulatlan használatát. A rendszerek, gépek és komponensek a vállalati hálózathoz vagy az internethez csak akkor csatlakozhatnak, ha és ameddig ez szükséges, és a megfelelő védelmi intézkedések (pl. tűzfal használata vagy hálózatszegmentálás) végre lettek hajtva.

Figyelembe kell venni a Siemens megfelelő védelmi intézkedésekre vonatkozó javaslatait is. További információ találhatók az Industrial Security-ről:

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

A Siemens termékek és megoldások a még nagyobb biztonság érdekében folyamatosan tovább vannak fejlesztve. A Siemens kifejezetten javasolja a frissítést, amint a megfelelő frissítések rendelkezésre állnak, és a mindenkori legfrissebb termékverzió használatát. A régebbi vagy már nem támogatott verziók használata növelheti a kiber-támadások kockázatát.

A termék-frissítésekről szóló folyamatos tájékoztatásért iratkozzon fel a Siemens Industrial Security RSS hírfolyamára:

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

További információk az interneten találhatók:

Industrial Security tervezési kézikönyv (https://support.industry.siemens.com/cs/ww/en/view/108862708)

⚠ FIGYELMEZTETÉS

Bizonytalan üzemi állapotok a szoftver manipulációja miatt

A szoftver manipulációja, pl. vírusok, trójaiak, levelek vagy férgek által, az Ön berendezésének bizonytalan üzemi állapotait okozhatja, amelyek halált, súlyos testi sérülést vagy tárgyi károkat okozhatnak.

- Tartsa a szoftver aktuális állapotban.
- Integrálja az automatizálási és hajtás komponenseket a berendezés vagy gép egységes Industrial Security koncepciójába a technika aktuális szintjének megfelelően.
- Az Ön egységes Industrial Security koncepciójában vegye figyelembe az összes alkalmazott terméket.
- Védje a fájlokat a cserélhető tároló elemeken a kártékony szoftverektől megfelelő védelmi intézkedésekkel, pl. vírus keresőkkel.
- Védje a hajtást a jogosulatlan változtatásoktól, amennyiben a „Know-How védelem“ funkciót aktiválja.

Rugalmas NC programozás 2

2.1 Változók

A változók alkalmazásával a rendszeradatok és felhasználói adatok tartományokból, különösen a számítási és vezérlési funkciókkal kapcsolatban, az NC programokat és ciklusokat különösen rugalmasan alakíthatjuk ki.

- Rendszeradatok
  A rendszeradatokban vannak a rendszerben előre definiált változók. Ezeknek a változóknak definiált jelentésük van. Ezeket elsősorban a rendszerszoftver használja. A felhasználó ezeket a változókat az NC programokban és a ciklusokban olvashatja és írhatja. Példa: gépadatok, beállítási adatok, rendszerváltozók.
  Ámbár egy rendszeradat jelentése fixen megadott, a felhasználó a tulajdonságait újra definiálással bizonyos mértékben meg tudja változtatni.
  Lásd "Rendszeradatok, felhasználói adatok és NC utasítások újra definiálása (FRDEF) (Oldal 35)"
- Felhasználói adatok
  A felhasználói adatokban vannak a felhasználó által definiált változók. amelyek jelentését kizárólag a felhasználó határozza meg. Ezeket a rendszer nem értékeli ki.
  A felhasználói változók felosztása:
  - Előre definiált felhasználói változók
    Az előre definiált felhasználói változók a rendszerben már definiált változók, amelyeknek a szám gépadatokkal lesz paraméterezve. Ezeknek a változóknak a tulajdonságait a felhasználó illesztheti.
    Lásd "Rendszeradatok, felhasználói adatok és NC utasítások újra definiálása (FRDEF) (Oldal 35)".
  - Felhasználó által definiált változók
    A felhasználó által definiált változók olyan változók, melyeket a felhasználó definiál és amelyeket a rendszer a futásidőben hoz létre. Ezek számát, adattípusát, láthatóságát és összes többi tulajdonságát kizárólag a felhasználó adja meg.
    Lásd "Felhasználói változók definíciója (DEF) (Oldal 29)"

2.1.1 Rendszeradatok

A rendszeradatok a rendszerben előre definiált változók, amelyek lehetővé teszik az NC programokban és ciklusokban a hozzáférést a vezérlés aktuális paraméterezéséhez, továbbá a gép, a vezérlés és a folyamat állapotaihoz.

Előrefutás-változók

Az előrefutás-változók olyan rendszeradatok, amelyeket az előrefutás állapotában, vagyis a mondatnak, amelyikben a változó programozva van, értelmezése időpontjában lehet olvasni és írni. Az előrefutás-változók nem váltanak ki előrefutás-álljt.

Főfutás-változók

Az főfutás-változók olyan rendszeradatok, amelyeket a főfutás állapotában, vagyis a mondatnak, amelyikben a változó programozva van, értelmezése időpontjában lehet olvasni és írni. A főfutás-változók:

- rendszerváltozók, amelyeket a szinkronakciókban lehet programozni (olvasni/írni)
- rendszerváltozók, amelyeket az NC programban lehet programozni és előrefutás-álljt kiváltani (olvasni/írni)
- változók, amelyeket az NC programban lehet programozni és az értéket az előrefutásban megállapítani, de csak a főfutásban lesz írva (főfutás-szinkron: csak írni)

Prefix szisztéma

A rendszeradatok külön jelölésére normál esetben a név előtt egy prefix van, amely egy $ karakterből és az azt követő egy vagy két betűből és egy alsó-vonalból tevődik össze.

$ + 1. Betű	Jelentés: adatfajta
Előrefutás adatok (előrefutásban olvasható / írható rendszeradatok)	
$M	gépadatok [1]
$S	beállítási adatok, védőtartományok [1]
$T	szerszámkezelési adatok
$P	programozott értékek
$C	ISO-ciklusok ciklusváltozói
$O	opció adatok
R	R-paraméterek (számítási paraméterek) [2]
Főfutás adatok (főfutásban olvasható / írható rendszeradatok)	
$$M	gépadatok [1]
$$S	beállítási adatok [1]
$A	aktuális főfutás-adatok
$V	helyzetszabályozó adatok
$R	R-paraméterek (számítási paraméterek) [2]
[1] A gépadatok és beállítási adatok elő- vagy főfutam változóként kezelése attól függ, hogy egy vagy két $ karakterrel vannak-e írva. Az írásmód felhasználó-specifikusan szabadon választható. [2] Az R-paramétereknek használatánál a munkadarabprogramban / ciklusban előrefutás-változóként nem lesz prefix írva, pl R10. A szinkronakciókban főfutás-változóként használatnál a prefix egy $ karakterrel lesz írva, pl. $R10.	

2. betű	Jelentés: láthatóság
N	NC-globális változó (**N**C)
C	csatorna-specifikus változó (**C**hannel)
A	tengely-specifikus változó (**A**xis)

Peremfeltételek

Kivételek a prefix szisztematikában

A következő rendszerváltozók eltérnek a megadott prefix szisztematikától:

- $TC_...: A 2. betű, C, nem egy csatorna-specifikus, hanem szerszámtartó-specifikus rendszerváltozóra utal (TC = Tool Carrier)
- $P_ ...: csatorna-specifikus rendszerváltozók

Gépadatok és beállítási adatok használata a szinkronakciókban

A gépadatok és beállítási adatok használatánál szinkronakciókban a prefix-szel lehet meghatározni, hogy a gépadatot vagy a beállítási adatot előrefutás- vagy főfutás-szinkron lehet olvasni/írni..

Ha az adat a megmunkálás alatt változatlan marad, lehet előrefutás-szinkron olvasni. A gépadatok és beállítási adatok prefixét ehhez egy $ karakterrel kell kezdeni:

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

Ha az adat a megmunkálás alatt változik, főfutás-szinkron kell olvasni / írni. A gépadatok és beállítási adatok prefixét ehhez két $ karakterrel kell írni:

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

Megjegyzés

Gép- és beállítási-adatok írása

Egy gépadat vagy beállítási adat írásánál ügyelni kell arra, hogy az aktív hozzáférési fok a munkadarabprogram / ciklus végrehajtásánál az írás hozzáférést megengedje és az adat hatásossága "IMMEDIATE" legyen.

Irodalom

A összes rendszerváltozó teljes áttekintése itt található:

Jegyzékes kézikönyv Rendszerváltozók

Lásd még

Változók (Oldal 21)

2.1.2 Előre definiált felhasználói változók: Számítási paraméterek

2.1.2.1 Csatorna-specifikus számítási paraméterek (R)

A csatorna-specifikus számítási paraméterek vagy R-paraméterek előre definiált felhasználói változók az R megjelöléssel, REAL típusú mezőként definiálva. Történeti okokból az R-paraméterekre a mezőindexes írásmód pl. `R[10]` mellet mezőindex nélküli írásmód pl.`R10` is megengedett.

A szinkronakciókban való alkalmazásnál a $ karaktert kell elé írni pl. `$R10`.

Szintaxis

Előrefutás-változóként használatnál:
```
R<n>
R[<kifejezés>]
```

Főfutás-változóként használatnál:
```
$R<n>
$R[<kifejezés>]
```

Jelentés

R:	Jelölő előrefutás-változóként használatnál pl. munkadarabprogramban	
$R:	Jelölő főfutás-változóként használatnál pl. szinkronakciókban	
	típus:	REAL
	értéktartomány:	Nem exponenciális írásmódnál: ± (0.000 0001 ... 9999 9999) **Utalás:** Maximum 8 decimális hely megengedett
		Exponenciális írásmódnál: ± ($1*10^{-300}$ ... $1*10^{+300}$) **Utalás:** • Írásmód: <mantissza>EX<kitevő> pl. 8.2EX-3 • Maximum 10 jegy megengedett az előjellel és a tizedesponttal együtt.
<n>:	R-paraméter száma	
	típus:	INT
	értéktartomány:	0 - MAX_INDEX **Utalás** MAX_INDEX az R-paraméterek paraméterezett számából adódik:: MAX_INDEX = (MD28050 $MN_MM_NUM_R_PARAM) - 1
<kifejezés>:	mező-index A mezőindex lehet egy tetszőleges kifejezés, amennyiben a kifejezés eredményét INT adattípusba át lehet alakítani (INT, REAL, BOOL, CHAR)	

Példa

Hozzárendelés R-paraméterhez és az R-paraméterek használata matematikai funkciókban:

Programkód	Kommentár
R0=3.5678	; hozzárendelés előrefutásban
R[1]=-37.3	; hozzárendelés előrefutásban
R3=-7	; hozzárendelés előrefutásban
$R4=-0.1EX-5	; hozzárendelés főfutásban R4 = -0.1 * 10^-5
$R[6]=1.874EX8	; hozzárendelés főfutásban R6 = 1.874 * 10^8
R7=SIN(25.3)	; hozzárendelés előrefutásban
R[R2]=R10	; közvetett címzés R-paraméterrel
R[(R1+R2)*R3]=5	; közvetett címzés matematikai kifejezéssel
X=(R1+R2)	; mozgás X tengellyel az R1 és R2 összegéből adódó pozíci-óra
Z=SQRT(R1*R1+R2*R2)	; mozgás Z tengellyel a (R1^2 + R2^2) négyzetgyök pozícióra

Lásd még

Változók (Oldal 21)

2.1.2.2 Globális számítási paraméterek (RG)

Funkció

A csatorna-specifikus R-paraméterek mellett a felhasználónak rendelkezésére állnak a globális R-paraméterek is. Ezek a vezérlésben egyszer fordulnak elő és az összes csatornából írhatók/olvashatók.

A globális R-paramétereket használunk pl. az információknak az egyik csatornából a másikba átviteléhez. Egy másik példa a globális beállítások, amelyeket az összes csatorna ki kell értékeljen, mint pl. a nyersdarab kilógása az orsóból.

A globális R-paraméterek olvasása und írása a kezelőfelületen vagy az NC programban előrefutásban történik. Az alkalmazásuk szinkronakciókban vagy technológiai ciklusokban nem lehetséges.

Megjegyzés

A globális R-paraméterek olvasásánál és írásánál **nincs** csatorna szinkronizáció.

Mivel az írás és az olvasás előrefutásban történik, az időpont, amikor az egyik csatorna által írt érték a másik csatornában hatásos lesz, nem definiált.

Példa:

A csatorna 1-ben fut egy hurok egy globális R-paraméterrel, mint hurokszámlálóval. A csatorna 2 ír egy értéket ebbe a globális R-paraméterbe, ami a csatorna 1-ben a hurok megszakadását okozza. Azonban az összes, a csatorna 1-ben eddig az időpontig interpretált hurok még végre lesz hajtva. Az, hogy ez hány hurok, nincs definiálva, és többek között a csatorna terhelésétől függ.

A csatornák közötti szinkronizációt a felhasználó kell megoldja, pl WAIT jelölőkkel!

Szintaxis

Írás NC-programban

```
RG[<n>]=<érték>
RG[<kifejezés>]=<érték>
```

Olvasás NC-programban

```
R...=RG[<n>]
R...=RG[<kifejezés>]
```

Jelentés

RG:	**globális** R paraméterek NC címeinek alap neve **Utalás:** Az NC cím neve az MD15800 $MN_R_PARAM_NCK_NAME-ben állítható be	
<n>:	Globális R-paraméterek számai	
	típus:	INT
	értéktartomány:	0 ... MAX_INDEX **Utalás** MAX_INDEX a globális R-paraméterek paraméterezett számából adódik: MAX_INDEX = (MD18156 $MN_MM_NUM_R_PARAM_NCK) - 1
<kifejezés>:	A mezőindex lehet egy tetszőleges kifejezés, amennyiben a kifejezés eredményét INT adattípusba át lehet alakítani (INT, REAL, BOOL, CHAR)	

<érték>:	Globális R-paraméterek értéke	
	típus:	REAL
	értéktartomány:	Nem exponenciális írásmódnál: ± (0.000 0001 ... 9999 9999) **Utalás:** Maximum 8 decimális hely megengedett
		Exponenciális írásmódnál: ± ($1*10^{-300}$ ... $1*10^{+300}$) **Utalás**: • Írásmód: <mantissza>EX<kitevő> pl. 8.2EX-3 • Maximum 10 jegy megengedett az előjellel és a tizedesponttal együtt.

2.1.3 Előre definiált felhasználói változók: Link-változók

A Link-változókkal az "NCU-Link" funkció keretében egy hálózatban egymással összekötött NCU-k között ciklikusan adatokat lehet cserélni. Ezek lehetővé tesznek egy adatformátum specifikus hozzáférést a Link-változó tárolóhoz. A Link-változó tárolót a felhasználó a nagysága és az adatszerkezete szempontjából felhasználó- / gépgyártó-specifikusan adja meg.

A Link-változók rendszer-globális felhasználói változók, amelyeket létrehozott Link-kommunikációnál a Link-egyesülés összes NCU-ja a munkadarabprogramokból és ciklusokból olvasni és írni tud. A globális felhasználói változókkal (GUD) ellentétben a Link-változók a szinkronakciókban is alkalmazhatók.

Az aktív NCU-Link nélküli berendezésekben a Link-változók a vezérlésekben helyileg használhatók a globális felhasználói változók (GUD) mellett kiegészítő globális felhasználói változókként.

Szintaxis

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

Jelentés

$A_DLB:	BYTE adatformátumú Link-változó (1 bájt)	
	adattípus:	UINT
	értéktartomány:	0 ... 255
$A_DLW:	WORD adatformátumú Link-változó (2 bájt)	
	adattípus:	INT
	értéktartomány:	-32768 ... 32767

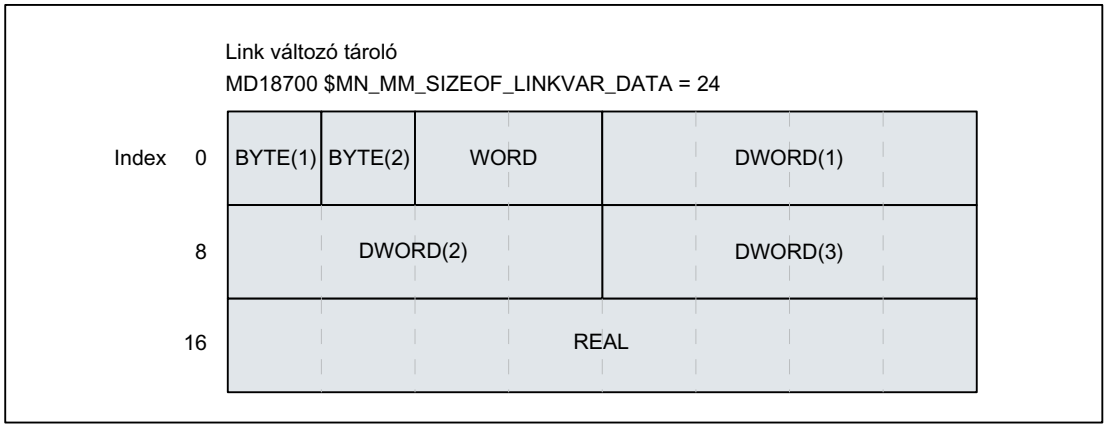
`$A_DLD`:	DWORD adatformátumú Link-változó (4 bájt)	
	adattípus:	INT
	értéktartomány:	-2147483648 ... 2147483647
`$A_DLR`:	REAL adatformátumú Link-változó (8 bájt)	
	adattípus:	REAL
	értéktartomány:	±(2,2*10^{-308} … 1,8*10^{+308})
`<Index>`:	cím-index bájtban, a Link-változó tároló elejétől számítva	
	adattípus:	INT
	értéktartomány:	0 - MAX_INDEX **Utalás** • Az MAX_INDEX a Link-változó tároló paraméterezett méretéből adódik: MAX_INDEX = (MD18700 $MN_MM_SIZEOF_LINKVAR_DATA) - 1 • Csak olyan indexeket szabad programozni, hogy a Link-változó tárolóban címzett bájtok egy adatformátum határra essenek ⇒ index = n * bájt, n = 0, 1, 2, ... – `$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, ...

Példa

Az automatizálási berendezésben 2 NCU van (NCU1 és NCU2). Az NCU1-hez az AX2 géptengely van csatlakoztatva, amelyet Link-tengelyként az NCU2 fog mozgatni.

Az NCU1 ciklikusan írja az AX2 tengely áram-valósértékét ($VA_CURR) a Link-változó tárolóba. Az NCU2 ciklikusan olvassa a Link-kommunikációval az átadott áram-valósértéket és a túllépésnél a 61000 vészjelzést adja.

A Link-változó tároló adatszerkezetét a következő kép ábrázolja. Az áram-valósérték a REAL értékkel lesz átvive.



NCU1

Az NCU1 egy statikus szinkronakcióban ciklikusan IPO-ütemben írja az AX2 tengely áram-valósértékét a $A_DLR[16] Link-változóval a Link-változó tárolóba.

Programkód

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

NCU2

Az NCU2 egy statikus szinkronakcióban ciklikusan IPO-ütemben olvassa az AX2 tengely áram-valósértékét a $A_DLR[16] Link-változóval a Link-változó tárolóból. Ha az áram-valósérték nagyobb, mint 23.0 A, a 61000 vészjelzés lesz kijelezve.

Programkód

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

Lásd még

Változók (Oldal 21)

2.1.4 Felhasználói változók definíciója (DEF)

A `DEF` utasítással tudunk felhasználó-specifikus változókat vagy felhasználói változókat (User Data) definiálni és azoknak értéket adni.

A felhasználói változóknak az érvényességi tartományuk, vagyis azon tartomány, amelyben láthatóak, szerint a következő kategóriáik vannak:

- Lokális felhasználói változók (LUD)
  A lokális felhasználói változók (LUD) olyan változók, amelyek egy olyan NC programban vannak definiálva, amely a feldolgozás időpontjában nem a főprogram. Ezek az NC program felhívásánál lesznek létrehozva és a program-vége Reset-tel ill. a következő vezérlés-felfutásnál törölve lesznek. A LUD-hoz csak az az NC program tud hozzáférni, amelyikben definiálva lett.
- Program-globális felhasználói változók (PUD)
  A program-globális felhasználói változók (PUD) olyan változók, amelyek egy főprogramként használt NC programban lettek definiálva. Ezek az NC program felhívásánál lesznek létrehozva és a program-vége Reset-tel ill. a következő vezérlés-felfutásnál törölve lesznek. A PUD-hoz a főprogramban és az összes alprogramban hozzá lehet férni.

  Megjegyzés

  Program-globális felhasználói változók (PUD) rendelkezésre állása

  A főprogramban definiált program-globális felhasználói változók (PUD) csak akkor állnak rendelkezésre az alprogramban, ha a következő gépadat be van állítva:

  MD11120 $MN_LUD_EXTENDED_SCOPE = 1

  Az MD11120 = 0 értékével főprogramban definiált program-globális felhasználói változók csak a főprogramban állnak rendelkezésre.

- Globális felhasználói változók (GUD)
  A globális felhasználói változók (GUD) NC- ill. csatorna-globális változók, amelyek egy adatmodulban (SGUD, MGUD, UGUD, GUD4 ... GUD9) vannak definiálva és megmaradnak a program-vége Reset ill. a következő vezérlés felfutás után is. A GUD-hoz minden NC programban hozzá lehet férni.

A felhasználói változókat az alkalmazásuk (olvasás / írás) előtt definiálni kell. Ennél a következő szabályokat kell figyelembe venni:

- A GUD-okat mindig egy definíciós fájlban, pl. _N_DEF_DIR/_N_UGUD_DEF, kell definiálni.
- A PUD-okat és LUD-okat egy NC program definíciós részében kell definiálni.
- Az adatok definíciója egy önálló mondatban kell legyen.
- Egy adat-definícióban csak egy adattípust lehet használni.
- Adat-definíciókként azonos adattípusúra több változót lehet definiálni.

Szintaxis

LUD és PUD

```
DEF <típus> <fiz. egység> <határértékek> <név>[<érték_1>, <érték_2>,
<érték_3>]=<kezdő_érték>
```

GUD

```
DEF <tartomány> <VL_Stop> <hozzáférési_jogok> <adat-osztály> <típus>
<fiz_egység> <határértékek> <név>[<érték_1>, <érték_2>,
<érték_3>]=<Init_érték>
```

Jelentés

DEF:	utasítás GUD, PUD és LUD felhasználói változók definíciójára	
<tartomány>:	érvényességi tartomány, csak GUD-nál számít:	
	NC:	NC-globális felhasználói változó
	CHAN:	csatorna-globális felhasználói változó
<EF_állj>:	előrefutás-állj, csak GUD-nál számít (opcionális)	
	SYNR:	előrefutás-állj olvasásnál
	SYNW:	előrefutás-állj írásnál
	SYNRW:	előrefutás-állj olvasásnál/írásnál
<hozzáférési_jogok>:	Védelmi fokozat GUD írásra / olvasásra NC programból vagy BTSS-ről (opcionális)	
	APRP <védelmi_fokozat>:	olvasás: NC program
	APWP <védelmi_fokozat>:	írás: NC program
	APRB <védelmi_fokozat>:	olvasás: BTSS
	APWB <védelmi_fokozat>:	írás: BTSS
	<védelmi fokozat>:	értéktartomány: 0 ... 7
	lásd "Tulajdonság: Hozzáférési jog (APR, APW, APRP, APWP, APRB, APWB) (Oldal 45)"	
<adat-osztály>:	adat-osztály hozzárendelés (csak SINUMERIK 828D)	
	DCM:	adat-osztály M (= manufacturer)
	DCI:	adat-osztály I (= Individual)
	DCU:	adat-osztály U (= user)
	Lásd "Tulajdonság: Adat-osztály (DCM, DCI, DCU) - csak SINUMERIK 828D (Oldal 50)".	
<típus>:	adattípus:	
	INT:	egészszám előjellel
	REAL:	valósszám (LONG REAL IEEE szerint)
	BOOL:	igazságérték IGAZ (1) / HAMIS (0)
	CHAR:	ASCII karakter
	STRING[<MaxHossz>]:	adott hosszú karakterlánc
	AXIS:	tengely-/orsó-jelölő
	FRAME:	geometriai adatok egy statikus transzformációhoz
	lásd "Adattípusok (Oldal 57)"	
<fiz_egység>:	fizikai egység (opcionális)	
	PHU <egység>:	fizikai egység
	lásd "Tulajdonság: Fizikai egység (PHU) (Oldal 43)"	

<határértékek>:	alsó vagy felső határérték (opcionális)	
	LLI <határérték>:	alsó határérték (lower limit)
	ULI <határérték>:	felső határérték (upper limit)
	lásd "Tulajdonság: Határértékek (LLI, ULI) (Oldal 41)"	
<név>:	változók neve **Utalás** • maximum 31 karakter • Az első két jelnek betűnek és/vagy alsó-vonalnak kell lennie. • A "$" karakter a rendszerváltozókra foglalt és nem szabad használni.	
[<érték_1>, <érték_2>, <érték_3>]:	mező-méretek megadása 1 .. max, 3 dimenziójú mező-változókra (opcionális) Mező-változók inicializálásához lásd" Mező-változók (DEF, SET, REP) definíciója és inicializálása (Oldal 52)"	
<inic_érték>:	inicializálási érték (opcionális) lásd "Tulajdonság: Inicializálási érték (Oldal 38)" Mező-változók inicializálásához lásd" Mező-változók (DEF, SET, REP) definíciója és inicializálása (Oldal 52)"	

Példák

Példa 1: Felhasználói változók definíciói adatmodulban gépgyártónak

Programkód	Kommentár
%_N_MGUD_DEF	; GUD modul: Gépgyártó
$PATH=/_N_DEF_DIR	
DEF CHAN REAL PHU 24 LLI 0 ULI 10 STROM_1, STROM_2	
; leírás	
; két GUD definíciója: STROM_1, STROM_2	
; érvényességi tartomány: csatornában	
; adattípus: REAL	
; VL-Stop: nincs programozva => alapérték = nincs VL-Stop	
; fizikai egység: 24 = [A]	
; határértékek: Low = 0.0, High = 10.0	
; hozzáférési jogok: nincs programozva => alapérték = 7 = kulcsos-kapcsoló állás 0	
; inicializálási érték: nincs programozva => alapérték = 0.0	
DEF NCK REAL PHU 13 LLI 10 APWP 3 APRP 3 APWB 0 APRB 2 ZEIT_1=12, ZEIT_2=45	
; leírás	
; két GUD definíciója: ZEIT_1, ZEIT_2	
; érvényességi tartomány: NC-szerte	
; adattípus: REAL	
; VL-Stop: nincs programozva => alapérték = nincs VL-Stop	
; fizikai egység: 13 = [s]	
; határértékek: Low = 10.0, High = nincs programozva => felső definíciós tartomány határ	

Programkód	Kommentár
; hozzáférési jogok:	
; NC program írás/olvasás = 3 = végfelhasználó	
; BTSS: írás = 0 = Siemens, olvasás = 3 = végfelhasználó	
; inicializálási érték: ZEIT_1 = 12.0, ZEIT_2 = 45.0	
DEF NCK APWP 3 APRP 3 APWB 0 APRB 3 STRING[5] GUD5_NAME = "COUNTER"	
; leírás	
; egy GUD definíciója: GUD5_NAME	
; érvényességi tartomány: NC-szerte	
; adattípus: STRING, max. 5 karakter	
; VL-Stop: nincs programozva => alapérték = nincs VL-Stop	
; fizikai egység: nincs programozva => alapérték = 0 = nincs fizikai egység	
; határértékek: nincs programozva => definíciós tartomány határok: Low = 0, High = 255	
; hozzáférési jogok:	
; NC program írás/olvasás = 3 = végfelhasználó	
; BTSS: írás = 0 = Siemens, olvasás = 3 = végfelhasználó	
; inicializálási érték: "COUNTER"	
M30	

Példa 2: Program-globális és -lokális felhasználói változók (PUD / LUD)

Programkód	Kommentár
PROC MAIN	; főprogram:
DEF INT VAR1	; PUD definíció
...	
SUB2	; alprogramhívás
...	
M30	

Programkód	Kommentár
PROC SUB2	; SUB2 alprogram
DEF INT VAR2	; LUD definíció
...	
IF (VAR1==1)	; PUD olvasás
VAR1=VAR1+1	; PUD olvasás és írás
VAR2=1	; LUD írás
ENDIF	
SUB3	; alprogramhívás
...	
M17	

Programkód	Kommentár
PROC SUB3	; SUB3 alprogram

```
Programkód                      Kommentár
...
IF (VAR1==1)                    ; PUD olvasás
 VAR1=VAR1+1                    ; PUD olvasás és írás
 VAR2=1                         ; hiba: LUD a SUB2-ből nem ismert
ENDIF
...
M17
```

Példa 3: Felhasználói változó definíció és alkalmazás, AXIS adattípus

```
Programkód                                  Kommentár
DEF AXIS ABSZISSE                           ; 1. geometria-tengely
DEF AXIS SPINDLE                            ; orsó
...
IF ISAXIS(1) == FALSE GOTOF WEITER
 ABSZISSE = $P_AXN1
WEITER:
...
SPINDLE=(S1)                                ; 1. Orsó
OVRA[SPINDLE]=80                            ; orsó-override = 80%
SPINDLE=(S3)                                ; 3. Orsó
```

Peremfeltételek

Globális felhasználói változók (GUD)

A globális felhasználói változók (GUD) definíciója keretében a következő gépadatokat kell figyelembe venni:

Nr.	Jelölő: $MN_	Jelentés
11140	GUD_AREA_ SAVE_TAB	GUD modulok kiegészítő mentése
18118 [1]	MM_NUM_GUD_MODULES	GUD fájlok száma az aktív fájlrendszerben
18120 [1]	MM_NUM_GUD_NAMES_NCK	globális GUD nevek száma
18130 [1]	MM_NUM_GUD_NAMES_CHAN	csatorna-spec. GUD nevek
18140 [1]	MM_NUM_GUD_NAMES_AXIS	tengely-spec. GUD nevek
18150 [1]	MM_GUD_VALUES_MEM	tárolóhely globális GUD értékekre
18660 [1]	MM_NUM_SYNACT_GUD_REAL	REAL adattípusú létrehozható GUD-ok száma
18661 [1]	MM_NUM_SYNACT_GUD_INT	INT adattípusú létrehozható GUD-ok száma
18662 [1]	MM_NUM_SYNACT_GUD_BOOL	BOOL adattípusú létrehozható GUD-ok száma
18663 [1]	MM_NUM_SYNACT_GUD_AXIS	AXIS adattípusú létrehozható GUD-ok száma
18664 [1]	MM_NUM_SYNACT_GUD_CHAR	CHAR adattípusú létrehozható GUD-ok száma
18665 [1]	MM_NUM_SYNACT_GUD_STRING	STRING adattípusú létrehozható GUD-ok szá-ma

[1] MD a SINUMERIK 828D-nél csak olvasható!

AXIS adattípusú NC-globális felhasználói változók csatornákat átfogó alkalmazása

Egy AXIS adattípusú NC-globális felhasználói változót, amely a definíciónál az adatmodulban egy tengely-jelölővel lett inicializálva, csak akkor lehet az NC különböző csatornáiban használni, hat a tengelynek ezekben a csatornákban azonos csatornatengely-száma van.

Ha ez nem így van, a változót az NC program elején kell betölteni vagy, mint a következő példában az AXNAME(...) funkciót kell használni (lásd"Tengely funkciók (AXNAME, AX, SPI, AXTOSPI, ISAXIS, AXSTRING, MODAXVAL) (Oldal 629)").

Programkód	Kommentár
DEF NCK STRING[5] ACHSE="X"	; definíció az adatmodulban
...	
N100 AX[AXNAME(ACHSE)]=111 G00	; felhasználás az NC programban

2.1.5 Rendszeradatok, felhasználói adatok és NC utasítások újra definiálása (FRDEF)

A REDEF utasítással a rendszeradatok, felhasználói adatok és NC utasítások tulajdonságait meg lehet változtatni. Az újra definiálás előfeltétele, hogy időben a megfelelő definíció után legyen végrehajtva.

Az újra definiálásnál nem lehet több jellemzőt egyszerre megváltoztatni. Minden megváltoztatandó tulajdonságra egy külön REDEF utasítást kell programozni.

Ha több konkuráló tulajdonság-változtatás van programozva, mindig az utolsó változtatás aktív.

Attribútum értékek visszaállítása

Az REDEF-fel definiált hozzáférési jog és inicializálási időpont attribútumokat vissza lehet állítani az alaphelyzet értékekre az REDEF újbóli programozását követően a változó vagy NC utasítást nevével:

- hozzáférési jogok: védelmi fok 7
- inicializálási időpont: nincs inicializálás ill. az aktuális érték megtartása

Újra definiálható tulajdonságok

Lásd "Definiálható és újra definiálható tulajdonságok áttekintése (Oldal 51)".

Lokális felhasználói változók (PUD / LUD)

A lokális felhasználói változókra (PUD / LUD) nem lehetséges az újra definiálás.

Szintaxis

```
REDEF <név> <EF_állj>
REDEF <név> <fiz_egység>
REDEF <név> <határértékek>
REDEF <név> <Hozzáférési jogok>
REDEF <név> <inic_időpont>
```

```
REDEF <név> <inic_időpont> <inic_érték>
REDEF <név> <adat-osztály>
REDEF <név>
```

Jelentés

REDEF:	utasítás egy adott attribútum újra újra definiálásához ill. a rendszerváltozók, felhasználói változók és NC utasítások "hozzáférési jog" és/vagy "inicializálási időpont" attribútumának visszaállításához	
<név>:	egy már definiált változó vagy NC utasítás neve	
<VL-Stop>:	előrefutás-állj	
	SYNR:	előrefutás-állj olvasásnál
	SYNW:	előrefutás-állj írásnál
	SYNRW:	előrefutás-állj olvasásnál/írásnál
<fiz_egység>:	fizikai egység	
	PHU <egység>:	fizikai egység
	Lásd "Tulajdonság: Fizikai egység (PHU) (Oldal 43)". **Utalás** Nem definiálható újra: • Rendszerváltozók • Globális felhasználói adatok (GUD) adattípusokhoz: BOOL, AXIS, STRING, FRAME	
<határértékek>:	alsó / felső határérték	
	LLI <határérték>:	alsó határérték (lower limit)
	ULI <határérték>:	felső határérték (upper limit)
	Lásd "Tulajdonság: Határértékek (LLI, ULI) (Oldal 41)". **Utalás** Nem definiálható újra: • rendszerváltozók • globális felhasználói adatok (GUD) következő adattípusokhoz: BOOL, AXIS, STRING, FRAME	
<hozzáférési_jogok>:	hozzáférési jogok írásra / olvasásra munkadarabprogramból vagy BTSS-ről	
	APX <védelmi-fokozat>:	végrehajtás: NC nyelvi elem
	APRP <védelmi_fokozat>:	olvasás: munkadarabprogram
	APWP <védelmi_fokozat>:	írás: munkadarabprogram
	APRB <védelmi_fokozat>:	olvasás: BTSS
	APWB <védelmi_fokozat>:	írás: BTSS
	<védelmi fokozat>:	értéktartomány: 0 ... 7
	Lásd "Tulajdonság: Hozzáférési jog (APR, APW, APRP, APWP, APRB, APWB) (Oldal 45)".	

<inic_időpont>:	változók újra-inicializálásának időpontja	
	INIPO:	Power On
	INIRE:	főprogram vége, NC-Reset vagy Power On
	INICF:	NEWCONF vagy főprogram vége, NC-Reset vagy Power On
	PRLOC:	főprogram vége, NC-Reset helyi változások után vagy Power On
	Lásd "Tulajdonság: Inicializálási érték (Oldal 38)".	
<inic_érték>:	inicializálási érték Az inicializálási értek újra definiálásánál mindig meg kell adni egy újra inicializálási időpontot (lásd <inic_időpont> is. Lásd "Tulajdonság: Inicializálási érték (Oldal 38)". Mező-változók inicializálásához lásd "Mező-változók (DEF, SET, REP) definíciója és inicializálása (Oldal 52)". **Utalás** Nem definiálható újra a rendszerváltozókra a beállítási adatokon kívül.	
<adat-osztály>:	adat-osztály hozzárendelés (csak SINUMERIK 828D)	
	DCM:	adat-osztály M (= manufacturer)
	DCI:	adat-osztály I (= Individual)
	DCU:	adat-osztály U (= user)
	Lásd "Tulajdonság: Adat-osztály (DCM, DCI, DCU) - csak SINUMERIK 828D (Oldal 50)".	

Példa

A $TC_DPCx rendszerváltozó újra definiálása adatmodulban gépgyártó számára

Programkód

```
%_N_MGUD_DEF                                ; GUD modul: Gépgyártó
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
```

N100-hoz: írás hozzáférési jog: BTSS = védelmi fok 2, munkadarabprogram = védelmi fok 3

N200-hoz: fizikai egység [%]

N300-hoz: alsó határérték = 0, felső határérték = 200

N400-hoz:	A mező-változó a PowerOn-nál négy értékkel lesz inicializálva.
N800 / N900-hoz	"hozzáférési jog" és/vagy "inicializálási időpont" attribútumok visszaállítása

Megjegyzés

ACCESS fájlok alkalmazása

Az ACCESS fájlok használatánál a hozzáférési jogok újra definiálását a _N_MGUD_DEF-ből át kell helyezni a _N_MACCESS_DEF-be.

Peremfeltételek

Granularitás

Az újra definiálás mindig a teljes, a neve által egyértelműen megadott változóra vonatkozik. Nem lehetséges pl. a mező-változóknál az egyes mezőelemekhez eltérő tulajdonság-értékeket hozzárendelni.

2.1.6 Tulajdonság: Inicializálási érték

Felhasználói változók definíciója (DEF)

A definíciónál a következő felhasználói változókra lehet megadni inicializálási értéket:

- Globális felhasználói változók (GUD)
- Program-globális felhasználói változók (PUD)
- Lokális felhasználói változók (LUD)

Rendszerváltozók és felhasználói változók újra definíciója (REDEF)

Az újra definíciónál a következő változókra lehet megadni inicializálási értéket:

- rendszeradatok
  - beállítási-adatok
- felhasználói adatok
  - R-paraméterek
  - szinkronizációs változók ($AC_MARKER, $AC_PARAM, $AC_TIMER)
  - szinkronakciók–GUD (SYG_xy[ ], x=R, I, B, A, C, S és y=S, M, U, 4, ..., 9)
  - EPS-paraméterek
  - OEM szerszámadatok
  - OEM táradatok
  - globális felhasználói változók (GUD)

Újra inicializálási időpont

Az újra definíciónál meg lehet adni egy időpontot, amikor a változót újra kell inicializálni, vagyis ismét az inicializálási értékre kell állítani:

- `INIPO` (Power On)
  A változó Power On-nál újra lesz inicializálva.
- `INIRE` (Reset)
  A változó NC-Reset-nél, BAG-Reset-nél, munkadarabprogram végénél (`M02` / `M30`) vagy Power On-nál újra lesz inicializálva.
- `INICF` (NEWCONF)
  A változó "gépadatokat hatásossá tenni" funkciónál a HMI-től, `NEWCONFIG` munkadarab utasításnál vagy NC-Reset-nél, BAG-Reset-nél, munkadarabprogram végénél (M02 / M30) vagy Power On-nál újra lesz inicializálva.
- `PRLOC` (program-lokális változás)
  A változó csak akkor lesz NC-Reset-nél, BAG-Reset-nél vagy a munkadarabprogram végénél (M02 / M30) újra inicializálva, ha az aktuális munkadarabprogram keretében lett megváltoztatva.
  A `PRLOC` tulajdonságot csak a programozható beállítási adatokkal (lásd a következő táblázatot) összefüggésben szabad használni..

Táblázat 2-1 Programozható beállítási adatok

Szám	Jelölő	G-utasítás [1)]
42000	$SC_THREAD_START_ANGLE	`SF`
42010	$SC_THREAD_RAMP_DISP	`DITS` / `DITE`
42400	$SA_PUNCH_DWELLTIME	`PDELAYON`
42800	$SA_SPIND_ASSIGN_TAB	`SETMS`
43210	$SA_SPIND_MIN_VELO_G25	`G25`
43220	$SA_SPIND_MAX_VELO_G26	`G26`
43230	$SA_SPIND_MAX_VELO_LIMS	`LIMS`
43300	$SA_ASSIGN_FEED_PER_REV_SOURCE	`FPRAON`

Szám	Jelölő	G-utasítás [1]
43420	$SA_WORKAREA_LIMIT_PLUS	G26
43430	$SA_WORKAREA_LIMIT_MINUS	G25
43510	$SA_FIXED_STOP_TORQUE	FXST
43520	$SA_FIXED_STOP_WINDOW	FXSW
43700	$SA_OSCILL_REVERSE_POS1	OSP1
43710	$SA_OSCILL_REVERSE_POS2	OSP2
43720	$SA_OSCILL_DWELL_TIME1	OST1
43730	$SA_OSCILL_DWELL_TIME2	OST2
43740	$SA_OSCILL_VELO	FA
43750	$SA_OSCILL_NUM_SPARK_CYCLES	OSNSC
43760	$SA_OSCILL_END_POS	OSE
43770	$SA_OSCILL_CTRL_MASK	OSCTRL
43780	$SA_OSCILL_IS_ACTIVE	OS
43790	$SA_OSCILL_START_POS	OSB
1) ezzel a G-utasítással lesz a beállítási adat megszólítva		

Peremfeltételek

Inicializálási érték: globális felhasználói változók (GUD)

- A globális felhasználói változókra (GUD) az NC érvényességi tartománnyal az inicializálási időpontra csak INIPO (Power On) adható meg.
- A globális felhasználói változókra (GUD) a CHAN érvényességi tartománnyal az inicializálási időpontra az INIPO (Power On) mellett INIRE (Reset) vagy INICF (NEWCONF) is megadható.
- A globális felhasználói változóknál (GUD) a CHAN érvényességi tartománnyal és INIRE (Reset) vagy INICF (NEWCONF) inicializálási időponttal, az NC-Reset-nél, BAG-Reset-nél és "gépadatokat hatásossá váltásánál" a változók csak azokban a csatornákban lesznek újra inicializálva, amelyekben a megnevezett esemény ki lett váltva.

Inicializálási érték: FRAME adattípus

A FRAME adattípusú változókra nem szabad inicializálási értéket megadni. FRAME adattípusú változók közvetetten mindig az alap-frame-mel lesznek inicializálva.

Inicializálási érték: CHAR adattípus

A CHAR adattípusú változókra az ASCII kód (0...255) lehet a megfelelő ASCII karaktert is idézőjelek között programozni, pl. "A"

Inicializálási érték: STRING adattípus

A STRING adattípusú változókra a karakterláncot idézőjelek között kell megadni, pl.: ...= "MASCHINE_1"

Inicializálási érték: AXIS adattípus

Az AXIS adattípusú változókra a tengely-jelölőt a bővített cím-írásmódnál zárójelekben kell megadni, pl.: ...=(X3)

Inicializálási érték: rendszerváltozó

A rendszerváltozókra az újra definíciónál nem lehet felhasználó-specifikus inicializálási értékeket megadni: A rendszerváltozók inicializálási értékét a rendszer adja meg. Az újra definícióval a reinicializálás időpontot (`INIRE`, `INICF`) a rendszerváltozókhoz meg lehet változtatni.

Közvetett inicializálási érték: AXIS adattípus

Az `AXIS` adattípusú változókra a következő közvetett inicializálási érték lesz használva:

- rendszeradatok: "első geometria-tengely"
- szinkronizációs GUD (jelölés: SYG_A*), PUD, LUD:
  tengely-jelölő gépadatból: MD20082 $MC_AXCONF_CHANAX_DEFAULT_NAME

Közvetett inicializálási érték: szerszám- és tár-adatok

A szerszám- és táradatokra az inicializálási értékeket a következő gépadatokkal lehet megadni: MD17520 $MN_TOOL_DEFAULT_DATA_MASK

Megjegyzés

Szinkronizálás

Az események szinkronizálása, amit egy globális változó újra inicializálása vált ki ezen változónak egy más helyen való olvasásával, kizárólag a felhasználó / gépgyártó felelőssége.

Lásd még

Változók (Oldal 21)

2.1.7 Tulajdonság: Határértékek (LLI, ULI)

A definíciós tartományokra felső és alsó határértéket csak a következő adattípusokra lehet megadni:

- INT
- REAL
- CHAR

Felhasználói változók definíciója (DEF): Határértékek és implicit inicializálási értékek

Ha a felhasználói változók definíciójánál a fenti adattípusok egyikénél nincs inicializálási érték definiálva, a változó az adattípus implicit inicializálási értékére lesz beállítva:

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

Ha az implicit inicializálási érték a programozott határértékekkel megadott definíciós tartományon kívül esik, azzal a határértékkel lesz inicializálva, amelyik az implicit inicializálási értékhez a legközelebb van.

- implicit inicializálási érték < alsó határérték (LLI) ⇒ inicializálási érték = alsó határérték
- implicit inicializálási érték > felső határérték (ULI) ⇒ inicializálási érték = felső határérték

Példák:

Programkód	Kommentár
DEF REAL GUD1	; alsó határérték = definíciós tartomány határ ; felső határérték = definíciós tartomány határ ; nincs inicializálási érték programozva ; => implicit inicializálási érték = 0.0
DEF REAL LLI 5.0 GUD2	; alsó határérték = 5.0 ; felső határérték = definíciós tartomány határ ; => inicializálási érték = 5.0
DEF REAL ULI -5 GUD3	; alsó határérték = definíciós tartomány határ ; felső határérték = -5.0 ; => inicializálási érték = -5.0

Felhasználói változók újra definíciója (REDEF): Határértékek ás aktuális valósértékek

Ha az újra definíciónál egy felhasználói változó határértékei úgy lesznek megváltoztatva, hogy az aktuális valósérték az új definíciós tartományon kívül van, egy vészjelzés történik és a határértékek nem lesznek átvéve.

Megjegyzés

Felhasználói változók újra definíciója (REDEF):

Egy felhasználói változó határértékeinek újra definiálásánál ügyelni kell a következő értékek konzisztens változtatására:

- határértékek
- Valósérték
- Inicializálási érték újra definiálásnál és automatikus újra definiálásnál INIPO, INIRE vagy INICF miatt

---

Lásd még

Változók (Oldal 21)

2.1.8 Tulajdonság: Fizikai egység (PHU)

Fizikai egységet csak a következő adattípusú változókra lehet megadni:

- INT
- REAL

Programozható fizikai egységek (PHU)

A fizikai egység megadása fixpontos számként történik: `PHU <egység>`

A következő fizikai egységeket lehet programozni:

<Egység>	Jelentés	Fizikai egység
0	nincs fizikai egység	-
1	lineáris- vagy szögpozíció 1)2)	[mm], [hüvelyk], [fok]
2	lineáris pozíció 2)	[mm], [hüvelyk]
3	szögpozíció	[fok]
4	lineáris- vagy szögsebesség 1)2)	[mm/perc], [hüvelyk/perc], [ford/perc]
5	lineáris sebesség 2)	[mm/perc]
6	szögsebesség	[ford/perc]
7	lineáris- vagy szög-gyorsulás 1)2)	[m/s^2], [$hüv/s^2$], [$ford/s^2$]
8	lineáris gyorsulás 2)	[m/s^2], [$hüv/s^2$]
9	szög-gyorsulás	[$ford/s^2$]
10	lineáris- vagy szög-rándítás 1)2)	[m/s^3], [$hüv/s^3$], [$ford/s^3$]
11	lineáris rándítás 2)	[m/s^3], [$hüv/s^3$]
12	szög-rándítás	[$ford/s^3$]
13	idő	[s]
14	helyzetszabályzó erősítés	[16.667/s]
15	fordulati előtolás 2)	[mm/ford], [hüvelyk/ford]
16	hőmérséklet-kompenzáció 1)2)	[mm], [hüvelyk]
18	erő	[N]
19	tömeg	[kg]
20	tehetetlenségi nyomaték3)	[kgm^2]
21	százalék	[%]
22	frekvencia	[Hz]
23	feszültség	[V]
24	áram	[A]
25	hőmérséklet	[°C]
26	szög	[fok]
27	KV	[1000/perc]
28	lineáris- vagy szögpozíció 3)	[mm], [hüvelyk], [fok]
29	vágósebesség 2):	[m/perc], [láb/perc]
30	kerületi sebesség 2)	[m/perc], [láb/perc]
31	ellenállás	[Ohm]

<Egység>	Jelentés	Fizikai egység
32	induktivitás	[mH]
33	forgatónyomaték 3)	[Nm]
34	forgatónyomaték állandó 3)	[Nm/A]
35	áramszabályzó erősítés	[V/A]
36	fordulatszámszabályzó erősítés 3)	[Nm/(rad*s)]
37	Fordulatszám	[ford/perc]
42	teljesítmény	[kW]
43	áram, kicsi	[µA]
46	forgatónyomaték, kicsi 3)	[µNm]
48	ezrelék	-
49	-	[Hz/s]
65	áramlás	[l/perc]
66	nyomás	[bar]
67	térfogat 3)	[cm^3]
68	szakasz erősítés 3)	[mm/(V*perc)]
69	erőszabályzó szakasz erősítés	[N/V]
155	menetemelkedés 3)	[mm/ford], [hüvelyk/ford]
156	menetemelkedés változás 3)	[mm/ford / ford], [hüvelyk/ford / ford]
1)A fizikai egységek függnek a tengelytípustól: lineáris- vagy körtengely		
2) mértékrendszer átkapcsolás G70/G71(hüvelyk/metrikus) Az alaprendszer átkapcsolása után (MD10240 $MN_SCALING_SYSTEM_IS_METRIC) G70/G71-gyel az író/olvasó hozzáférésnél a hosszúság vonatkozású rendszer- és felhasználói változókra **nem** lesz érték-átszámítás (valósérték, alapérték és határértékek) G700/G710(hüvelyk/metrikus) Az alaprendszer átkapcsolása után (MD10240 $MN_SCALING_SYSTEM_IS_METRIC) G700/G710-gyel az író/olvasó hozzáférésnél a hosszúság vonatkozású rendszer- és felhasználói változókra **nem** lesz érték-átszámítás (valósérték, alapérték és határértékek)		
3) A változó **nem** lesz automatikusan átszámítva az NC aktuális mértékrendszerébe (hüvelyk/metrikus). Az átszámítás kizárólag a felhasználó / gépgyártó felelőssége.		

Megjegyzés

Szint túlcsordulás formátum-számítással

Az összes felhasználói változó (GUD / PUD / LUD) belső tárolási formátuma a hossz-vonatkozású egységeknél metrikus. Az ilyen változók intenzív használata az NCK főfutásában, pl. szinkronakciókban, a mértékrendszer átkapcsolásánál számítási idő túlcsordulást okozhat az interpolátor-szinten, vészjelzés 4240.

Megjegyzés

Egységek kompatibilitása

A változók használatánál (hozzárendelés, összehasonlítás, kiszámítás, stb.) az érintett egységek kompatibilitása nem lesz megvizsgálva. Az esetleg szükséges átszámítás kizárólag a felhasználó / gépgyártó felelőssége.

Lásd még

Változók (Oldal 21)

2.1.9 Tulajdonság: Hozzáférési jog (APR, APW, APRP, APWP, APRB, APWB)

Megnevezés

Az `AP`... hozzáférés jelölés a következőkből áll össze:

1. A: **A**ccess
2. P: **P**rotection
3. R / W: **R**ead / **W**rite
4. P / B: **P**rogram / **B**TSS

Hozzáférési jogok / védelmi fokozatok:

A hozzáférési jognak a programozásnál megadandó következő védelmi fokok felelnek meg:

Hozzáférési jog	Védelmi fokozat
rendszer jelszó	0
gépgyártó jelszó	1
szerviz jelszó	2
vég-felhasználó jelszó	3
kulcsos-kapcsoló 3	4
kulcsos-kapcsoló 2	5
kulcsos-kapcsoló 1	6
kulcsos-kapcsoló 0	7

Felhasználói adatok definíciója (`DEF`)

Hozzáférési jogokat (`APR`... / `APW`...) a következő fájlokra lehet definiálni:

- globális felhasználói adatok (GUD)

Rendszeradatok és felhasználói adatok újra definíciója (REDEF)

Hozzáférési jogokat (APR... / APW...) a következő adatokra lehet definiálni:

- rendszeradatok
  - gépadatok

    Megjegyzés

    Gépadatok olvasási jogainak újra definiálása

    A gépadatok olvasása védelmi fokozatot csak a APR kulcsszóval a munkadarabprogramra és a BTSS-re együtt lehet beállítani.

    Az APRP és APRB kulcsszavak az olvasási jogok újra definiálásánál nem támogatottak és a 12490 "APRP/APRB hozzáférési jog <védelmi fok> nem lett beállítva" vészjelzést okozzák.

  - beállítási-adatok
  - rendszerváltozó
  - folyamat adatok
  - táradatok
  - szerszámadatok
- felhasználói adatok
  - R-paraméterek
  - szinkronizációs változók ($AC_MARKER, $AC_PARAM, $AC_TIMER)
  - szinkronakciók–GUD (SYG_xy[ ], x=R, I, B, A, C, S és y=S, M, U, 4, ..., 9)
  - EPS-paraméterek
  - OEM szerszámadatok
  - OEM táradatok
  - globális felhasználói változók (GUD)

    Megjegyzés

    Az újra definiálásnál egy változó védelmi fokozatát a legalacsonyabb védelmi fokozat 7 és a saját védelmi fokozat, pl. 1 (gépgyártó) között szabadon lehet megadni.

NC nyelvi elemek újra definíciója (REDEF)

A hozzáférési és végrehajtási jogot (APX) a következő NC utasításokra lehet újra definiálni::

- G utasítások / útfeltételek
  Irodalom
  Programozási kézikönyv, Alapok; Fejezet: G utasítások / útfeltételek
- Előre definiált funkciók
  Irodalom
  Programozási kézikönyv, Alapok; Fejezet: Előre definiált funkciók

- Előre definiált alprogram-hívások
  Irodalom
  Programozási kézikönyv, Alapok; Fejezet: Előre definiált alprogram-hívások
- `DO` utasítás szinkronakcióknál
- Ciklusok programjelölői
  A ciklusnak egy ciklus-könyvtárban kell lenni és egy PROC utasítást tartalmazni.

NC programok és ciklusok hozzáférési jogai (`APRP`, `APWP`)

A különböző hozzáférési jogoknak egy NC programból ill. ciklusból a hozzáférésre következő hatásuk van:

- `APRP 0` / `APWP 0`
  - Az NC program végrehajtásánál a rendszer jelszó be kell legyen állítva
  - A ciklusnak az _N_CST_DIR (rendszer) könyvtárban kell lennie
  - Az _N_CST_DIR könyvtárban az MD11160 $MN_ACCESS_EXEC_CST-ben a végrehajtási jog rendszerre kell legyen állítva
- `APRP 1` / `APWP 1` ill. `APRP 2` / `APWP 2`
  - Az NC program végrehajtásánál a gépgyártó ill. szerviz jelszó be kell legyen állítva
  - A ciklusnak az _N_CMA_DIR (gépgyártó) vagy az _N_CST_DIR könyvtárban kell lennie
  - Az _N_CMA_DIR ill. _N_CST_DIR könyvtárakra az MD11161 $MN_ACCESS_EXEC_CMA ill. MD11160 $MN_ACCESS_EXEC_CST gépadatban a végrehajtási jog legalább gépgyártóra kell legyen állítva
- `APRP 3` / `APWP 3`
  - Az NC program végrehajtásánál a végfelhasználó jelszó be kell legyen állítva
  - A ciklusnak az _N_CUS_DIR (felhasználó), az _N_CMA_DIR vagy az _N_CST_DIR könyvtárban kell lennie
  - Az _N_CUS_DIR, _N_CMA_DIR ill. _N_CST_DIR könyvtárakra az MD11162 $MN_ACCESS_EXEC_CUS, MD11161 $MN_ACCESS_EXEC_CMA ill. MD11160 $MN_ACCESS_EXEC_CST gépadatban a végrehajtási jog legalább végfelhasználóra kell legyen állítva
- `APRP 4...7` / `APWP 4...7`
  - Az NC program végrehajtásánál a 3 ... 0 kulcsoskapcsoló állás be kell legyen állítva
  - A ciklusnak az _N_CUS_DIR, _N_CMA_DIR vagy _N_CST_DIR könyvtárban kell lennie
  - Az _N_CUS_DIR, _N_CMA_DIR ill. _N_CST_DIR könyvtárakra az MD11162 $MN_ACCESS_EXEC_CUS, MD11161 $MN_ACCESS_EXEC_CMA ill. MD11160 $MN_ACCESS_EXEC_CST gépadatban a végrehajtási jog legalább a megfelelő kulcsos-kapcsoló állásra kell legyen állítva

BTSS hozzáférési jogai (APRP, APWP)

A hozzáférési jogok (APRB, APWB) a BTSS-en az összes komponensre (HMI, PLC, külső számítógép, EPS szolgálatok, stb) a hozzáférést a rendszerváltozókhoz és a felhasználói változókhoz azonos mértékben korlátozzák.

Megjegyzés

HMI lokális hozzáférési jogok

A rendszeradatok hozzáférési jogainak a változtatásánál ügyelni kell arra, hogy ezek a HMI mechanizmusokkal megadott hozzáférési jogokkal konzisztensen történjenek.

Hozzáférési jogok APR / APW

Kompatibilitási okokból a APR és APW tulajdonságok közvetve le lesznek képezve az APRP / APRB és APWP / APWB tulajdonságokra:

- APR x ⇒ APRP x APRB x
- APW y ⇒ APWP y APWB y

Hozzáférési jogok ACCESS fájlokkal

Az ACCESS fájlok alkalmazásánál a hozzáférési jogok megadására a rendszeradatok, felhasználói adatok és NC utasítások hozzáférési jogainak újra definiálását csak ezekben az ACCESS fájlokban szabad programozni. Kivételt képeznek a globális felhasználói adatok (GUD). Ezekre a hozzáférési jogok újra definiálását továbbra is a megfelelő *_DEF definíciós fájlokban kell programozni.

Egy következetes hozzáférési védelemhez a hozzáférési jogok gépadatait és a megfelelő könyvtárak hozzáférés védelmét következetesen kell illeszteni.

Ebből a következő elvi eljárás adódik:

1. A szükséges definíciós fájlok létrehozása:
   - _N_DEF_DIR/_N_SACCESS_DEF
   - _N_DEF_DIR/_N_MACCESS_DEF
   - _N_DEF_DIR/_N_UACCESS_DEF
2. A definíciós fájlok írási jogainak paraméterezése az újra definiáláshoz szükséges értékre:
   - MD11170 $MN_ACCESS_WRITE_SACCESS = <védelmi fok>
   - MD11171 $MN_ACCESS_WRITE_MACCESS = <védelmi fok>
   - MD11172 $MN_ACCESS_WRITE_UACCESS = <védelmi fok>

3. A védett elemekhez hozzáférésre a ciklusokból a _N_CST_DIR, _N_CMA_DIR és _N_CST_DIR ciklus-könyvtárak végrehajtási és írási jogait illeszteni kell.
   Végrehajtási jogok
   - MD11160 $MN_ACCESS_EXEC_CST = <védelmi fok>
   - MD11161 $MN_ACCESS_EXEC_CMA = <védelmi fok>
   - MD11162 $MN_ACCESS_EXEC_CUS = <védelmi fok>

   Írási jogok
   - MD11165 $MN_ACCESS_WRITE_CST = <védelmi fok>
   - MD11166 $MN_ACCESS_WRITE_CMA = <védelmi fok>
   - MD11167 MN_ACCESS_WRITE_CUS = <védelmi fok>

   A végrehajtási fokot legalább a használt elemek legmagasabb védelmi fokával azonos védelmi fokra kell állítani.
   Az írási jog legalább a végrehajtási joggal azonos védelmi fokra kell legyen állítva.
4. A HMI lokális ciklus-könyvtárak védelmi jogait az NC lokális ciklus-könyvtárakkal azonos védelmi fokra kell állítani.
   Irodalom
   Kezelési kézikönyv

Alprogram-hívások ACCESS fájlokban

A hozzáférési jogok további strukturálásához az ACCESS fájlokban lehet alprogramokat (SPF vagy MPF jelölő) is felhívni. Az alprogramok ekkor öröklik a felhívó ACCESS fájl végrehajtási jogait.

Megjegyzés

Az ACCESS fájlokban csak a hozzáférési jogokat lehet újra definiálni. A többi tulajdonságokat továbbra is a megfelelő definíciós fájlokban lehet programozni ill. újra definiálni.

Lásd még

Változók (Oldal 21)

2.1.10 Tulajdonság: Adat-osztály (DCM, DCI, DCU) - csak SINUMERIK 828D

Az adatkezelés egyszerűsítésére az üzembehelyezés, sorozat-üzembehelyezés és a gépek ill. gépsorok felújítása keretében az NC-ben az összes rendszer és felhasználói adat adat-osztályokba van felosztva.

Adat-osztály	Adatok
S = System	A Siemens által rendelkezésre bocsájtott rendszeradatok, mint gép- és beállítási-adatok, alap-beállítások, mérőciklusok, definíciók (SGUD) és makrók (SMAC) stb.
M = Manufacturer (gépgyártó)	Gépsor-specifikus üzembehelyezési adatok, mint gyártói ciklusok, definíciók (MGUD) és makrók (MMAC) és gépadatok, amelyek a gép működési terjedelmét megadják.
I = Individual (gép-specifikus)	Gép-specifikus üzembehelyezési adatok, mint kompenzációs adatok és referenciapont eltolások.
U = User (felhasználó)	Gép-specifikus adatok, amelyek a gép üzemelése során keletkeznek, mint szerszámadatok, beállítási-adatok, alprogramok, felhasználói ciklusok, definíciók (UGUD) und makrók (UMAC).

Irodalom:
SINUMERIK 828D Üzembehelyezési kézikönyv Esztergálás és marás; fejezet: "Bevezetés és adat-osztályok alkalmazása"

Felhasználói adatok definíciója (DEF)

A fájl vagy könyvtár adat-osztályával, amelyben a felhasználói adat definiálva van, közvetetten az adat adat-osztálya meg van adva. Az adat adat-osztályát nem lehet megváltoztatni.

A felhasználói adat definíciójánál (`DEF`) az **adatértékre** egy saját, az adat adat-osztályától eltérő adat-osztályt lehet előre megadni.

Az adatérték adat-osztályára érvényes kell legyen:

adatérték adat-osztályának prioritása ≤ adat adat-osztályának prioritása

Példa:

A GUD definíciója, ami egy mérőtapintót leír, abban az M (= Manufacturer) adat-osztályban kell legyen, mivel az a gyártói ciklusok ciklusok lefutásához szükséges. Az adat értéke azonban, mivel a mérőtapintó típusa gépről gépre eltérő lehet, az I (= Individual) adat-osztályhoz kell tartozzon.

MGUD.DEF (M adat-osztály)

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

Rendszeradatok újra definíciója (REDEF)

A rendszeradatok adat-osztályát újra definícióval (`REDEF`) meg lehet változtatni. Az újra definíció egy S vagy M osztályú definíciós fájlban kell történjen.

Az ACCESS fájlok használatánál az újra definíciónak az ACCESS fájlban kell történni.

A gép-, beállítási és opciós adatok valamint rendszerváltozók mindenkori adatosztálya a következő helyen található:

- Részletes gépadat leírás Jegyzékes kézikönyv, paraméter: "osztály"
- Jegyzékes kézikönyv Rendszerváltozók

2.1.11 Definiálható és újra definiálható tulajdonságok áttekintése

A következő táblázat mutatja, hogy melyik adatfajtáknál melyik tulajdonságokat lehet definiálni (`DEF`) és/vagy újra definiálni (`REDEF`).

Rendszeradatok

Adatfajta	Inic_érték	Határértékek	Fiz. egység	Hozzáférési jogok	Adat-osztály (csak 828D)
gépadatok	---	---	---	REDEF	REDEF
beállítási-adatok	REDEF	---	---	REDEF	---
FRAME adatok	---	---	---	REDEF	---
folyamat adatok	---	---	---	REDEF	---
orsó-emelkedési hiba-komp. (EEC)	---	---	---	REDEF	---
belógás kompenzáció (CEC)	---	---	---	REDEF	---
negyed-hiba kompenzáció (QEC)	---	---	---	REDEF	---
táradatok	---	---	---	REDEF	---
szerszámadatok	---	---	---	REDEF	---
védelmi tartományok	---	---	---	REDEF	---
tájolható szerszámtartó	---	---	---	REDEF	---
kinemtikus láncok	---	---	---	REDEF	---
3D-s védelmi tartományok	---	---	---	REDEF	---
munkatér-határolás	---	---	---	REDEF	---

Felhasználói adatok

Adatfajta	Inic_érték	Határértékek	Fiz. egység	Hozzáférési jogok	Adat-osztály
R-paraméterek	REDEF	REDEF	REDEF	REDEF	---
szinkronakció változók ($AC_...)	REDEF	REDEF	REDEF	REDEF	---
szinkronakció GUD (SYG_...)	REDEF	REDEF	REDEF	REDEF	---
EPS-paraméterek	REDEF	REDEF	REDEF	REDEF	---
OEM szerszámadatok	REDEF	REDEF	REDEF	REDEF	---
OEM táradatok	REDEF	REDEF	REDEF	REDEF	---
globális felhasználói változók (GUD)	DEF / REDEF	DEF	DEF	DEF / REDEF	DEF / REDEF
lokális felhasználói változók (PUD / LUD)	DEF	DEF	DEF	---	---

Lásd még

Változók (Oldal 21)

2.1.12 Mező-változók (DEF, SET, REP) definíciója és inicializálása

Egy felhasználói változót 1-től maximum 3-dimenziós mezőként (array) lehet definiálni:

- 1-dimenziós: `DEF <adattípus> <változónév>[<n>]`
- 2-dimenziós: `DEF <adattípus> <változónév>[<n>,<m>]`
- 3-dimenziós: `DEF <adattípus> <változónév>[<n>,<m>,<o>]`

Megjegyzés

A STRING típusú felhasználói változókat maximum 2-dimenziós mezőként lehet definiálni.

Adattípusok

Felhasználói változókat mezőkként a következő adattípusokkal lehet definiálni: BOOL, CHAR, INT, REAL, STRING, AXIS, FRAME

Érték-hozzárendelés mezőelemekhez

Az érték-hozzárendelések a mezőelemekhez a következő időpontokban történhetnek:

- mezők definíciójánál (inicializálási értékek)
- programfutás közben

Érték-hozzárendelés történhet:

- egy mezőelem közvetlen megadásával
- egy mezőelem közvetlen megadásával kezdőelemként és egy értéklista megadásával (`SET`)
- egy mezőelem közvetlen megadásával kezdőelemként és egy érték és az ismétlések számának megadásával (`REP`)

Megjegyzés

A FRAME adattípusú felhasználói változókhoz nem lehet inicializálási értékeket hozzárendelni.

Szintakszis (DEF)

```
DEF <adattípus> <változónév>[<n>,<m>,<o>]
DEF STRING [<string hossz>] <változónév>[<n>,<m>]
```

Szintakszis (DEF...=SET...)

Egy értéklista használata:

- a definíciónál:
  ```
  DEF <adattípus> <változónév>[<n>,<m>,<o>]=
  SET(<érték1>,<érték2>,...)
  ```
  azonos jelentésű:
  ```
  DEF <adattípus> <változónév>[<n>,<m>,<o>]= (<érték1>,<érték2>,...)
  ```

  Megjegyzés

  Az inicializálásnál értéklistával a `SET` megadása opcionális.

- egy érték-hozzárendelésnél:
  ```
  <változónév>[<n>,<m>,<o>]= SET(<érték1>,<érték2>,...)
  ```

Szintakszis (DEF...=REP...)

Egy érték használata ismétléssel

- a definíciónál:
  ```
  DEF <adattípus> <változónév>[<n>,<m>,<o>]= REP(<érték>)
  DEF <adattípus> <változónév>[<n>,<m>,<o>]=
  REP(<érték>,<mezőelemek_száma>)
  ```
- egy érték-hozzárendelésnél:
  ```
  <változónév>[<n>,<m>,<o>]= REP(<érték>)
  <változónév>[<n>,<m>,<o>]= REP(<érték>,<mezőelemek_száma>)
  ```

Jelentés

`DEF`:	utasítás változók definíciójához	
`<adattípus>`:	változók adattípusa	
	értéktartomány: • rendszerváltozóknál: BOOL, CHAR, INT, REAL, STRING, AXIS • GUD vagy LUD változóknál: BOOL, CHAR, INT, REAL, STRING, AXIS, FRAME	
`<string_hossz>`:	karakterek maximális száma STRING adattípusnál	
`<változó_név>`:	változó-név	
`[<n>,<m>,<o>]`:	mezőméretek ill. mezőindexek	
`<n>`:	mezőméret ill. mezőindex az 1. dimenzióhoz	
	típus:	INT (rendszerváltozóknál AXIS is)
	értéktartomány:	max. mező-méret: 65535 mező-index: 0 ≤ n ≤ 65534
`<m>`:	mezőméret ill. mezőindex az 2. dimenzióhoz	
	típus:	INT (rendszerváltozóknál AXIS is)
	értéktartomány:	max. mező-méret: 65535 mező-index: 0 ≤ m ≤ 65534

<o>:	mezőméret ill. mezőindex az 3. dimenzióhoz	
	típus:	INT (rendszerváltozóknál AXIS is)
	értéktartomány:	max. mező-méret: 65535 mező-index: 0 ≤ o ≤ 65534
SET:	érték-hozzárendelés a megadott értéklistával	
(<érték1>,<érték2>,...):	értéklista	
REP:	érték-hozzárendelés a megadott <érték>-kel	
<érték>:	érték, amivel a mezőelemek a REP-pel való inicializálásnál írva lesznek.	
<mezőelemek_száma>:	mezőelemek száma, amelyek a megadott <érték>-kel írva lesznek. A maradék mezőelemekre érvényes az időponttól függően: • inicializálás a mező definíciónál: → a maradék mezőelemek nullával lesznek írva • hozzárendelés programfutás közben: → a mezőelemek aktuális értékei változatlanok maradnak Ha a paraméter nincs programozva, az összes mezőelem az <érték>-kel lesz írva. Ha a paraméter nulla, az időponttól függően érvényes: • inicializálás a mező definíciónál: → összes elem nullával lesz feltöltve • hozzárendelés programfutás közben: → a mezőelemek aktuális értékei változatlanok maradnak	

Mező-index

A mezőelemek közvetett sorrendje pl. egy SET vagy REP érték-hozzárendelésnél a mezőindexek szerint iterálva jobbról balra történik.

Példa: Egy 3-dimenziós mező inicializálása 24 mezőelemmel:

```
DEF INT FELD[2,3,4] = REP(1,24)
  FELD[0,0,0] = 1      1. mezőelem
  FELD[0,0,1] = 1      2. mezőelem
  FELD[0,0,2] = 1      3. mezőelem
  FELD[0,0,3] = 1      4. mezőelem
  ...
  FELD[0,1,0] = 1      5. mezőelem
  FELD[0,1,1] = 1      6. mezőelem
  ...
  FELD[0,2,3] = 1      12. mezőelem
  FELD[1,0,0] = 1      13. mezőelem
  FELD[1,0,1] = 1      14. mezőelem
  ...
  FELD[1,2,3] = 1      24. mezőelem
```

megfelelően:

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

Példa: Teljes változómezők inicializálása

Aktuális kitöltést lásd az ábrában.

Programkód

```
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)
```

mező-index 2

[1,2]	N10: Inicializálás a definíciónál			N20/N30: inicializálás azonos értékekkel			N40/N50: inicializálás különböző értékekkel		
	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
	A [5,0] ... [9,2] mezőelemek az alapértékkel (0.0) lettek inicializálva.						A [3,1] ... [4,0] mezőelemek az alapértékkel (0.0) lettek inicializálva. A [6,0] ... [8,0] mezőelemek nem lettek megváltoztatva.		

1

Lásd még

Mező-változók (DEF, SET, REP) definíciója és inicializálása: További információk (Oldal 56)

Változók (Oldal 21)

2.1.13 Mező-változók (DEF, SET, REP) definíciója és inicializálása: További információk

További információk (SET)

Inicializálás a definíciónál

- Annyi mezőelem lesz inicializálva az 1. mezőelemtől kezdődően az értéklista elemeivel, amennyi elem az értéklistában programozva van
- Az érték megadás nélküli mezőelemek (hézagok az értéklistákban) 0-val lesznek feltöltve.
- Az AXIS típusú változóknál a hézagok az értéklistákban nem megengedettek.
- Ha az értéklista több értéket tartalmaz, mint amennyi mezőelem definiálva van, egy vészjelzés lesz kijelezve.

Érték-hozzárendelés a programfutásban

Az érték-hozzárendelés a programfutásban a definícióknál előbb leírt szabályok érvényesek. Továbbá a következő lehetőségek vannak:

- Az értéklista elemeiként kifejezések is megengedettek.
- Az érték-hozzárendelés a programozott mezőindexnél kezdődik. Ezzel lehetséges rész-mezőket célzottan értékekkel ellátni.

Példa:

Programkód	Kommentár
`DEF INT FELD[5,5]`	`; mező definíció`
`FELD[0,0]=SET(1,2,3,4,5)`	`; érték-hozzárendelés az első 5 mezőelemhez [0,0] - [0,4]`
`FELD[0,0]=SET(1,2, , ,5)`	`; érték-hozzárendelés az első 5 mezőelemhez [0,0] - [0,4], mezőelem [0,2] és [0,3] = 0`
`FELD[2,3]=SET(VARIABLE,4*5.6)`	`; érték-hozzárendelés változóval és kifejezés-sel a [2,3] mezőindextől: [2,3] = VARIABLE [2,4] = 4 * 5.6 = 22.4`

További információk (REP)

Inicializálás a definíciónál

- Az összes vagy az opcionálisan megadott számú mezőelem a megadott értékkel (állandó) lesz inicializálva.
- FRAME adattípusú változókat nem lehet inicializálni.

Példa:

Programkód	Kommentár
DEF REAL varName[10]=REP(3.5,4)	; mező definíció és [0] ... [3] mezőelemek inicializálása 3,5 értékkel

Érték-hozzárendelés a programfutásban

Az érték-hozzárendelés a programfutásban a definícióknál előbb leírt szabályok érvényesek. Továbbá a következő lehetőségek vannak:

- Az értéklista elemeiként kifejezések is megengedettek.
- Az érték-hozzárendelés a programozott mezőindexnél kezdődik. Ezzel lehetséges részmezőket célzottan értékekkel ellátni.

Példák:

Programkód	Kommentár
DEF REAL varName[10]	; mező definíció
varName[5]=REP(4.5,3)	; [5] ... [7] mezőelemek = 4,5
R10=REP(2.4,3)	; R10 ... R12 R-paraméterek = 2,4
DEF FRAME FRM[10]	; mező definíció
FRM[5]=REP(CTRANS(X,5))	; [5] ... [9] mezőelemek = CTRANS(X,5)

Lásd még

Mező-változók (DEF, SET, REP) definíciója és inicializálása (Oldal 52)

2.1.14 Adattípusok

A következő adattípusok állnak rendelkezésre az NC-ben:

Adattípus	Jelentés	Értéktartomány
INT	egészszám előjellel	-2147483648 ... +2147483647
REAL	valósszám (LONG REAL IEEE szerint)	±(~2,2*10^{-308} … ~1,8*10^{+308})
BOOL	igazságérték IGAZ (1) és HAMIS (0)	1, 0
CHAR	ASCII karakter	ASCII kód 0 ... 255
STRING	adott hosszú karakterlánc	maximum 200 karakter (nem különleges karakter)
AXIS	tengely-/orsó-jelölő	csatornajelölő
FRAME	geometriai adatok egy statikus koordinátatranszformációhoz (eltolás, forgatás, skálázás, tükrözés)	---

Közvetett adattípus átalakítások

A következő adattípus átalakítások lehetségesek és hozzárendelésnél és paraméter-átadásnál közvetetten lesznek végrehajtva:

ből↓/ be →	REAL	INT	BOOL
REAL	x	o	&
INT	x	x	&
BOOL	x	x	x
x: korlátozás nélkül lehetséges o: adatvesztés lehetséges az értéktartomány túllépése miatt ⇒ vészjelzés; kerekítés: tizedespont utáni érték ≥ 0,5 ⇒ felkerekíteni, tizedespont utáni érték < 0,5 ⇒ lekerekíteni &: érték≠ 0 ⇒ IGAZ, érték== 0 ⇒ HAMIS			

Lásd még

Változók (Oldal 21)

2.1.15 Egy változó előfordulását megvizsgálni (ISVAR)

Az ISVAR előre definiált funkcióval meg lehet vizsgálni, hogy egy rendszer-/felhasználói változó (pl. gépadat, beállítási adat, rendszerváltozó, általános változó mint GUD) ismert-e az NC-ben.

Változó

A lekérdezendő változó következőképpen legyen felépítve:

dimenzió nélküli változó:	<változó>
egy-dimenziós változó mező-index nélkül:	<változó>[ ]
egy-dimenziós változó n mező-indexszel:	<változó>[<n>]
két-dimenziós változó mező-index nélkül:	<változó>[,]
két-dimenziós változó n és m mező-indexekkel:	<változó>[<n>,<m>]

Szintaxis

```
<eredmény>=ISVAR(<változó>[<n>,<m>])
```

Jelentés

<eredmény>:	visszaadási érték		
	adattípus:	BOOL	
	értéktartomány:	1	változó létezik
		0	változó ismeretlen
ISVAR:	Megvizsgálja, hogy a megadott rendszer / felhasználói változó az NC-ben ismert-e.		

<változó>:	rendszer- /felhasználói változó neve	
	adattípus:	STRING
<n>:	első dimenzió mező-indexe (opcionális)	
	adattípus:	INT
<m>:	második dimenzió mező-indexe (opcionális)	
	adattípus:	INT

Az átadási paraméternek megfelelően a következő vizsgálatok lesznek végrehajtva:

- ismert-e a név
- a változó egy mező-e
- egy- vagy kétdimenziós mezőről van-e szó
- mindenkori mező-index a megengedett tartományban van-e

Ha az összes vizsgálat pozitív, TRUE (1) lesz visszaadva.

Ha a vizsgálat negatív vagy egy szintakszis lépett fel, FALSE (0) kerül visszaadásra.

Példák

Programkód	Kommentár
DEF INT VAR1	
DEF BOOL IS_VAR=FALSE	
N10 IS_VAR=ISVAR("VAR1")	; IS_VAR ebben az esetben TRUE.

Programkód	Kommentár
DEF REAL VARARRAY[10,10]	
DEF BOOL IS_VAR=FALSE	
N10 IS_VAR=ISVAR("VARARRAY[,]")	; IS_VAR ebben az esetben TRUE, egy két-dimenziós mező.
N20 IS_VAR=ISVAR("VARARRAY")	; IS_VAR TRUE, változó létezik.
N30 IS_VAR=ISVAR("VARARRAY[8,11]")	; IS_VAR FALSE, mezőindex nem megengedett.
N40 IS_VAR=ISVAR("VARARRAY[8,8")	; IS_VAR FALSE, "]" hiányzik (szintaktikai hiba).
N50 IS_VAR=ISVAR("VARARRAY[,8]")	; IS_VAR TRUE, mező-index megengedett.
N60 IS_VAR=ISVAR("VARARRAY[8,]")	; IS_VAR TRUE, mező-index megengedett.

Programkód	Kommentár
DEF BOOL IS_VAR=FALSE	
N100 IS_VAR=ISVAR("$MC_GCODE_RESET_VALUES[1]"	; Átadási paraméter egy gépadat, IS_VAR itt TRUE.

Programkód	Kommentár
DEF BOOL IS_VAR=FALSE	
N10 IS_VAR=ISVAR("$P_EP")	; IS_VAR ebben az esetben TRUE.

Programkód	Kommentár
N20 IS_VAR=ISVAR("$P_EP[X]")	; IS_VAR ebben az esetben TRUE.

2.1.16 Attribútum érték / adattípus olvasása (GETVARPHU, GETVARAP, GETVARLIM, GETVARDIM, GETVARDFT, GETVARTYP)

A GETVARPHU, GETVARAP, GETVARLIM és GETVARDFT előre definiált funkciókkal lehet olvasni a rendszer- /felhasználói változók attribútum értékeit, a GETVARTYP-pel egy rendszer- /felhasználói változó típusát

Fizikai egységet olvasni

Szintaxis:
`<eredmény>=GETVARPHU("<név>")`

Jelentés:

<eredmény>:	fizikai egység számértéke		
	adattípus:	INT	
	értéktartomány:	lásd a "Tulajdonság: Fizikai egység (PHU) (Oldal 43)" táblázatban	
		Hiba esetén:	
		- 2	A megadott <név> nincs egy rendszerváltozóhoz vagy felhasználói változóhoz rendelve.
GETVARPHU:	egy rendszer- /felhasználói változó fizikai egységének olvasása		
<név>:	rendszer- /felhasználói változó neve		
	adattípus:	STRING	

Példa:

Az NC a következő GUD változókat tartalmazza:

DEF CHAN REAL PHU 42 LLI 0 ULI 10000 electric

Programkód	Kommentár
DEF INT result=0	
result=GETVARPHU("electric")	; GUD változó fizikai egységének megállapítása.
IF (result < 0) GOTOF error	

Eredményként a 42 érték lett visszaadva. Ez megfelel a [kW] fizikai egységnek.

Megjegyzés

A GETVARPHU-val pl. meg lehet vizsgálni, hogy egy a = b érték-hozzárendelés mindkét változójának az elvárt fizikai egysége van-e.

Olvasás hozzáférési jog

Szintaxis:
`<eredmény>=GETVARAP(<név>,<hozzáférés>)`

Jelentés:

`<eredmény>`:	védelmi fokozat a megadott <hozzáférés>-hez		
	adattípus:	INT	
	értéktartomány:	0 ... 7	Lásd "Tulajdonság: Hozzáférési jog (APR, APW, APRP, APWP, APRB, APWB) (Oldal 45)".
		Hiba esetén:	
		- 1	nem írható (csak <hozzáférés> "`WP`" és "`WB`" releváns)
		- 2	A megadott `<név>` nincs egy rendszerváltozóhoz vagy felhasználói változóhoz rendelve.
		- 3	helytelen érték a `<hozzáférés>`-re
`GETVARAP`:	egy rendszer- /felhasználói változó hozzáférési jogának olvasása		
`<név>`:	rendszer- /felhasználói változó neve		
	adattípus:	STRING	
`<hozzáférés>`:	hozzáférés fajtája		
	adattípus:	STRING	
	értéktartomány:	`"RP"`	olvasás munkadarabprogrammal
		`"WP"`	írás munkadarabprogrammal
		`"RB"`	olvasás BTSS által
		`"WB"`	írás BTSS által

Példa:

Programkód	Kommentár
`DEF INT result=0`	
`result=GETVA-RAP("$TC_MAP8","WB")`	`; Állapítsa meg a "Magazinposition" rendszer-paramé-ter hozzáférési védelmét a BTSS írás vonatkozásában.`
`IF (result < 0) GOTOF error`	

Eredményként a 7 érték lett visszaadva. Ez megfelel a 0-s kulcsoskapcsoló-állásnak (= nincs hozzáférés védelem)

Megjegyzés

A GETVARAP-pal pl lehet készíteni egy vizsgáló programot, amely megvizsgálja az alkalmazástól elvárt hozzáférési jogokat.

Határértékek olvasása

Szintaxis:
`<állapot>=GETVARLIM(<név>,<határérték>,<eredmény>)`

Jelentés:

<állapot>:	funkció állapot		
	adattípus:	INT	
	értéktartomány:	1	o. k.
		-1	nincs határérték definiálva (AXIS, STRING, FRAME típusú változóknál)
		-2	A megadott <név> nincs egy rendszerváltozóhoz vagy felhasználói változóhoz rendelve.
		-3	helytelen érték a <határérték>-re
GETVARLIM:	egy rendszer- /felhasználói változó alsó/felső határértékének olvasása		
<név>:	rendszer- /felhasználói változó neve		
	adattípus:	STRING	
<határérték>:	Megadja, hogy melyik határértéket kell kiolvasni.		
	adattípus:	CHAR	
	értéktartomány:	"L":	= alsó határérték
		"U":	= felső határérték
<eredmény>:	határérték visszaadása		
	adattípus:	VAR REAL	

Példa:

Programkód	Kommentár
DEF INT state=0	
DEF REAL result=0	
state=GETVARLIM("$MA_MAX_AX_VELO","L",result)	; Állapítsa meg az MD32000 $MA_MAX_AX_VELO alsó határértékét.
IF (result < 0) GOTOF error	

Attribútum/adat-típus olvasás

Szintaxis:

<eredmény>=GETVARDIM(<név>, Index)

Jelentés:

<eredmény>:	mező dimenziója/száma <Index>	
	adattípus:	INT
GETVARDIM:	egy rendszer- /felhasználói változó alsó/felső határértékének olvasása	
<név>:	a mező elemek számának olvasása	
	adattípus:	STRING
<index>:	mező száma, max 3.	
	adattípus:	INT

Példa:

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

Alapértéket olvasni

Szintaxis:

```
<állapot>=GETVARDFT(<név>,<eredmény>[,<Index_1>,<Index_2>,<Index_3>]
)
```

Jelentés:

<állapot>:	funkció állapot		
	adattípus:	INT	
	értéktartomány:	1	o. k.
		-1	nem áll rendelkezésre alapérték (pl. mert <eredmény> típusa nem megfelelő a <név>-hez)
		-2	A megadott <név> nincs egy rendszerváltozóhoz vagy felhasználói változóhoz rendelve.
		-3	helytelen érték <Index_1>-re, Dimenzió egynél kisebb (= nem Array = Skalar)
		-4	helytelen érték a <Index_2>-re
		-5	helytelen érték a <Index_3>-ra
GETVARDFT:	egy rendszer- /felhasználói változó alapértékének olvasása		
<név>:	rendszer- /felhasználói változó neve		
	adattípus:	STRING	
<eredmény>:	alapérték visszaadása		
	adattípus:	VAR REAL (INT, REAL, BOOL, AXIS típusú változók alapértékének olvasásánál)	
		VAR STRING (STRING és CHAR típusú változók alapértékének olvasásánál)	
		VAR FRAME (FRAME típusú változók alapértékének olvasásánál)	
<Index_1>:	index az első dimenzióra (opcionális)		
	adattípus:	INT	
	nincs programozva jelentése = 0		
<Index_2>:	index a második dimenzióra (opcionális)		
	adattípus:	INT	
	nincs programozva jelentése = 0		
<Index_3>:	index a harmadik dimenzióra (opcionális)		
	adattípus:	INT	
	nincs programozva jelentése = 0		

Példa:

Programkód	Kommentár
DEF INT state=0	
DEF REAL resultR=0	; Változó az INT, REAL, BOOL, AXIS típusok alapértékeinek felvételéhez.
DEF FRAME resultF=0	; Változó a FRAME típus alapértékeinek felvételéhez
IF (GETVARTYP("$MA_MAX_AX_VELO") <> 4) GOTOF error	
state=GETVARDFT("$MA_MAX_AX_VELO", resultR, AXTOINT(X))	; Állapítsa meg az "X" tengely alapértékét.
IF (resultR < 0) GOTOF error	
IF (GETVARTYP("$TC_TP8") <> 3) GOTOF error	
state=GETVARDFT("$TC_TP8", resultR)	
IF (GETVARTYP("$P_UBFR") <> 7) GOTOF error	
state=GETVARDFT("$P_UBFR", resultF)	

Adattípus olvasása

Szintaxis:

`<eredmény>=GETVARTYP("<név>")`

Jelentés:

<eredmény>:	megadott rendszer- /felhasználói változó adattípusa		
	adattípus:	INT	
	értéktartomány:	1	= BOOL
		2	= CHAR
		3	= INT
		4	= REAL
		5	= STRING
		6	= AXIS
		7	= FRAME
		Hiba esetén:	
		< 0	A megadott <név> nincs egy rendszerváltozóhoz vagy felhasználói változóhoz rendelve.
GETVARTYP:	egy rendszer- /felhasználói változó adattípusának olvasása		
<név>:	rendszer- /felhasználói változó neve		
	adattípus:	STRING	

Példa:

Programkód	Kommentár
DEF INT result=0	
DEF STRING name="R"	
result=GETVARTYP(name)	; Állapítsa meg az R paraméter típusát.
IF (result < 0) GOTOF error	

Eredményként a 4 érték lett visszaadva. Ez a REAL adattípusnak felel meg.

2.2 Közvetett programozás

2.2.1 Címek közvetett programozása

A címek közvetett programozásánál a bővített cím (<index>) egy megfelelő típusú változóval lesz helyettesítve.

Megjegyzés

A címek közvetett programozása nem lehetséges:

- N (mondatszám)
- L (alprogram)
- beállítható címek
  (pl. X[1] nem megengedett az X1 helyett)

Szintaxis

```
<CÍM>[<index>]
```

Jelentés

`<CÍM>[...]`:	fix cím bővítéssel (index)
`<index>`:	változó pl. orsó-szám, tengely....

Példák

Példa 1: Egy orsó-szám közvetett programozása

Közvetlen programozás:

Programkód	Kommentár
`S1=300`	`; 300 ford/perc fordulatszám az 1-es számú orsónak.`

Közvetett programozás:

Programkód	Kommentár
`DEF INT SPINU=1`	`; INT típusú változó definiálása és érték-hozzárendelés.`
`S[SPINU]=300`	`; 300 ford/perc fordulatszám az orsónak, amelynek a száma a SPINU változóban van (ebben a példában az 1-es számú orsónak).`

Példa 2: Egy tengely közvetett programozása

Közvetlen programozás:

Programkód	Kommentár
`FA[U]=300`	`; Előtolás 300 az "U" tengelyre.`

Közvetett programozás:

Programkód	Kommentár
DEF AXIS AXVAR2=U	; AXIS típusú változó definiálása és érték-hozzárendelés.
FA[AXVAR2]=300	; Előtolás 300 a tengelyre, amelynek tengelyneve az AXVAR2 változóban van.

Példa 3: Egy tengely közvetett programozása

Közvetlen programozás:

Programkód	Kommentár
$AA_MM[X]	; "X" tengely mérőtapintó mérésértékét (GKR) olvasni.

Közvetett programozás:

Programkód	Kommentár
DEF AXIS AXVAR3=X	; AXIS típusú változó definiálása és érték-hozzárendelés.
$AA_MM[AXVAR3]	; Mérőtapintó mérésértékét (GKR) olvasni tengelyre, amelynek a neve az AXVAR3 változóban van.

Példa 4: Egy tengely közvetett programozása

Közvetlen programozás:

Programkód
X1=100 X2=200

Közvetett programozás:

Programkód	Kommentár
DEF AXIS AXVAR1 AXVAR2	; Két AXIS típusú változó definíciója.
AXVAR1=(X1) AXVAR2=(X2)	; Tengelynevek hozzárendelése.
AX[AXVAR1]=100 AX[AXVAR2]=200	; Tengelyek mozgatása, amelyek tengelyneve az AXVAR1 és AXVAR2 nevű változókban van.

Példa 5: Egy tengely közvetett programozása

Közvetlen programozás:

Programkód
G2 X100 I20

Közvetett programozás:

Programkód	Kommentár
DEF AXIS AXVAR1=X	; AXIS típusú változó definiálása és érték-hozzárendelés.
G2 X100 IP[AXVAR1]=20	; Középpont megadás közvetett programozás a tengelyre, amelynek tengelyneve az AXVAR1 nevű változóban van.

Példa 6: Mezőelemek közvetett programozása

Közvetlen programozás:

Programkód	Kommentár
DEF INT FELD1[4,5]	; 1. mező definíciója.

Közvetett programozás:

Programkód	Kommentár
DEFINE DIM1 AS 4	; A mező-dimenzióknál a mezőméreteket fix értékekként kell megadni.
DEFINE DIM2 AS 5	
DEF INT FELD[DIM1,DIM2]	
FELD[DIM1-1,DIM2-1]=5	

Példa 7: Közvetett alprogram hívás

Programkód	Kommentár
CALL "L" << R10	; Program felhívása, amelynek a száma az R10-ben van (string láncolás).

2.2.2 G utasítások közvetett programozása

A G utasítások közvetett programozása hatékony ciklus-programozást tesz lehetővé.

Szintaxis

```
G[<csoport>]=<szám>
```

Jelentés

G[...]:	G utasítás bővítéssel (index)	
<csoport>:	index paraméter G-csoport	
	típus:	INT
<szám>:	változó a G utasítás számhoz	
	típus:	INT vagy REAL

Megjegyzés

Általában csak a nem szintakszis-meghatározó G utasításokat lehet közvetetten programozni.

A szintakszis-meghatározó G utasítások közül csak a G-csoport 1 lehetséges.
A 2, 3 és 4 G-csoport szintakszis-meghatározó G utasításai nem lehetségesek.

Megjegyzés

A közvetett G utasítás programozásban aritmetikai funkciók nem megengedettek. A G utasítás szám szükséges kiszámítása egy külön programsorban kell történjen a közvetett G utasítás programozás előtt.

Példák

Példa 1: Beállítható nullaponteltolás (G-csoport 8)

Programkód	Kommentár
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 utasítás kiszámítás
N1110 G[8]=INT_VAR G1 X0 Y0	; G55

Példa 2: Sík választás (G-csoport 6)

Programkód	Kommentár
N2010 R10=$P_GG[6]	; 6-os G-csoport aktív G utasítását olvasni
...	
N2090 G[6]=R10	

Irodalom

Információk a G-csoportokhoz, lásd:
Programozási kézikönyv, alapok; "G-csoportok" fejezet

2.2.3 Pozíció tulajdonságok (GP) közvetett programozása

A pozíció tulajdonságokat, mint pl. a tengelypozíciók növekményes vagy abszolút programozása a `GP` kulcsszóval változóként lehet közvetetten programozni.

Alkalmazás

A pozíció tulajdonságok közvetett programozását a **helyettesítő ciklusokban** használjuk, mert a következő előnyei vannak a pozíció tulajdonságok kulcsszóként (pl. `IC`, `AC`, ...) való programozásával szemben:

A változóként való közvetett programozás miatt **nem** szükséges `CASE` utasítás, ami az összes lehetséges pozíció tulajdonságra elágazna.

Szintaxis

```
<POZÍCIONÁLÓ_UTASÍTÁS>[<tengely/orsó>]=
GP(<pozíció>,<pozíció_tulajdonság)
[<tengely/orsó>]=GP(<pozíció>,<pozíció_tulajdonság)
```

Jelentés

`<POZÍCIONÁLÓ_UTASÍTÁS>[]`:	A következő pozícionáló utasításokat lehet a GP kulcsszóval együtt programozni: `POS`, `POSA`, `SPOS`, `SPOSA` Ezen kívül lehetséges: • a csatornában előforduló összes tengely-/orsó-jelölő: `<tengely/orsó>` • változó tengely-/orsó-jelölő `AX`
`<tengely/orsó>`:	engely/orsó, amit pozícionálni kell.
`GP()`:	kulcsszó pozícionáláshoz
`<pozíció>`:	paraméter 1 tengely-/orsópozíció állandóként vagy változóként
`<pozíció_tulajdonság>`:	paraméter 2 pozíció tulajdonság(pl. pozícióra menet modus) változóként (pl. $P_SUB_SPOSMODE) vagy kulcsszóként (IC, AC, ...)

A változó által adott értékek jelentése a következő:

Érték	Jelentés	Megengedett:
0	Nincs pozíció tulajdonság változás	
1	AC	`POS`, `POSA`, `SPOS`, `SPOSA`, `AX`, tengelycím
2	IC	`POS`, `POSA`, `SPOS`, `SPOSA`, `AX`, tengelycím
3	DC	`POS`, `POSA`, `SPOS`, `SPOSA`, `AX`, tengelycím
4	ACP	`POS`, `POSA`, `SPOS`, `SPOSA`, `AX`, tengelycím
5	ACN	`POS`, `POSA`, `SPOS`, `SPOSA`, `AX`, tengelycím
6	OC	-
7	PC	-
8	DAC	`POS`, `POSA`, `AX`, tengelycím
9	DIC	`POS`, `POSA`, `AX`, tengelycím
10	RAC	`POS`, `POSA`, `AX`, tengelycím
11	RIC	`POS`, `POSA`, `AX`, tengelycím
12	CAC	`POS`, `POSA`
13	CIC	`POS`, `POSA`
14	CDC	`POS`, `POSA`
15	CACP	`POS`, `POSA`
16	CACN	`POS`, `POSA`

Példa

Egy aktív szinkronorsó-csatolásnál az S1 vezetőorsó és az S2 követőorsó között az `SPOS` utasítással a főprogramban a következő helyettesítő-ciklus lesz felhívva az orsó pozícionálásához.

A pozícionálás az `N2230`-ban levő utasítással történik:
`SPOS[1]=GP($P_SUB_SPOSIT,$P_SUB_SPOSMODE) SPOS[2]=GP($P_SUB_SPOSIT, $P_SUB_SPOSMODE)`

Az elérendő pozíció az $P_SUB_SPOSIT rendszerváltozóból, a pozícióra menet modus a $P_SUB_SPOSMODE rendszerváltozóból lesz olvasva.

Programkód	Kommentár
N1000 PROC LANG_SUB DISPLOF SBLOF	
...	
N2100 IF($P_SUB_AXFCT==2)	
N2110	; SPOS / SPOSA / M19 utasítások helyettesítése aktív szinkronorsó-csatolásnál
N2185 DELAYFSTON	; állj késleltetés tartomány kezdete
N2190 COUPOF(S2,S1)	; szinkronorsó-csatolás deaktiválása
N2200	; vezető- és követő-orsót pozicionálni
N2210 IF($P_SUB_SPOS==TRUE) OR ($P_SUB_SPOSA==TRUE)	
N2220	; orsót SPOS-sal pozicionálni:
N2230 SPOS[1]=GP($P_SUB_SPOSIT, $P_SUB_SPOSMODE)	
SPOS[2]=GP($P_SUB_SPOSIT, $P_SUB_SPOSMODE)	
N2250 ELSE	
N2260	; orsót M19-cel pozicionálni:
N2270 M1=19 M2=19	; vezető- és követő-orsót pozicionálni
N2280 ENDIF	
N2285 DELAYFSTOF	; állj késleltetés tartomány vége
N2290 COUPON(S2,S1)	; szinkronorsó-csatolás aktiválása
N2410 ELSE	
N2420	; további helyettesítés lekérdezése
...	
N3300 ENDIF	
...	
N9999 RET	

Peremfeltételek

- A szinkronakciókban a pozíció tulajdonságok közvetett programozása nem lehetséges.

Irodalom

Alapfunkciók működési kézikönyv; BAG, Csatorna, Program üzem, Reset viselkedés (K1), fejezet: NC funkciók helyettesítése alprogramokkal

2.2.4 Munkadarabprogram sorok közvetett programozása (EXECSTRING)

AZ `EXECSTRING` programutasítással lehetséges egy előzőleg létrehozott string változót munkadarabprogram sorként végrehajtani.

Szintaxis

Az `EXECSTRING` egy külön munkadarabprogram sorban lesz programozva:
`EXECSTRING(<string változó>)`

Jelentés

`EXECSTRING`:	utasítás egy string változó végrehajtására munkadarabprogram sorként
`<string változó>`:	STRING típusú változó, amely a végrehajtandó munkadarabprogram sort tartalmazza

Megjegyzés

A `EXECSTRING`-gel a vezérlő-struktúrák (Oldal 113) kivételével az összes munkadarabprogram konstrukció megvalósítható, amelyek a **munkadarabprogram programrészben** programozhatók. Ezzel kizártak a `PROC` és `DEF` utasítások, valamint általában az INI és DEF fájlok.

Példa

Programkód	Kommentár
`N100 DEF STRING[100] MY_BLOCK`	`; String-változó definiálása a végrehajtandó munka-darabprogram-sorok számára.`
`N110 DEF STRING[10] MFCT1="M7"`	
`...`	
`N200 EXECSTRING(MFCT1 << "M4711")`	`; "M7 M4711" munkadarabprogram-sor végrehajtása`
`...`	
`N300 R10=1`	
`N310 MY_BLOCK="M3"`	
`N320 IF(R10)`	
`N330 MY_BLOCK = MY_BLOCK << MFCT1`	
`N340 ENDIF`	
`N350 EXECSTRING(MY_BLOCK)`	`; "M3 M7" munkadarabprogram-sor végrehajtása.`

2.3 Számítási műveletek

Operátorok / számítási funkciók	Jelentés
+	összeadás
–	kivonás
*	szorzás
/ [1]	osztás [1]
DIV [1]	egész-szám osztás[1]
MOD [1]	modulo osztás (egy egész-szám osztás maradékát adja) [1]
:	láncolás operátor FRAME változókhoz
SIN()	sinus
COS()	cosinus
TAN	tangens
ASIN	arcussinus
ACOS	arcuscosinus
ATAN2(,) [1]	arcustangens2 [1]
SQRT()	négyzetgyök
ABS()	érték
POT()	2. hatvány (négyzet)
TRUNC()	egész-számú rész pontosság összehasonlító műveleteknél beállítható TRUNC-kal (lásd"Pontosság-korrekció összehasonlítási hibáknál (TRUNC) (Oldal 78)")
ROUND()	kerekítés egész-számra
LN()	természetes logaritmus
EXP()	exponenciális függvény
MINVAL()	kisebb érték két változóból (lásd "Változók minimuma, maximuma és tartománya (MINVAL, MAXVAL, BOUND) (Oldal 80)")
MAXVAL()	nagyobb érték két változóból (lásd "Változók minimuma, maximuma és tartománya (MINVAL, MAXVAL, BOUND) (Oldal 80)")
BOUND()	változóérték a definiált értéktartományban (lásd "Változók minimuma, maximuma és tartománya (MINVAL, MAXVAL, BOUND) (Oldal 80)")
CTRANS()	eltolás
CROT()	forgatás
CSCALE()	lépték változtatás
CMIRROR()	tükrözés
1) lásd "Példák" bekezdés	

Programozás

A számítási műveleteknél a szokásos matematikai írásmód érvényes. A feldolgozási sorrendet a kerek zárójelekkel lehet meghatározni. A trigonometriai és azok inverz függvényeire a szögmegadás érvényes (derékszög = 90°).

Példák

Osztás: /

(REAL típus) = (INT típus vagy REAL típus) / (INT típus vagy REAL típus);

Példa: 3 / 4 = 0.75

Egész-szám osztás: DIV

(INT típus) = (INT típus vagy REAL típus) / (INT típus vagy REAL típus);

Példa: 7 DIV 4.1 = 1

Modulo osztás (egy egész-szám osztás maradékát adja): MOD

(REAL típus) = (INT típus vagy REAL típus) MOD (INT típus vagy REAL típus);

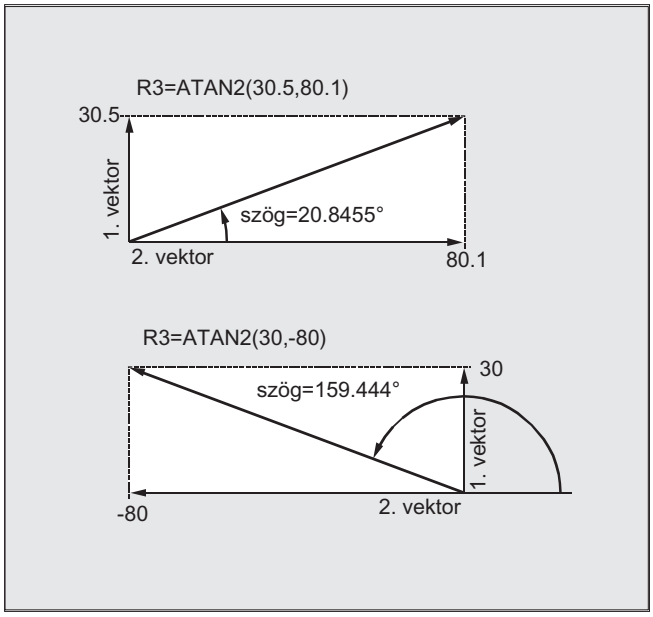
Példa: 7 MOD 4.1 = 2.9

Arcustangens2: ATAN2

AZ ATAN2 számítási funkció kiszámítja két egymásra merőleges vektorból az eredő vektor szögét.

Az eredmény a négy negyed tartományában van (–180° < 0 < +180°).

A szög vonatkoztatásának alapja mindig a 2. értéktől pozitív irányban.



Programozási példák

Programkód	Kommentár
R1=R1+1	; új R1 = régi R1 + 1
R1=R2+R3 R4=R5-R6 R7=R8*R9	
R10=R11/R12 R13=SIN(25.3)	
R14=R1*R2+R3	; A pont-műveletek megelőzik a vonal-művelteket.
R14=(R1+R2)*R3	; Először a zárójelek lesznek kiszámítva.
R15=SQRT(POT(R1)+POT(R2))	; Először a belső zárójelek lesznek feloldva:
	R15 = négyzetgyök((R1^2 + R2^2))
RESFRAME=FRAME1:FRAME2	; FRAME kapcsolat láncolás operátorral
FRAME3=CTRANS(…):CROT(…)	Érték hozzárendelés egy FRAME komponenshez

2.4 Összehasonlító és logikai műveletek

Összehasonlító operátorok használhatók pl. egy ugrás-feltétel megadásához. Összehasonlíthatók összetett kifejezések is.

Az összehasonlító műveletek a `CHAR`, `INT`, `REAL` és `BOOL` típusú változókra alkalmazhatók. A `CHAR` típusnál a kódérték kerül összehasonlításra.
A `STRING`, `AXIS` és `FRAME` típusoknál lehetséges: **==** und **<>**, amelyek a `STRING` típus műveleteire szinkron-akciókban is alkalmazhatók.

Az összehasonlító műveletek eredménye mindig `BOOL` típusú.

Logikai műveletek az igazság-értékek összekapcsolását szolgálják.

A logikai műveletek csak a `BOOL` típusú változókra alkalmazhatók. Belső típus-átalakításokkal a `CHAR`, `INT`, és `REAL` adattípusokra is alkalmazhatók.

A logikai (Bool) műveleteknél a `BOOL`, `CHAR`, `INT` és `REAL` adattípusokra érvényes:

- 0-nak megfelel: FALSE
- nem egyenlő 0-nak megfelel: TRUE

Bitenkénti logikai operátorok

A `CHAR` és `INT` típusú változókkal lehet bitenkénti logikai műveleteket is végezni. Adott esetben a típuskonverzió automatikusan történik.

Programozás

Összehasonlító operátor	Jelentés
==	egyenlő
<>	nem egyenlő
>	nagyobb
<	kisebb
>=	nagyobb vagy egyenlő
<=	kisebb vagy egyenlő

Logikai operátorok	Jelentés
AND	ÉS
OR	VAGY
NOT	negálás
XOR	kizáró VAGY

Bitenkénti logikai operátorok	Jelentés
B_AND	bitenkénti ÉS
B_OR	bitenkénti VAGY
B_NOT	bitenkénti negálás
B_XOR	bitenkénti kizáró VAGY

Megjegyzés

Az aritmetikai kifejezésekben a kerek zárójelek alkalmazásával a műveletek feldolgozási sorrendjét meg lehet adni és ezzel eltérni a normális prioritási szabályoktól.

Megjegyzés

A BOOL operandusok és az operátorok között kihagyás kell legyen.

Megjegyzés

A `B_NOT` operátor csak egy operandusra vonatkozik. Ez a művelet után áll.

Példák

Példa 1: Összehasonlító operátorok

```
IF R10>=100 GOTOF ZIEL
```

vagy

```
R11=R10>=100
IF R11 GOTOF ZIEL
```

Az `R10>=100` összehasonlítás eredménye az `R11` közbenső tárolásra kerül.

Példa 2: Logikai operátorok

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

vagy

```
IF NOT R10 GOTOB START
```

A `NOT` operátor csak egy operandusra vonatkozik.

Példa 3: Bitenkénti logikai operátorok

```
IF $MC_RESET_MODE_MASK B_AND 'B10000' GOTOF ACT_PLANE
```

2.5 Pontosság-korrekció összehasonlítási hibáknál (TRUNC)

A TRUNC utasítás levágja az egy pontossági-tényezővel összeszorzott operandusokat.

Beállítható pontosság összehasonlító utasításoknál

A REAL típusú munkadarabprogram-adatok belül 64 bites IEEE-formátumban vannak ábrázolva. Ezen ábrázolási forma miatt a decimális számok leképezése pontatlan lehet, ami az ideálisan számított értékkel összehasonlítva váratlan eredményhez vezethet.

Relatív egyenlőség

Az összehasonlító utasításoknál nem az abszolút egyenlőség, hanem egy relatív egyenlőség lesz megvizsgálva, így az ábrázolási forma által okozott pontatlanság nem befolyásolja a program lefutását.

Szintaxis

Pontosság-korrekció összehasonlítási hibáknál

```
TRUNC (R1*1000)
```

Jelentés

TRUNC:	tizedesjegyek levágása

A figyelembe vett relatív egyenlőség 10^{-12} a következőknél

- egyenlőség: (==)
- nem egyenlőség: (<>)
- nagyobb-egyenlő (>=)
- kisebb-egyenlő: (<=)
- nagyobb/kisebb: (><) abszolút egyenlőséggel
- nagyobb: (>)
- kisebb: (<)

Kompatibilitás

Kompatibilitási okokból a relatív egyenlőség vizsgálatát (>) és (<) esetén az MD 10280: PROG_FUNCTION_MASK Bit0 = 1 gépadat beállításával deaktiválni lehet.

Megjegyzés

A REAL típusú adatok összehasonlítása az előbbi okokból bizonyos pontatlansággal jár. Nem elfogatható eltéréseknél át kell térni INTEGER-számításokra, amennyiben az operandust egy pontossági tényezővel megszorozzuk és utána TRUNC-kal levágjuk.

Szinkron-akciók

Az összehasonlító műveletek leírt viselkedése érvényes szinkron-akcióknál is.

Példák

Példa 1: Pontosság személetek

Programkód	Kommentár
N40 R1=61.01 R2=61.02 R3=0.01	; kezdő értékek hozzárendelése
N41 IF ABS(R2-R1) > R3 GOTOF FEHLER	; az ugrás eddig végre lett volna hajtva
N42 M30	; programvég
N43 FEHLER: SETAL(66000)	
R1=61.01 R2=61.02 R3=0.01	; kezdő értékek hozzárendelése
R11=TRUNC(R1*1000) R12=TRUNC(R2*1000) R13=TRUNC(R3*1000)	; pontosság-korrekció
IF ABS(R12-R11) > R13 GOTOF FEHLER	; az ugrás többé nem lesz végre-hajtva
M30	; programvég
FEHLER: SETAL(66000)	

Példa 2: Két operandus hányadosának képzésére és kiértékelésére

Programkód	Kommentár
R1=61.01 R2=61.02 R3=0.01	; kezdő értékek hozzárendelése
IF ABS((R2-R1)/R3)-1) > 10EX-5 GOTOF FEHLER	; az ugrás nem lesz végrehajtva
M30	; programvég
FEHLER: SETAL(66000)	

2.6 Változók minimuma, maximuma és tartománya (MINVAL, MAXVAL, BOUND)

A MINVAL és MAXVAL műveletekkel össze lehet egymással hasonlítani két változó értékét. Eredményként a kisebb érték (MINVAL-nál) ill. a nagyobb érték (MAXVAL-nál) lesz visszaadva.

A BOUND művelet megvizsgálja, hogy egy vizsgált változó értéke a definiált értéktartományban van-e.

Szintaxis

```
<kisebb érték>=MINVAL(<változó1>,<változó2>)
<nagyobb érték>=MAXVAL(<változó1>,<változó2>)
<visszaadás érték>=<BOUND>(<minimum>,<maximum>,<vizsgált változó>)
```

Jelentés

MINVAL:	megállapítja a **kisebb** értéket két változóból (<változó1>, <változó2>)
<kisebb érték>:	a MINVAL utasítás eredmény-vátozója a kisebb változóértékre lesz beállítva
MAXVAL:	megállapítja a **nagyobb** értéket két változóból (<változó1>, <változó2>)
<nagyobb érték>:	a MAXVAL utasítás eredmény-vátozója a nagyobb változóértékre lesz beállítva
BOUND:	megvizsgálja, hogy egy változó (<vizsgált változó>) egy definiált értéktartományon belül van-e
<minimum>:	változó, ami az értéktartomány minimum értékét definiálja
<maximum>:	változó, ami az értéktartomány maximum értékét definiálja
<visszaadási érték>:	a BOUND utasítás eredmény-vátozója Ha a vizsgált változó értéke a definiált értéktartományon belül van, akkor az eredmény-változó a vizsgált változó értékére lesz beállítva. Ha a vizsgált változó értéke nagyobb a maximum értéknél, akkor az eredmény-változó a definíciós tartomány maximum értékére lesz beállítva. Ha a vizsgált változó értéke kisebb a minimum értéknél, akkor az eredmény-változó a definíciós tartomány minimum értékére lesz beállítva.

Megjegyzés

MINVAL, MAXVAL és BOUND programozható szinkron-akciókban is.

Megjegyzés

Viselkedés egyenlőségnél

Egyenlőség esetén a MINVAL/MAXVAL műveleteknél azonos érték lesz visszaadva. A BOUND esetén a vizsgálandó változó értéke lesz visszaadva.

Példa

Programkód	Kommentár
DEF REAL rVar1=10.5, rVar2=33.7, rVar3, rVar4,	rVar5, rValMin, rValMax, rRetVar
rValMin=MINVAL(rVar1,rVar2)	; rValMin értékét 10.5-re állítja
rValMax=MAXVAL(rVar1,rVar2)	; rValMax értékét 33.7-re állítja
rVar3=19.7	
rRetVar=BOUND(rVar1,rVar2,rVar3)	; rVar3 a határokon belül van, rRetVar értékét 19.7-re állítja.
rVar3=1.8	
rRetVar=BOUND(rVar1,rVar2,rVar3)	; rVar3 a minimum-határ alatt van, rRetVar értékét 10.5-re állítja.
rVar3=45.2	
rRetVar=BOUND(rVar1,rVar2,rVar3)	; rVar3 a maximum-határ felett van, rRetVar értékét 33.7-re állítja.

2.7 Műveletek prioritása

Minden művelethez van egy prioritás rendelve. Egy kifejezés kiértékelésénél először a magasabb prioritású műveletek lesznek végrehajtva. Azonos prioritású műveleteknél a végrehajtás balról jobbra halad.

Az aritmetikai kifejezésekben a kerek zárójelek alkalmazásával a műveletek feldolgozási sorrendjét meg lehet adni és ezzel eltérni a normális prioritási szabályoktól.

Műveletek sorrendje

A legmagasabbtól a legalacsonyabb prioritásig

1.	NOT, B_NOT	negálás, bitenkénti negálás
2.	*, /, DIV, MOD	szorzás, osztás
3.	+, –	összeadás, kivonás
4.	B_AND	bitenkénti ÉS
5.	B_XOR	bitenkénti kizáró VAGY
6.	B_OR	bitenkénti VAGY
7.	AND	ÉS
8.	XOR	kizáró VAGY
9.	OR	VAGY
10.	<<	stringek láncolása, eredmény típus STRING
11.	==, <>, >, <, >=, <=	összehasonlító operátorok

Megjegyzés

A láncolás operátor ":" framek-re nem lehet más műveletekkel egy kifejezésben. Erre a műveletre nem szükséges prioritás megadása.

Példa If-utasítással

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

2.8 Lehetséges típus-konverziók

Funkció

Típus-konverzió hozzárendelésnél

Az állandó érték, a változó vagy a kifejezés, amit egy változóhoz hozzárendelünk, annak a típusával összeférő kell legyen. Ha ez adott, a hozzárendelésnél a típus-konverzió automatikusan megtörténik.

Lehetséges típus-konverziók

-ba	REAL	INT	BOOL	CHAR	STRING	AXIS	FRAME
-ból							
REAL	igen	igen*	igen[1])	igen*	–	–	–
INT	igen	igen	igen[1])	igen[2])	–	–	–
BOOL	igen	igen	igen	igen	igen	–	–
CHAR	igen	igen	igen[1])	igen	igen	–	–
STRING	–	–	igen[4])	igen[3])	igen	–	–
AXIS	–	–	–	–	–	igen	–
FRAME	–	–	–	–	–	–	igen

Magyarázatok

* A típus-konverziónál REAL-ból INT-ra a >= 0.5 törtérték felfelé lesz kerekítve, egyébként lefelé (mint a ROUND funkció)

[1]) érték <> 0-ak megfelel a TRUE-nak, érték== 0-nak megfelel a FALSE-nak

[2]) ha az érték a megengedett számtartományban van

[3]) ha csak 1 karakter

[4]) string-hossz 0 = > FALSE, egyébként TRUE

Megjegyzés

Ha a konverziónál az érték nagyobb, mint a céltartomány, hibajelzés keletkezik.

Ha egy kifejezésben vegyesen vannak típusok, a típusok illesztése automatikusan megtörténik. Típus-átalakítások lehetségesek szinkron-akcióknál is, lásd. Mozgásszinkron-akciók, implicit típus-átalakítás.

2.9 String műveletek

String műveletek

A klasszikus "Hozzárendelés" és "Összehasonlítás" műveletek mellett további lehetőségek vannak a stringek manipulálására:

- Típus-konverzió STRING-re (AXSTRING) (Oldal 84)
- Típus-konverzió STRING-ből (NUMBER, ISNUMBER, AXNAME) (Oldal 85)
- Stringek láncolása (<<) (Oldal 86)
- Kis- /nagybetű átalakítás (TOLOWER, TOUPPER) (Oldal 87)
- String hosszának meghatározása (STRLEN) (Oldal 88)
- Karakter/string keresése stringben (INDEX, RINDEX, MINDEX, MATCH) (Oldal 89)
- Rész-string kiválasztása (SUBSTR) (Oldal 90)
- Egyes karakterek írása és olvasása (Oldal 91)
- String formattálás (SPRINT) (Oldal 92)

A 0-karakter külön jelentése

A 0-karakter belül egy string-vége jelként van értelmezve. Ha egy jelet a 0-karakterrel helyettesítünk, ezzel a stringet lerövidítjük.

Példa:

Programkód	Kommentár
`DEF STRING[20] STRG="Achse . steht"`	
`STRG[6]="X"`	
`MSG(STRG)`	`; "X tengely áll" jelzést adja.`
`STRG[6]=0`	
`MSG(STRG)`	`; "Tengely" jelzést adja.`

2.9.1 Típus-konverzió STRING-re (AXSTRING)

A "Típus-konverzió STRING-re" funkcióval lehetséges a különböző típusú változókat egy üzenet (MSG) részeként használni.

A << művelet alkalmazásánál implicit történik az INT, REAL, CHAR és BOOL típusokra (lásd " Stringek láncolása (<<) (Oldal 86)").

Egy INT-érték a normál olvasható formába lesz átalakítva. A REAL-értékek a tizedespont után 10 helyig lesznek megadva.

Az `AXSTRING` utasítással változókat lehet az AXIS típusból STRING-re átalakítani.

Szintaxis

```
<STRING_ERG> = << <tetsz._típ>
<STRING_ERG> = AXSTRING(<tengely-jelölő>)
```

Jelentés

<STRING_ERG>:	változó a típus-konverzió eredményének	
	típus:	STRING
<tetsz._típ>:	INT, REAL, CHAR, STRING és BOOL változó-típusok	
AXSTRING:	Az AXSTRING utasítás a megadott tengely-jelölőt szolgáltatja string-ként.	
<tengely-jelölő>:	változó a tengely-jelölőnek	
	típus:	AXIS

Megjegyzés

A FRAME-változók nem konvertálhatók.

2.9.2 Típus-konverzió STRING-ből (NUMBER, ISNUMBER, AXNAME)

A NUMBER utasítással lehet STRING-ből REAL-re konvertálni. A konvertálhatóságot az ISNUMBER utasítással lehet megvizsgálni.

Az AXNAME utasítással egy stringet az AXIS adattípusra lehet konvertálni.

Szintaxis

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

Jelentés

NUMBER:	A NUMBER utasítás a <String>-gel ábrázolt számot REAL értékként adja vissza.	
<String>:	STRING típusú konvertálandó változó	
<REAL_ERG>:	változó a NUMBER típus-konverzió eredményének	
	típus:	REAL
ISNUMBER:	Az ISNUMBER utasítással meg lehet vizsgálni, hogy a <String>et át lehet-e alakítani egy érvényes számra.	

<BOOL_ERG>:	változó az ISNUMBER lekérdezés eredményének		
	típus:	BOOL	
	Érték:	TRUE	Az ISNUMBER az IGAZ értéket adja, ha a <String> egy, a nyelv szabályainak megfelelő érvényes REAL-számot ad meg.
		FALSE	Ha az ISNUMBER a FALSE értéket adja, a NUMBER felhívásánál azonos <String>-gel vészjelzés lesz kiadva.
AXNAME:	Az AXNAME utasítás a megadott <String>-et átalakítja egy tengely-jelölővé. **Utalás:** Ha a <String>-hez nem lehet egy megtervezett tengely-jelölőt hozzárendelni, vészjelzés lesz kiadva.		
<AXIS_ERG>:	változó az AXNAME típus-konverzió eredményének		
	típus:	AXIS	

Példa

Programkód	Kommentár
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 Stringek láncolása (<<)

A "Stringek láncolása' funkció lehetővé teszi egy string összeállítását egyes részekből.

A láncolásra az "<<" operátor van lefoglalva. Ennek a műveletnek az eredmény-típusa a CHAR, BOOL, INT, REAL és STRING alaptípusok összes kombinációjára STRING. Egy esetleg szükséges konverzió a megadott szabályok szerint történik.

Szintaxis

```
<tetsz._típ> << <tetsz._típ>
```

Jelentés

<tetsz._típ> :	CHAR, BOOL, INT, REAL vagy STRING változó-típusok
<< :	operátor a <tetsz._típ> változók láncolására egy összetett karakterlánccá (STRING típusú) Ez az operátor egyedül úgynevezett "unáris" változatban áll rendelkezésre. Ezzel lehetséges egy explicit típus-átalakítás STRING-re (FRAME és AXIS esetén nem). << <tetsz._típ>

Például így egy jelzést vagy egy parancsot szöveglistákból össze lehet állítani és paramétereket (modulneveket) beilleszteni:
`MSG(STRG_TAB[LOAD_IDX]<<BAUSTEIN_NAME)`

Megjegyzés

A közbenső eredmény a string láncolásánál nem lépheti túl a maximális string-hosszat.

Megjegyzés

A FRAME és AXIS típusok ezzel a "<<" operátorral nem használhatók.

Példák

Példa 1: Stringek láncolása

Programkód	Kommentár
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)	; kijelzés: "index:2/érték:9.654"
NO_MSG:	

Példa 2: Explicit típus-konverzió <<-vel

Programkód	Kommentár
DEF REAL VALUE=3.5	
<<VALUE	; A megadott REAL típusú változó STRING típusúra lesz átalakítva.

2.9.4 Kis- /nagybetű átalakítás (TOLOWER, TOUPPER)

A "Kis- /nagybetű átalakítás" funkció lehetővé teszi egy string minden betűjét egységes ábrázolásúra alakítani.

Szintaxis

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

Jelentés

TOUPPER:	A TOUPPER utasítással egy karakterlánc összes betűje **nagy**betűre lesz átalakítva.	
TOLOWER:	A TOLOWER utasítással egy karakterlánc összes betűje **kis**betűre lesz átalakítva.	
<String>:	átalakítandó karakterlánc	
	típus:	STRING
<STRING_ERG>:	változó az átalakítás eredményének	
	típus:	STRING

Példa

Mivel ez lehetséges a kezelőfelület alkalmazói beadásaira is, az egységes ábrázolás kis- vagy nagybetűkkel elérhető:

Programkód

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

2.9.5 String hosszának meghatározása (STRLEN)

A STRLEN utasítás lehetővé teszi egy karakterlánc hosszának a meghatározását.

Szintaxis

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

Jelentés

STRLEN:	A STRLEN utasítással lesz meghatározva a megadott karakterlánc hossza. A kezdettől számítva a 0-tól eltérő karakterek száma lesz visszaadva.	
<String>:	karakterlánc, aminek a hosszát meg kell határozni	
	típus:	STRING
<INT_ERG>:	változó a meghatározás eredményének	
	típus:	INT

Példa

A funkció lehetővé teszi az egyes karakter eléréssel együtt egy string végének a meghatározását:

Programkód

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

2.9.6 Karakter/string keresése stringben (INDEX, RINDEX, MINDEX, MATCH)

Ez a funkció lehetővé teszi egyes karakterek ill. egy string keresését egy további stringben. A funkció eredménye megadja, hogy a megvizsgált string melyik helyén található a keresett karakter/string.

Szintaxis

`INT_ERG=INDEX(STRING,CHAR)` ; **eredmény-típus: INT**

`INT_ERG=RINDEX(STRING,CHAR)` ; **eredmény-típus: INT**

`INT_ERG=MINDEX(STRING,STRING)` ; **eredmény-típus: INT**

`INT_ERG=MATCH(STRING,STRING)` ; **eredmény-típus: INT**

Szemantika

Kereső funkciók: Ezek visszaadják azt a helyet a stringben (első paraméter), ahol a keresés sikeres volt. Ha a karakter/string nem található, a –1 érték kerül visszaadásra. Az első hely itt a 0 pozíció.

Jelentés

INDEX:	keresi a második paraméterként megadott karaktert (elölről) az első paraméterben
RINDEX:	keresi a második paraméterként megadott karaktert (hátulról) az első paraméterben
MINDEX:	megfelel az INDEX funkciónak azzal az eltéréssel, hogy a karakterek egy listája kerül megadásra (stringként), amelyekből az első megtalált karakter indexe lesz visszaadva
MATCH:	keres egy stringet egy stringben

Így lehet a stringeket meghatározott szempontok szerint szétbontani, pl. az üres jeleknél vagy az ág elválasztó jeleknél ("/").

Példa

Egy beadás felbontására ág- és modulnévre

Programkód	Kommentár
DEF INT PFADIDX, PROGIDX	
DEF STRING[26] EINGABE	
DEF INT LISTIDX	

Programkód	Kommentár
EINGABE = "/_N_MPF_DIR/_N_EXECU-TE_MPF"	
LISTIDX = MINDEX (EINGABE, "M,N,O,P") + 1	; A LISTIDX a 3 értéket adja vissza; mivel az "N" az első karakter elölről a EINGABE paramé-terben a választék-listából.
PFADIDX = INDEX (EINGABE, "/") +1	; ezzel érvényes: PFADIDX = 1
PROGIDX = RINDEX (EINGABE, "/") +1	; ezzel érvényes: PROGIDX = 12
	; a következő bekezdésben bevezetésre kerülő SUBSTR funkció segítségével a EINGABE változót az "ág" és a "modul" kompo-nensekre lehet felbontani:
VARIABLE = SUBSTR (EINGABE, PFADIDX, PROGIDX-PFADIDX-1)	; "_N_MPF_DIR"-t adja
VARIABLE = SUBSTR (EINGABE, PROGIDX)	; "_N_EXECUTE_MPF"-t adja

2.9.7 Rész-string kiválasztása (SUBSTR)

A SUBSTRING funkcióval egy string-en belül tetszőleges részeket lehet olvasni.

Szintaxis

```
<STRING_ERG>=SUBSTR(<string>,<index>,<hossz>)
<STRING_ERG>=SUBSTR(<string>,<index>)
```

Jelentés

SUBSTR:	A funkció a <string>-ből egy rész sring-et ad az <index>-től kezdődően a megadott <hossz>-szal Ha a <hossz> paraméter nincs megadva, a funkció egy rész string-et ad az <index>-től a string végéig.
<Index>:	A rész string kezdő pozíciója a string-en belül. Ha a kezdő pozíció a string vége mögött van, egy üres string (" ") kerül visszaadásra. A string első karaktere: Index = 0 értéktartomány: 0 ... (string-hossz - 1)
<hossz>:	rész string hossza Túl nagy hossz megadása esetén csak a rész string a string végéig lesz visszaadva. értéktartomány: 1 ... (string-hossz - 1)

Példa

Programkód	Kommentár
DEF STRING[29] ERG	

```
Programkód                                          Kommentár
;                        1
;              0123456789012345678
ERG = SUBSTR("QUITTUNG: 10 ... 99", 10, 2)          ; ERG == "10"
ERG = SUBSTR("QUITTUNG: 10 ... 99", 10)             ; ERG == "10... 99"
```

2.9.8 Egyes karakterek írása és olvasása

Egy string-en belül lehet olvasni és írni egyes karaktereket.

Ennél a következő peremfeltételeket kell figyelembe venni:

- csak felhasználó által definiált változóknál lehetséges, rendszerváltozóknál nem
- egy string egyes karakterei az alprogramhívásoknál csak "call by value" lesznek átdva

Szintaxis

```
<karakter>=<string>[<index>]
<karakter>=<String_Array>[<Array_Index>,<Index>]
<String>[<Index>]=<karakter>
<String_Array>[<Array_Index>,<Index>]=<karakter>
```

Jelentés

`<String>`:	tetszőleges string
`<karakter>`:	CHAR típusú változó
`<Index>`:	karakter kezdő pozíciója a string-en belül A string első karaktere: Index = 0 értéktartomány: 0 ... (string-hossz - 1)

Példák

Példa 1: Változó jelentés

```
Programkód                          Kommentár
;                        0123456789
DEF STRING [50] MELDUNG = "Tengely n elérte a pozíciót"
MELDUNG [6] = "X"
MSG (MELDUNG)                       ; "Tengely X elérte a pozíciót"
```

Példa 2: Egy rendszerváltozó kiértékelése

```
Programkód                          Kommentár
DEF STRING [50] STRG                ; közbenső tároló rendszerváltozónak
...
STRG = $P_MMCA                      ; rendszerváltozót betölteni
```

Programkód	Kommentár
IF STRG[0] == "E" GOTO ...	; rendszerváltozó kiértékelése

Példa 3: Paraméter átadás "call by value" és "call by reference"

Programkód	Kommentár
; 0123456	
DEF STRING[50] STRG = "Achse X"	
DEF CHAR CHR	
...	
EXTERN UP_VAL(ACHSE)	; alprogram definíció "call by value" paraméterrel
EXTERN UP_REF(VAR ACHSE)	; alprogram definíció "call by ref." paraméterrel Paraméter
...	
UP_VAL(STRG[6])	; paraméter átadás "by value"
...	
CHR = STRG[6]	; közbenső tárolás
UP_REF(CHR)	; paraméter átadás "by reference"

2.9.9 String formattálás (SPRINT)

A SPRINT előre definiált funkcióval karakterláncokat lehet formatálni és pl. külső készülékre kiadáshoz előkészíteni (lásd "Process DataShare - Kiadás egy külső készülékre/fájlba (EXTOPEN, WRITE, EXTCLOSE) (Oldal 657)" is).

Szintaxis

```
"<eredmény_string>"=SPRINT("<formátum_string>",<érték_1>,<érték_2>,.
.., <érték_n>)
```

Jelentés

SPRINT:	Egy előre definiált funkció jelölője, ami egy STRING típusú értéket szállít.
"<formátum_string>":	karakterlánc, ami fix és változó részeket tartalmaz A változó részek egy % formátumjellel és egy utána következő formátum-leírással vannak megadva.
< érték_1>,< érték_2>,…,< érték_n>:	Érték egy állandó vagy egy NC-változó formájában, ami azon a helyen, ahol az n. % formátumjel áll, a <formátum_string> formátum-leírásnak megfelelően be lesz szúrva.
"<eredmény_string>":	formattált karaktersor (maximum 400 bájt)

Rendelkezésre álló formátum-leírások

%B:	Átalakítás a "TRUE" stringbe, ha az átalakítandó érték:
	• nem egyenlő 0.
	• nem üres-string (string értékeknél).
	Átalakítás a "FALSE" stringbe, ha az átalakítandó érték:
	• egyenlő 0.
	• üres-string.
	Példa:
	`N10 DEF BOOL BOOL_VAR=1`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF BOOL_VAR:%B", BOOL_VAR)`
	eredmény: A RESULT string-változó a "CONTENT OF BOOL_VAR:TRUE" karakterlánccal van leírva.

%C:	Átalakítás ASCII karakterbe.
	Példa:
	`N10 DEF CHAR CHAR_VAR="X"`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF CHAR_VAR:%C",CHAR_VAR)`
	eredmény: A RESULT string-változó a "CONTENT OF CHAR_VAR:X" karakterlánccal van leírva.

%D:	Átalakítás egy stringbe egy egészszám értékkel (INTEGER).
	Példa:
	`N10 DEF INT INT_VAR=123`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF INT_VAR:%D",INT_VAR)`
	eredmény: A RESULT string-változó a "CONTENT OF INT_VAR:123" karakterlánccal van leírva.

%<m>D:	Átalakítás egy stringbe egy egészszám értékkel (INTEGER). A string legalább <m> karakter hosszú. A hiányzó helyek balra igazítva üres jelekkel lesznek feltöltve.
	Példa:
	`N10 DEF INT INT_VAR=-123`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF INT_VAR:%6D",INT_VAR)`
	eredmény: A RESULT string-változó a "CONTENT OF INT_VAR:xx-123" karakterlánccal van leírva ("x" a példában az üres helyeket jelenti).

%F:	Átalakítás egy stringbe egy 6 tizedesjegyű decimális számmal. A tizedesjegyek esetleg kerekítve lesznek vagy 0-val feltöltve.
	Példa:
	`N10 DEF REAL REAL_VAR=-1.2341234EX+03`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%F",REAL_VAR)`
	eredmény: A RESULT string-változó a "CONTENT OF REAL_VAR: -1234.123400" értékkel lesz írva.

%<m>F:	Átalakítás egy stringbe egy 6 tizedesjegyű és legalább <m> karakter hosszú decimális számmal. A tizedesjegyek esetleg kerekítve lesznek vagy 0-val feltöltve. A hiányzó karakterek az <m> összhosszra balra igazítva üres-jelekkel lesznek feltöltve.
	Példa:
	`N10 DEF REAL REAL_VAR=-1.23412345678EX+03`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%15F",REAL_VAR)`
	eredmény: A RESULT string-változó a "CONTENT OF REAL_VAR: xxx-1234.123457" karakterlánccal van leírva ("x" a példában az üres-helyet jelenti).

%.<n>F:	Átalakítás egy stringbe egy <n> tizedesjegyű decimális számmal. A tizedesjegyek esetleg kerekítve lesznek vagy 0-val feltöltve.
	Példa:
	`N10 DEF REAL REAL_VAR=-1.2345678EX+03`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%.3F",REAL_VAR)`
	eredmény: A RESULT string-változó a "CONTENT OF REAL_VAR:-1234.568" karakterlánccal van leírva.

%<m>.<n>F:	Átalakítás egy stringbe egy <n> tizedesjegyű és legalább <m> karakter hosszú decimális számmal. A tizedesjegyek esetleg kerekítve lesznek vagy 0-val feltöltve. A hiányzó karakterek az <m> összhosszra balra igazítva üres-jelekkel lesznek feltöltve.
	Példa:
	`N10 DEF REAL REAL_VAR=-1.2341234567890EX+03`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%10.2F",REAL_VAR)`
	eredmény: A RESULT string-változó a "CONTENT OF REAL_VAR:xx-1234.12" karakterlánccal van leírva ("x" a példában az üres helyeket jelenti).

%E:	Átalakítás egy stringbe egy exponenciális ábrázolású decimális számmal. Az alap normalizálva egy jeggyel és 6 tizedesjeggyel lesz tárolva. A tizedesjegyek esetleg kerekítve lesznek vagy 0-val feltöltve. A kitevő az "EX" kulcsszóval kezdődik. Ezt az előjel ("+" vagy "-") és egy két- vagy háromjegyű szám követi.
	Példa:
	`N10 DEF REAL REAL_VAR=-1234.567890`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%E",REAL_VAR)`
	eredmény: A RESULT string-változó a "CONTENT OF REAL_VAR:-1.234568EX +03" karakterlánccal van leírva.

%<m>E:	Átalakítás egy stringbe egy legalább <m> karakter hosszú exponenciális ábrázolású decimális számmal. A hiányzó helyek balra igazítva üres jelekkel lesznek feltöltve. Az alap normalizálva egy jeggyel és 6 tizedesjeggyel lesz tárolva. A tizedesjegyek esetleg kerekítve lesznek vagy 0-val feltöltve. A kitevő az "EX" kulcsszóval kezdődik. Ezt az előjel ("+" vagy "-") és egy két- vagy háromjegyű szám követi.
	Példa:
	`N10 DEF REAL REAL_VAR=-1234.5`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%20E",REAL_VAR)`
	eredmény: A RESULT string-változó a "CONTENT OF REAL_VAR:xxxxxx-1.234500EX+03" karakterlánccal van leírva ("x" a példában az üres helyeket jelenti).

%.<n>E:	Átalakítás egy stringbe egy exponenciális ábrázolású decimális számmal. Az alap normalizálva egy jeggyel és <n> tizedesjeggyel lesz tárolva. A tizedesjegyek esetleg kerekítve lesznek vagy 0-val feltöltve. A kitevő az "EX" kulcsszóval kezdődik. Ezt az előjel ("+" vagy "-") és egy két- vagy háromjegyű szám követi.
	Példa:
	`N10 DEF REAL REAL_VAR=-1234.5678`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%.2E",REAL_VAR)`
	eredmény: A RESULT string-változó a "CONTENT OF REAL_VAR:-1.23EX+03" karakterlánccal van leírva.

%<m>.<n>E:	Átalakítás egy stringbe egy legalább <m> karakter hosszú exponenciális ábrázolású decimális számmal. A hiányzó helyek balra igazítva üres jelekkel lesznek feltöltve. Az alap normalizálva egy jeggyel és <n> tizedesjeggyel lesz tárolva. A tizedesjegyek esetleg kerekítve lesznek vagy 0-val feltöltve. A kitevő az "EX" kulcsszóval kezdődik. Ezt az előjel ("+" vagy "-") és egy két- vagy háromjegyű szám követi.
	Példa:
	`N10 DEF REAL REAL_VAR=-1234.5678`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%12.2E", REAL_VAR)`
	eredmény: A RESULT string-változó a "CONTENT OF REAL_VAR:xx-1.23EX+03" karakterlánccal van leírva ("x" a példában az üres helyeket jelenti).

%G:	Átalakítás egy stringre egy, az értéktartománytól függően decimális vagy exponenciális ábrázolású decimális számmal: ha az ábrázolandó érték kisebb, mint 1.0EX-04 vagy nagyobb/egyenlő 1.0EX+06, az exponenciális ábrázolás lesz választva, egyébként a decimális ábrázolás. Maximum a hat szignifikáns jegy lesz kijelezve, esetleg kerekítve lesz.
	Példa tizedes ábrázolással:
	`N10 DEF REAL REAL_VAR=1.234567890123456EX-04`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%G",REAL_VAR)`
	eredmény: A RESULT string-változó a "CONTENT OF REAL_VAR:0.000123457" karakterlánccal van leírva.
	Példa exponenciális ábrázolással:
	`N10 DEF REAL REAL_VAR=1.234567890123456EX+06`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%G",REAL_VAR)`
	eredmény: A RESULT string-változó a "CONTENT OF REAL_VAR:1.23457EX+06" karakterlánccal van leírva.

%<m>G:	Átalakítás egy stringre egy, az értéktartománytól függően decimális vagy exponenciális ábrázolású decimális számmal (mint %G). A string összhossza legalább <m> karakter. A hiányzó helyek balra igazítva üres jelekkel lesznek feltöltve. **Példa tizedes ábrázolással:** N10 DEF REAL REAL_VAR=1.234567890123456EX-04 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%15G",REAL_VAR) eredmény: A RESULT string-változó a "CONTENT OF REAL_VAR:xxxx0.000123457" karakterlánccal van leírva ("x" a példában az üres helyeket jelenti). **Példa exponenciális ábrázolással:** N10 DEF REAL REAL_VAR=1.234567890123456EX+06 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%15G",REAL_VAR) eredmény: A RESULT string-változó a "CONTENT OF REAL_VAR:xxx1.23457EX +06" karakterlánccal van leírva ("x" a példában az üres helyeket jelenti).
%.<n>G:	Átalakítás egy stringre egy, az értéktartománytól függően decimális vagy exponenciális ábrázolású decimális számmal. Maximum az <n>szignifikáns jegy lesz kijelezve, esetleg kerekítve lesz. Ha az ábrázolandó érték kisebb, mint 1.0EX-04 vagy nagyobb/egyenlő 1.0EX(+<n>), az exponenciális ábrázolás lesz választva, egyébként a decimális ábrázolás. **Példa tizedes ábrázolással:** N10 DEF REAL REAL_VAR=1.234567890123456EX-04 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%.3G",REAL_VAR) eredmény: A RESULT string-változó a "CONTENT OF REAL_VAR:0.000123" karakterlánccal van leírva. **Példa exponenciális ábrázolással:** N10 DEF REAL REAL_VAR=1.234567890123456EX+03 N20 DEF STRING[80] RESULT N30 RESULT = SPRINT("CONTENT OF REAL_VAR:%.3G",REAL_VAR) eredmény: A RESULT string-változó a "CONTENT OF REAL_VAR:1.23EX+03" karakterlánccal van leírva.
%<m>.<n>G:	Átalakítás egy stringre egy, az értéktartománytól függően decimális vagy exponenciális ábrázolású decimális számmal (mint %.<n>G). A string összhossza legalább <m> karakter. A hiányzó helyek balra igazítva üres jelekkel lesznek feltöltve. **Példa tizedes ábrázolással:** N10 DEF REAL REAL_VAR=1.234567890123456EX-04 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%12.4G",REAL_VAR) eredmény: A RESULT string-változó a "CONTENT OF REAL_VAR:xxx0.0001235" karakterlánccal van leírva ("x" a példában az üres helyeket jelenti). **Példa exponenciális ábrázolással:** N10 DEF REAL REAL_VAR=1.234567890123456EX+04 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%12.4G",REAL_VAR) eredmény: A RESULT string-változó a "CONTENT OF REAL_VAR:xxx1.235EX+06" karakterlánccal van leírva ("x" a példában az üres helyeket jelenti).

%.<n>P:	Egy REAL érték átalakítása egy INTEGER értékre <n> érték figyelembe vételével. Az INTEGER érték 32 bites bináris számként lesz kiadva. Ha az átalakítandó értéket nem lehet 32 bittel ábrázolni, a feldolgozás vészjelzéssel lesz megszakítva.
	Mivel az %.<n>P formattálási utasítással létrehozott bájt-sor tartalmazhat bináris nullákat is, az így létrehozott össz-string már nem felel meg a STRING NC-adattípusnak. Ezért azt nem lehet egy STRING típusú változóban tárolni és az NC nyelv string utasításaival feldolgozni. Az egyetlen lehetséges alkalmazás a paraméter átadás a WRITE utasításnak kiadással a megfelelő külső készülékre (lásd a következő példát).
	Ha a <Format_String> egy %P típusú formátum-leírást tartalmaz, az egész string, az %.<n>P-vel generált bináris számok kivételével, az MD10750 $MN_SPRINT_FORMAT_P_CODE-nak megfelelően ASCII karakterekben, ISO (DIN6024) vagy EIA (RS244) szerint lesz kiadva. Ha egy nem átalakítható karakter van programozva, a feldolgozás vészjelzéssel lesz megszakítva.
	Példa:
	`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`
	`...  ; Hiba kezelés`
	`N200 WRITE(ERROR,EXT_DEVICE,SPRINT("INTEGER BINARY CODED:%.3P",REAL_VAR)`
	`N210 IF ERROR <> 0`
	`...  ; Hiba kezelés`
	eredmény: A "INTEGER BINARY CODED: 'H0001E23A'" string ki lesz adva a kiviteli készülék /ext/dev/1-re. A 0x0001E23A hexadecimális érték megfelel a 123450 decimális értéknek.

%<m>.<n>P:	Egy REAL érték átalakítása stringbe az MD10751 $MN_SPRINT_FORMAT_P_DECIMAL gépadat beállításának megfelelően: • egy <m> + <n> jegyű egészszámba **vagy** • egy decimális számba maximum <m> tizedes vessző előtti és pontosan <n> tizedes vessző utáni hellyel. Mint a %.<n>P formátum-leírásnál, a teljes string az MD10750 $MN_SPRINT_FORMAT_P_CODE által megadott karakter kódban lesz eltárolva. **Átalakítás MD10751 = 0 esetén:** A REAL érték egy egy <m> + <n> jegyű egész-számba lesz átalakítva. A tizedesjegyek esetleg <n> jegyre lesznek kerekítve vagy 0-val feltöltve. A hiányzó helyek üres jelekkel lesznek feltöltve. A mínusz előjel balra igazítva lesz hozzáadva, a plusz előjel helyén egy üres jel lesz elhelyezve. **Példa:** N10 DEF REAL REAL_VAR=-123.45 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("PUNCHED TAPE FORMAT:%5.3P",REAL_VAR) eredmény: A RESULT string-változó a "PUNCHED TAPE FORMAT:-xx123450" karakterlánccal van leírva ("x" a példában az üres helyeket jelenti). **Átalakítás MD10751 = 1 esetén:** A REAL érték egy egy stringben egy tizedesszámmá lesz átalakítva maximum <m> tizedesvessző előtti jeggyel és pontosan <n> tizedesvessző utáni jeggyel. Esetleg a tizedesvessző előtti jegyek le lesznek vágva és a tizedesvessző utáni jegyek kerekítve lesznek vagy 0-val feltöltve. Ha <n> 0, nincs tizedesvessző. **Példa:** 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) eredmény: A RESULT string-változó a "PUNCHED TAPE FORMAT:-123.450 VAR2:23" karakterlánccal van leírva.
%S:	Egy string beszúrása. **Példa:** N10 DEF STRING[16] STRING_VAR="ABCDEFG" N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF STRING_VAR:%S",STRING_VAR) eredmény: A RESULT string-változó a "CONTENT OF STRING_VAR:ABCDEFG" karakterlánccal van leírva.
%<m>S:	Egy legalább <m> karakteres string beszúrása. A hiányzó helyek üres jelekkel lesznek feltöltve. **Példa:** N10 DEF STRING[16] STRING_VAR="ABCDEFG" N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF STRING_VAR:%10S",STRING_VAR) eredmény: A RESULT string-változó a "CONTENT OF STRING_VAR:xxxABCDEFG" karakterlánccal van leírva ("x" a példában az üres helyeket jelenti).

%.<n>S:	Egy string <n> karakterének beszúrása (az első karakterrel kezdve). **Példa:** N10 DEF STRING[16] STRING_VAR="ABCDEFG" N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF STRING_VAR:%.3S",STRING_VAR) eredmény: A RESULT string-változó a "CONTENT OF STRING_VAR:ABC" karakterlánccal van leírva.
%<m>.<n>S:	Egy string <n> karakterének beszúrása (az első karakterrel kezdve). A létrehozott string összhossza legalább <m> karakter. A hiányzó helyek üres jelekkel lesznek feltöltve. **Példa:** N10 DEF STRING[16] STRING_VAR="ABCDEFG" N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF STRING_VAR:%10.5S", STRING_VAR) eredmény: A RESULT string-változó a "CONTENT OF STRING_VAR:xxxxxABCDE" karakterlánccal van leírva ("x" a példában az üres helyeket jelenti).
%X:	Egy INTEGER érték átalakítása egy hexadecimális ábrázolású string-be. **Példa:** N10 DEF INT INT_VAR='HA5B8' N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("INTEGER HEXADECIMAL:%X",INT_VAR) eredmény: A RESULT string-változó a "INTEGER HEXADECIMAL:A5B8" karakterlánccal van leírva.

Megjegyzés

Az NC nyelvnek tulajdonsága, miszerint a jelölők és kulcsszavak vonatkozásában nem tesz különbséget a nagy- és kisbetűk között, a formátum-leírásokra is érvényes. Ezáltal működési eltérés nélkül lehet programozni nagy- és kisbetűkkel.

Kombinációs lehetőségek

A következő táblázat felvilágosítást ad, hogy melyik NC adattípus melyik formátum-leírással kombinálható. Az implicit adattípus-átalakítás szabályai érvényesek (lásd "Adattípusok (Oldal 57)").

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

Megjegyzés

A táblázat mutatja, hogy az AXIS és FRAME adattípusokat nem lehet közvetlenül használni a SPRINT funkcióval. Azonban lehetséges:

- az AXIS adattípust az AXSTRING művelettel egy stringgé átalakítani, amit a SPRINT-tel tovább fel lehet dolgozni.
- a FRAME adattípus egyes értékeit frame komponens hozzáféréssel olvasni. Ezzel egy REAL típusú adat keletkezik, amit a SPRINT-tel tovább fel lehet dolgozni.

2.10 Program ugrások és elágazások

2.10.1 Visszaugrás a programkezdetre (GOTOS)

A GOTOS utasítással lehetséges a fő- és alprogramok kezdetére visszaugrani a programismétlés céljából.

Gépadattal be lehet állítani, hogy a programkezdetre visszaugrásnál:

- a program-futásidő "0"-ra legyen állítva
- a munkadarab-számláló "1"-gyel növelve legyen.

Szintaxis

GOTOS

Jelentés

GOTOS:	ugrás utasítás programkezdet ugráscéllal	
	A végrehajtást az NC/PLC interfészjel vezérli: DB21, ... DBX384.0 (program-elágazást vezérelni)	
	Érték:	Jelentés:
	0	Nincs visszaugrás a programkezdetre. A program végrehajtása a következő programmondatban a GOTOS után lesz folytatva..
	1	Visszaugrás a programkezdetre. A program ismételve lesz.

Peremfeltételek

- GOTOS kivált egy STOPRE-t (előrefutás-állj).
- Egy munkadarabprogramban adat-definíciókkal (LUD változók) a GOTOS-sal az ugrás a definíciós szakasz utáni első programmondatra történik, vagyis az adat-definíciók nem lesznek újra végrehajtva. A definiált változók viszont megtartják a GOTOS mondatban elért értéket és nem lesznek visszaállítva a definíciós szakaszban programozott alapértékükre.
- A szinkron-akciókban és a technológiai ciklusokban a GOTOS utasítás nem áll rendelkezésre.

Példa

Programkód	Kommentár
N10 ...	; Programkezdet.
...	
N90 GOTOS	; Ugrás a programkezdetre.
...	

2.10.2 Programugrások ugrás-jelölőkre (GOTOB, GOTOF, GOTO, GOTOC)

Egy programban be lehet állítani ugrás-jelölőket (címkék), amelyekre ugyanannak a programnak más helyeiről a `GOTOF`, `GOTOB`, `GOTO` ill. `GOTOC` utasításokkal rá lehet ugrani. A program végrehajtása az ugrás-jelölő utáni utasítással folytatódik. Ezzel a programon belül elágazások valósíthatók meg.

Az ugrás-jelölők mellet ugráscélként lehetségesek fő- és mellémondat számok is.

Ha az ugrás utasítás előtt egy ugrás-feltétel (`IF ...`) van megfogalmazva, akkor a programugrás csak a feltétel teljesülése esetén történik meg.

Szintaxis

```
GOTOB <ugráscél>
IF <ugrásfeltétel> == TRUE GOTOB <ugráscél>

GOTOF <ugráscél>
IF <ugrásfeltétel> == TRUE GOTOF <ugráscél>

GOTO <ugráscél>
IF <ugrásfeltétel> == TRUE GOTO <ugráscél>

GOTOC <ugráscél>
IF <ugrásfeltétel> == TRUE GOTOC <ugráscél>
```

Jelentés

`GOTOB`:	ugrás utasítás programkezdet irányába ugráscéllal	
`GOTOF`:	ugrás utasítás programvég irányába ugráscéllal	
`GOTO`:	ugrás utasítás ugráscél kereséssel A keresés először a programvég irányába, majd a programkezdet irányába történik.	
`GOTOC`:	Úgy hat, mint a GOTO, különbség: a 14080 "Ugráscélt nem talált" vészjelzés el lesz nyomva. Ez azt jelenti, hogy a program végrehajtása egy eredménytelen ugráscél-keresés esetén nem szakad meg, hanem a GOTOC utasítás után következő programsorban folytatódik.	
`<ugráscél>`:	ugráscél-paraméter lehetséges megadások:	
	<ugrás-jelölő>:	Ugráscél a programban beállított ugrás-jelölő a felhasználó által definiált névvel: `<ugrás-jelölő>`:
	<mondatszám>:	ugráscél egy fő- vagy mellékmondat-szám (pl.: `200`, `N300`)
	STRING típusú változó:	változó ugráscél A változó egy ugráscél vagy egy mondatszám helyett van.
`IF`:	kulcsszó az ugrásfeltétel megadásához Az ugrásfeltétel megengedi az összes összehasonlító és logikai műveletet (eredmény: TRUE vagy FALSE). A programugrás akkor lesz végrehajtva, ha ezen művelet eredménye TRUE.	

Megjegyzés

Ugrás-jelölők (címkék)

Az ugrás-jelölők mindig a mondat elején állnak. Ha van programszám, az ugrás-jelölő közvetlenül a mondatszám után áll.

Az ugrás-jelölők elnevezésére a következő szabályok érvényesek:

- Karakterek száma:
  - legalább 2
  - maximum 32
- Megengedett karakterek:
  - betűk
  - számjegyek
  - alsóvonalak
- Az első két jelnek betűnek vagy aláhúzásnak kell lennie.
- Az ugrás-jelölő neve után egy kettőspont (":") következik.

Peremfeltételek

- Az ugráscél csak egy olyan mondat lehet ugrás-jelölővel vagy mondatszámmal,amelyik a programon **belül** van.
- Egy ugrás-utasítást ugrásfeltétel nélkül egy külön mondatban kell programozni. Az ugrás-utasításoknál ugrásfeltétellel ez a korlátozás nem érvényes. Itt meg lehet adni több ugrásfeltételt egy mondatban.
- Az ugrásfeltétel nélküli ugrás-utasításos programokban az `M2`/`M30` programvég utasítások nem kell feltétlenül a program végén legyenek..

Példák

Példa 1: Ugrások ugrás-jelölőkre

Programkód	Kommentár
N10 …	
N20 GOTOF Label_1	; Ugrás a programvég irányába a
	; "Label_1" ugrás-jelölőhöz.
N30 …	
N40 Label_0: R1=R2+R3	; "Label_0" ugrás-jelölő beállítva.
N50 …	
N60 Label_1:	; "Label_1" ugrás-jelölő beállítva.
N70 …	
N80 GOTOB Label_0	; Ugrás program kezdet irányába a
	; "Label_0" ugrás-jelölőhöz.
N90 …	

Példa 2: Közvetett ugrás mondatszámra

```
Programkód                          Kommentár
IF <feltétel> == TRUE
   R10=100                          ; ugráscél hozzárendelés
ELSE
   R10=110                          ; ugráscél hozzárendelés
ENDIF
; Ugrás a programvég irányában a mondatra, amelynek a mondatszáma az R10-ben van.
N10 GOTOF "N"<<R10
...
N90 …
N100 ...                            ; ugráscél
N110 ...
...
```

Példa 3: Ugrás változó ugráscélra

```
Programkód                          Kommentár
DEF STRING[20] ZIEL
IF <feltétel> == TRUE
   ZIEL = "Marke1"                  ; ugráscél hozzárendelés
ELSE
   ZIEL = "Marke2"                  ; ugráscél hozzárendelés
ENDIF
; Ugrás a programvég irányába a "ZIEL tartalma" változó ugráscélhoz
GOTOF ZIEL
Marke1: T="Bohrer1"                 ; ugráscél 1
...
Marke2: T="Bohrer2"                 ; ugráscél 2
...
```

Példa 4: Ugrás ugrásfeltétellel

```
Programkód                                          Kommentár
N40 R1=30 R2=60 R3=10 R4=11 R5=50 R6=20             ; kezdő értékek hozzárendelése
N41 LA1: G0 X=R2*COS(R1)+R5 Y=R2*SIN(R1)+R6         ; ugrás-jelölő LA1
N42 R1=R1+R3 R4=R4-1
IF <ugrásfeltétel> == TRUE
; THEN ugrás a programkezdet irányába az LA1 ugrás-jelölőhöz
N43 IF R4>0 GOTOB LA1
N44 M30                                             ; programvég
```

2.10.3 Program-elágazás (CASE ... OF ... DEFAULT ...)

A CASE utasítás lehetővé teszi egy változó (INT típus) vagy egy számítási funkció aktuális értékét megvizsgálni és az eredménytől függően a program különböző helyeire ugrani.

Szintaxis

```
CASE(<kifejezés>) OF <állandó_1> GOTOF <ugráscél_1> <állandó_2>
GOTOF <ugráscél_2> ... DEFAULT GOTOF <ugráscél_n>
```

Jelentés

CASE:	ugrás utasítás	
<kifejezés>:	változó vagy számítási funkció	
OF:	kulcsszó a feltételes programelágazások megadásához	
<állandó_1>:	első megadott állandó érték a változónak vagy számítási funkciónak	
	típus:	INT
<állandó_2>:	második megadott állandó érték a változónak vagy számítási funkciónak	
	típus:	INT
DEFAULT:	Azokra az esetekre, amelyekben a változó vagy a számítási funkció a megadott állandó értékek egyikét sem veszi fel, a DEFAULT utasítással egy ugráscélt lehet megadni. **Utalás:** Ha nincs programozva DEFAULT utasítás , ezekben az esetekben a CASE utasítást követő mondat lesz az ugráscél.	
GOTOF:	ugrás utasítás programvég irányába ugráscéllal A GOTOF helyett az összes többi GOTO utasítás is programozható (lásd "Programugrások ugrás-jelölőkre" témát).	
<ugráscél_1>:	Elágazás erre az ugráscélra, ha a változó vagy a számítási funkció értéke megfelel az első megadott állandónak. Az ugráscélt a következők szerint lehet megadni:	
	<ugrás-jelölő>:	Ugráscél a programban beállított ugrás-jelölő a felhasználó által definiált névvel: <ugrás-jelölő>:
	<mondatszám>:	ugráscél egy fő- vagy mellékmondat-szám (pl.: 200, N300)
	STRING típusú változó:	változó ugráscél A változó egy ugráscél vagy egy mondatszám helyett van.
<ugráscél_2>:	Elágazás erre az ugráscélra, ha a változó vagy a számítási funkció értéke megfelel a második megadott állandónak.	
<ugráscél_n>:	Elágazás erre az ugráscélra, ha a változó vagy a számítási funkció értéke az egyik megadott állandónak sem felel meg.	

Példa

Programkód

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

Programkód

```
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
...
```

A `CASE` utasítás az `N30`-ból a következő program-elágazási lehetőségeket definiálja:

1. Ha a számítási funkció VAR1+VAR2-VAR3 = 7, akkor ugorjon a mondathoz a "Label_1" ugrás-jelölő definícióval (→ `N40`).
2. Ha a számítási funkció VAR1+VAR2-VAR3 = 9, akkor ugorjon a mondathoz a "Label_2" ugrás-jelölő definícióval (→ `N50`).
3. Ha a számítási funkció VAR1+VAR2-VAR3 értéke sem 7, sem 9, akkor ugorjon a mondathoz a "Label_3" ugrás-jelölő definícióval (→ `N60`).

2.11 Programrész ismétlése (REPEAT, REPEATB, ENDLABEL, P)

A programrész ismétlés lehetővé teszi a már megírt programrészek ismétlését egy programon belül tetszőleges összetételben.

Az ismétlendő programrészek ill. program-tartományok ugrás-jelölőkkel (címkék) lesznek megjelölve.

Megjegyzés

Ugrás-jelölők (címkék)

Az ugrás-jelölők mindig a mondat elején állnak. Ha van programszám, az ugrás-jelölő közvetlenül a mondatszám után áll.

Az ugrás-jelölők elnevezésére a következő szabályok érvényesek:

- Karakterek száma:
  - legalább 2
  - maximum 32
- Megengedett karakterek:
  - betűk
  - számjegyek
  - alsóvonalak
- Az első két jelnek betűnek vagy aláhúzásnak kell lennie.
- Az ugrás-jelölő neve után egy kettőspont (":") következik.

Szintaxis

1. Egyes programsorok ismétlése:

```
<ugrás-jelölő>: ...
...
REPEATB <ugrás-jelölő> P=<n>
...
```

2. Program-tartományt az ugrás-jelölő és a REPEAT utasítás között ismételni:

```
<ugrás-jelölő>: ...
...
REPEAT <ugrás-jelölő> P=<n>
...
```

3. Tartományt két ugrás-jelölő között ismételni:

```
<kezdet ugrás-jelölő>: ...
...
< vég ugrás-jelölő>: ...
...
REPEAT <kezdet ugrás-jelölő> <vég ugrás-jelölő> P=<n>
```

```
...
```

Megjegyzés

A REPEAT utasítást a két ugrás-jelelölővel közrefogni nem lehetséges.. Ha <kezdet ugrás-jelölő> a REPEAT utasítás előtt van és a <vég ugrás-jelölő> a REPEAT utasítás előtt nem lesz elérve, akkor az ismétlés a <kezdet ugrás-jelölő> és a REPEAT utasítás között lesz végrehajtva.

4. Tartományt az ugrás-jelölő és az ENDLABEL között ismételni:

```
<ugrás-jelölő>: ...
...
ENDLABEL: ...
...
REPEAT <ugrás-jelölő> P=<n>
...
```

Megjegyzés

A REPEAT utasítást az <ugrás-jelölő> és az ENDLABEL-lel közrefogni nem lehetséges. Ha az <ugrás-jelölő> a REPEAT utasítás előtt lett megtalálva és az ENDLABEL a REPEAT utasítás előtt nem lesz elérve, akkor az ismétlés az <ugrás-jelölő> és a REPEAT utasítás között lesz végrehajtva.

Jelentés

REPEATB:	utasítás egy programsor ismétlésére
REPEAT:	utasítás egy program-tartomány ismétlésére
<ugrás-jelölő>:	Az <ugrás-jelölő> jelöli: • az ismételendő programsort (REPEATB-nél) ill.: • az ismételendő program-tartomány kezdetét (REPEAT-nél)
	A <ugrás-jelölő>-vel megjelölt programsor állhat a REPEAT-/REPEATB utasítás előtt vagy után is. A keresés először a programkezdet irányába történik. Ha az ugrás-jelölő ebben az irányban nem található, a keresés a programvég irányába folytatódik. **Kivétel:** Ha a program-tartományt az ugrás-jelölő és a REPEAT utasítás között kell ismételni (lásd a 2. szintakszist), akkor a <ugrás-jelölő>-vel megjelölt programsor **előbb** kell álljon a REPEAT utasításnál, mivel ebben az estben **csak** a programkezdet irányába lesz keresve. Ha a sor az <ugrás-jelölő>-vel további utasításokat tartalmaz, ezek minden ismétlésnél újra végre lesznek hajtva.
ENDLABEL:	kulcsszó, ami egy ismétlendő program-tartomány végét jelöli Ha a sor az ENDLABEL-lel további utasításokat tartalmaz, ezek minden ismétlésnél újra végre lesznek hajtva. Az ENDLABEL-t a programban többször is lehet használni.

P:	cím az ismétlés számának megadására	
<n>:	program ismétlések száma	
	típus:	INT
	Az ismétlendő programrész <n>-szer lesz megismételve. Az utolsó ismétlés után a program a REPEAT-/REPEATB sora utáni sorral lesz folytatva. **Utalás:** Ha nincs P=<n> megadva, az ismételendő programrész pontosan egyszer lesz megismételve.	

Példák

Példa 1: Egyes programsorok ismétlése

```
Programkód                          Kommentár
N10 POSITION1: X10 Y20
N20 POSITION2: CYCLE(0,,9,8)        ; pozíció-ciklus
N30 ...
N40 REPEATB POSITION1 P=5           ; Hajtsd végre az N10 mondatot ötször.
N50 REPEATB POSITION2               ; Hajtsd végre az N20 mondatot egyszer.
N60 ...
N70 M30
```

Példa 2: Program-tartományt az ugrás-jelölő és a REPEAT utasítás között ismételni

```
Programkód                      Kommentár
N5 R10=15
N10 Begin: R10=R10+1            ; szélesség
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            ; Hajtsd végre az N10 ... N70 tartományt négyszer.
N90 Z10
N100 M30
```

Példa 3: Tartományt két ugrás-jelölő között ismételni

```
Programkód                          Kommentár
N5 R10=15
N10 Begin: R10=R10+1                ; szélesség
N20 Z=10-R10
N30 G1 X=R10 F200
N40 Y=R10
N50 X=-R10
N60 Y=-R10
```

```
Programkód                          Kommentár
N70 END: Z=10
N80 Z10
N90 CYCLE(10,20,30)
N100 REPEAT BEGIN END P=3           ; Hajtsd végre az N10 .. N70 tartományt háromszor.
N110 Z10
N120 M30
```

Példa 4: Tartományt az ugrás-jelölő és az ENDLABEL között ismételni

```
Programkód                      Kommentár
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          ; Hajtsd végre az N110 ... N120 tartományt háromszor.
N140 REPEAT BEGIN2 P=2          ; Hajtsd végre az N50 ... N80 tartományt kétszer.
N150 M100
N160 REPEAT BEGIN1 P=2          ; Hajtsd végre az N20 ... N80 tartományt kétszer.
N170 Z10
N180 X0 Y0
N190 M30
```

Példa 5: Marás megmunkálás, furat-pozíciókat különböző technológiákkal megmunkálni

```
Programkód                  Kommentár
N10 ZENTRIERBOHRER()        ; Központozó fúrót becserélni.
N20 POS_1:                  ; furat-pozíció 1
N30 X1 Y1
N40 X2
N50 Y2
N60 X3 Y3
N70 ENDLABEL:
N80 POS_2:                  ; furat-pozíció 2
N90 X10 Y5
N100 X9 Y-5
N110 X3 Y3
N120 ENDLABEL:
```

Programkód	Kommentár
N130 BOHRER()	; Fúrót cserélni és fúró-ciklus.
N140 GEWINDE(6)	; M6-os menetfúrót becserélni és menetfúró-ciklus.
N150 REPEAT POS_1	; Programszakaszt POS_1-től ENDLABEL-ig egyszer ismételni.
N160 BOHRER()	; Fúrót cserélni és fúró-ciklus.
N170 GEWINDE(8)	; M8-as menetfúrót becserélni és menetfúró-ciklus.
N180 REPEAT POS_2	; Programszakaszt POS_2-től ENDLABEL-ig egyszer ismételni.
N190 M30	

További információk

- A programrészek ismétlését lehet egymásba skatulyázva felhívni. Minden felhívás elfoglal egy alprogramszintet.
- Ha egy programrész ismétlése közben `M17` vagy `RET` van programozva, akkor a programrész ismétlés megszakad. A program a `REPEAT`- sor utáni mondattal lesz folytatva.
- Az aktuális program-kijelzőben a programrész ismétlés külön alprogramszintként lesz kijelezve.
- Ha a programrész ismétlés alatt szint-megszakítás lesz kiváltva, akkor a program a programrész feldolgozása hívása után lesz folytatva.
  Példa:

Programkód	Kommentár
N5 R10=15	
N10 BEGIN: R10=R10+1	; szélesség
N20 Z=10-R10	
N30 G1 X=R10 F200	
N40 Y=R10	; szint-megszakítás
N50 X=-R10	
N60 Y=-R10	
N70 END: Z10	
N80 Z10	
N90 CYCLE(10,20,30)	
N100 REPEAT BEGIN END P=3	
N120 Z10	; Program feldolgozását folytatni.
N130 M30	

- A vezérlő struktúrákat és a programrész ismétlést lehet kombináltan használni. Azonban ne legyenek átfedések. Egy programrész ismétlés egy vezérlő struktúra ágon belül, ill. egy vezérlő struktúra egy programrész ismétlésen belül kellene legyen.
- Az ugrások és a programrész ismétlések keverésénél a mondatok sorrendben lesznek feldolgozva. Pl. egy ugrásnál egy programrész ismétlésből a feldolgozás a programozott programrész végéig folytatódik.
  Példa:

Programkód

```
N10 G1 F300 Z-10
N20 BEGIN1:
N30 X=10
N40 Y=10
N50 GOTOF BEGIN2
N60 ENDLABEL:
N70 BEGIN2:
N80 X20
N90 Y30
N100 ENDLABEL: Z10
N110 X0 Y0 Z0
N120 Z-10
N130 REPEAT BEGIN1 P=2
N140 Z10
N150 X0 Y0
N160 M30
```

Megjegyzés

A `REPEAT` utasítás a mozgás-mondatok után álljon.

2.12 Vezérlő struktúrák

A vezérlés az NC-mondatokat alap-beállításként a programozott sorrendben dolgozza fel.

Ezt a sorrendet alternatív programblokkok és programhurkok programozásával lehet variálni. Ezeknek a vezérlő-struktúráknak a programozása az `IF`, `ELSE`, `ENDIF`, `LOOP`, `FOR`, `WHILE` és `REPEAT` kulcsszavakkal történik.

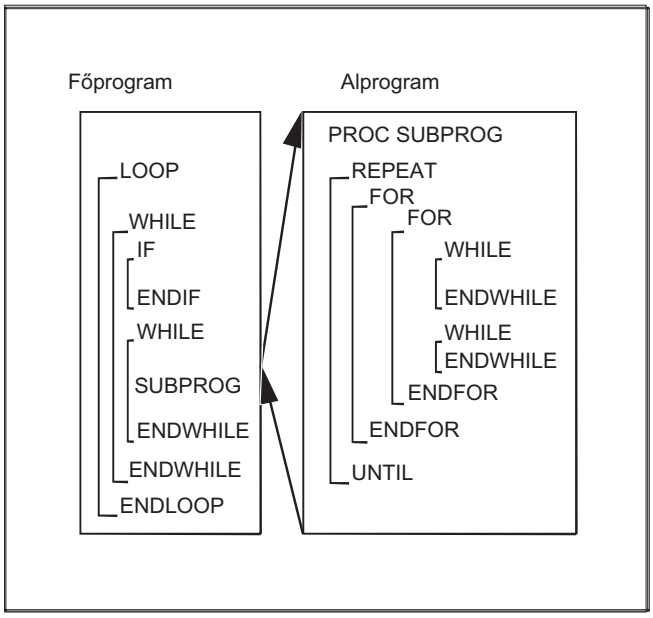
FIGYELEM
Programozási hiba A vezérlő-struktúrák csak egy program utasítás-részén belül lehetségesek. A definíciókat a programfejben nem lehet feltételesen vagy ismételve végrehajtani. A vezérlő-struktúrák kulcsszavait és az ugráscélokat nem szabad makrókkal átfedni. A makró definíciónál nincs ellenőrzés.

Hatásosság

Egy vezérlési szerkezetet nem lehet a programokat átfogóan megváltoztatni.

Skatulyázási mélység

Minden alprogram szinten a vezérlő-struktúrák max. 16-szoros egymásba skatulyázása lehetséges.



Futási idő viselkedés

A szabványosan aktív interpreter-üzemben a programugrások alkalmazásával gyorsabb programlefutást lehet elérni, mint a vezérlő-struktúrákkal.

Az előre lefordított ciklusoknál nincs különbség a programugrások és a vezérlő-struktúrák között.

Aktuális mondat kijelzés a programhurkoknál

Ha egy programhurkon belül csak előrefutás mondatok lesznek végrehajtva, az aktuális mondat kijelzőben az utolsó főmondat a programhurok **előtt** lesz kijelezve.

A feldolgozott előrefutás mondatok kijelzéséhez az aktuális mondat kijelzőben, pl. diagnózis célokra, aktiválni kell az SBL2 egyes mondat dekódolást.

Irodalom

Alap funkciók működési kézikönyv, fejezet: BAG, csatorna, programozás üzem, Reset viselkedés (K1) > egyes mondat> DSBL2 egyes mondat dekódolás közvetett előre-futás álljjal

Hurkok főfutam mondat nélkül

Ha egy programhurkon belül nincs főfutam mondat programozva, a hurok addig lesz az előrefutásban feldolgozva, amíg a hurok feltétel teljesül.

Ezáltal magas terhelés adódik és ez befolyásolhatja a kijelzést.

Segítségként kínálkozik a STOPRE utasítás vagy egy 0 másodperces G04 várakozási idő beszúrása a hurok**ba**.

Peremfeltételek

- A vezérlő-struktúra elemeket tartalmazó mondatok nem kapcsolhatók ki.
- Ugrás-jelölők (címkék) az ilyen mondatokban nem megengedettek.
- A vezérlő-struktúrák feldolgozása interpreter jellegű. A hurok vége felismerésekor történik a már felismert vezérlő-struktúráknak megfelelően a hurok kezdet keresése. Az interpreter üzemben a program blokk-szerkezete nincs teljesen megvizsgálva.
- Alapvetően nem ajánlott a vezérlő-struktúrák és az ugrások vegyes alkalmazása.
- A ciklusok elő-feldolgozásánál meg lehet vizsgálni a vezérlő-struktúrák korrekt skatulyázását.

2.12.1 Feltételes utasítás és elágazás (IF, ELSE, ENDIF)

Feltételes utasítás: IF - programblokk - ENDIF

Egy feltételes utasításnál az `IF` és `ENDIF` közötti programblokk csak akkor lesz végrehajtva, ha a feltétel teljesült.

Elágazás: IF - programblokk_1 - ELSE - programblokk_2 - ENDIF

Egy elágazásnál a két programblokk közül mindig az egyik lesz végrehajtva.

Ha teljesült a feltétel, az `IF` és `ELSE` közötti programblokk_1 lesz végrehajtva.

Ha **nem** teljesült a feltétel, az `ELSE` és `ENDIF` közötti programblokk_2 lesz végrehajtva.

Szintaxis

Feltételes utasítás

```
IF <feltétel>
   programblokk                          ; végrehajtás: <feltétel> == IGAZ esetén
ENDIF
```

Elágazás

```
IF <feltétel>
   Programmblock_1                       ; végrehajtás: <feltétel> == IGAZ esetén
ELSE
   Programmblock_2                       ; végrehajtás: <feltétel> == HAMIS esetén
ENDIF
```

Jelentés

`IF`:	Bevezeti a feltételes utasítást ill. elágazást.
`ELSE`:	Bevezeti az alternatív programblokkot
`ENDIF`:	A feltételes utasítás ill. elágazás végét jelöli.
`<feltétel>`:	Logikai kifejezés, amelynek kiértékelése IGAZ vagy HAMIS.

Példa: Szerszámcserélő alprogram

Programkód	**Kommentár**
`PROC L6`	szerszámcserélő rutin
`N500 DEF INT TNR_AKTUELL`	aktív T-szám változója
`N510 DEF INT TNR_VORWAHL`	előválasztott T-szám változója
	aktuális szerszámot megállapítani
`N520 STOPRE`	
`N530 IF $P_ISTEST`	programteszt üzemben ...
`N540 TNR_AKTUELL = $P_TOOLNO`	.... a program-környezetből az "aktuális" szerszám kiolvasva.
`N550 ELSE`	egyébként ...
`N560 TNR_AKTUELL = $TC_MPP6[9998,1]`	.... az orsóban levő szerszám kiolvasva.
`N570 ENDIF`	
`N580 GETSELT(TNR_VORWAHL)`	előválasztott szerszám T-szám olvasása az orsón
`N590 IF TNR_AKTUELL <> TNR_VORWAHL`	Ha az előválasztott szerszám még nem az aktuális szerszám, akkor ...

Programkód	Kommentár
N600 G0 G40 G60 G90 SUPA X450 Y300 Z300 D0	... szerszámcserepontra menni ...
N610 M206	.... és szerszámcserét végrehajtani.
N620 ENDIF	
N630 M17	

2.12.2 Végtelen programhurok (LOOP, ENDLOOP)

A végtelen hurkot a végtelen programokban alkalmazzuk. A hurok végén mindig visszaugrás történik a hurok elejére.

Szintaxis

```
LOOP
...
ENDLOOP
```

Jelentés

LOOP:	bevezeti a végtelen hurkot
ENDLOOP:	a hurok végét jelöli és egy visszaugrást okoz a hurok elejére

Példa

Programkód
...
LOOP
MSG ("nincs aktív szerszám-vágóél")
M0
STOPRE
ENDLOOP
...

2.12.3 Számláló hurok (FOR ... TO ..., ENDFOR)

A számláló hurkot akkor alkalmazzuk, ha egy munkafolyamatot adott számú átfutással kell ismételni.

Szintaxis

```
FOR <változó> = <kezdőérték> TO <végérték>
```

```
...
ENDFOR
```

Jelentés

FOR:	bevezeti a számláló hurkot	
ENDFOR:	a hurok végét jelöli és egy visszaugrást okoz a hurok elejére, amíg a számlálás a végértéket el nem érte	
<változó>:	számláló változó, amely a kezdő értéktől a végső értékig felfele lesz számlálva, minden lefutásnál "1" értékkel növelve	
	típus	INT vagy REAL **Utalás:** A REAL akkor lesz alkalmazva, ha pl. R-paraméter van programozva egy számláló hurokhoz. Ha a számláló változó REAL típusú, az értéke egészszámú értékre lesz kerekítve.
<kezdőérték>:	számlálás kezdőértéke Feltétel: A kezdőérték kisebb kell legyen a végértéknél.	
<végérték>:	számlálás végértéke	

Példák

Példa 1: INTEGER változó vagy R-paraméter számláló változóként

INTEGER változó számláló változóként:

```
Programkód                          Kommentár
DEF INT iVARIABLE1
R10=R12-R20*R1 R11=6
FOR iVARIABLE1= R10 TO R11          ; számláló változó = INTEGER változó
 R20=R21*R22+R33
ENDFOR
M30
```

R-paraméter számláló változóként:

```
Programkód                          Kommentár
R11=6
FOR R10=R12-R20*R1 TO R11           ; számláló változó = R-paraméter (valós változó)
 R20=R21*R22+R33
ENDFOR
M30
```

Példa 2: Adott számú munkadarab készítése

Programkód	Kommentár
DEF INT STUECKZAHL	; "STUECKZAHL" nevű, INT típusú változó definiálása.
FOR STUECKZAHL = 0 TO 100	; Bevezeti a számláló hurkot. A "STUECKZAHL" változó a "0" kezdőértéktől a "100" végértékig lesz felfelé számolva.
G01 …	
ENDFOR	; Számláló hurok vége.
M30	

2.12.4 Programhurok feltétellel a hurok elején (WHILE, ENDWHILE)

Egy WHILE huroknál a feltétel a hurok elején áll. Amíg a feltétel teljesül, a WHILE hurok le fog futni.

Szintaxis

```
WHILE <feltétel>:
...
ENDWHILE
```

Jelentés

WHILE:	bevezeti a programhurkot
ENDWHILE:	a hurok végét jelöli és egy visszaugrást okoz a hurok elejére
<feltétel>:	feltétel, aminek teljesülnie kell, hogy a WHILE hurok lefusson

Példa

Programkód	Kommentár
...	
WHILE $AA_IW[BOHRACHSE] > -10	; WHILE hurok hívása a következő feltétellel: az aktuális MKR parancsérték a fúró-tengelyre nagyobb kell legyen -10-nél
G1 G91 F250 AX[BOHRACHSE] = -1	
ENDWHILE	
...	

2.12.5 Programhurok feltétellel a hurok végén (REPEAT, UNTIL)

Egy REPEAT huroknál a feltétel a hurok végén áll. A REPEAT hurok egyszer lefut és addig lesz ismételve, amíg a feltétel teljesül.

Szintaxis

```
REPEAT
...
UNTIL <feltétel>
```

Jelentés

REPEAT:	bevezeti a programhurkot
UNTIL:	a hurok végét jelöli és egy visszaugrást okoz a hurok elejére
<feltétel>:	feltétel, aminek teljesülnie kell, hogy a REPEAT hurok ne fusson le többet

Példa

```
Programkód                    Kommentár
...
REPEAT                        ; REPEAT hurok hívása.
...
UNTIL ...                     ; A feltétel teljesülésének vizsgálata.
...
```

2.12.6 Programpélda skatulyázott vezérlő-struktúrákkal

```
Programkód                                    Kommentár
LOOP
   IF NOT $P_SEARCH                           ; IF nincs mondatkeresés
      G1 G90 X0 Z10 F1000
      WHILE $AA_IM[X] <= 100                  ; WHILE (X tengely parancsérték <=
                                              100)
         G1 G91 X10 F500                      ; furatkép
         Z-5 F100
         Z5
      ENDWHILE
   ELSE                                       ; ELSE mondatkeresés
      MSG("Keresés közben nincs fúrás")
   ENDIF                                      ; ENDIF
   $A_OUT[1] = 1                              ; következő fúrólap
   G4 F2
ENDLOOP
M30
```

2.13 Koordinációs utasítások (INIT, START, WAITM, WAITMC, WAITE, SETM, CLEARM)

Egy csatorna az NC-ben elvileg a benne elindított programot az üzemmód-csoportjának (BAG) többi csatornájától függetlenül végre tudja hajtani. Ha azonban a BAG több csatornájában egyidejűleg több program részt vesz egy munkadarab elkészítésében, a program lefutásokat a következő koordinálási utasításokkal kell a különböző csatornákban koordinálni.

Előfeltétel

A program koordinálásban érintett összes csatorna **azonos** üzemmód-csoporthoz (BAG) kell tartozzon.

MD10010 $MC_ASSIGN_CHAN_TO_MODE_GROUP[<csatorna>] = <BAG szám>

Csatornanév csatornaszám helyett

A program koordinálás előre definiált eljárásainak paramétereiként a csatornaszám helyett az MD20000 $MC_CHAN_NAME[<csatorna index>]-be beadott csatorna-nevet is lehet használni. A csatorna-nevek használatát az NC programokban először engedélyezni kell:

MD10280 $MN_PROG_FUNCTION_MASK, bit 1 = TRUE

Szintaxis

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

Jelentés

`INIT()`:	Előre definiált eljárás az NC program kiválasztásához, amelyet a megadott csatornában végre kell hajtani.
`START()`:	előre definiált eljárás a mindenkori csatornában kiválasztott program indítására
`WAITM()`:	előre definiált eljárás várakozáshoz egy várakozás jelölő elérésére a megadott csatornákban A saját csatornában a megadott várakozás jelölő a `WAITM`-mel lesz beállítva. Előző mondatot pontos-állj-jal befejezve. A várakozás jelölő szinkronakció után törölve lesz. Egyidejűleg max. 10 jelölőt lehet csatornánként beállítani.
`WAITE()`:	előre definiált eljárás várakozáshoz a programvégre egy vagy több másik csatornában
`WAITMC()`:[1]	előre definiált eljárás várakozáshoz egy várakozás jelölő elérésére a megadott csatornákban A `WAITM`-mel szemben a tengelyek lefékezése pontos-álljra csak akkor lesz elindítva, ha a többi csatorna a várakozás jelölőt még nem érte el.
`SETM()`:[1]	előre definiált eljárás egy vagy több várakozás jelölő beállításához csatorna-koordinálásra. A feldolgozást a saját csatornában ez nem befolyásolja. `SETM` megtartja az érvényességét a csatorna Reset és az NC Start után is.

CLEARM():[1]	előre definiált eljárás egy vagy több várakozás jelölő törlésére csatorna-koordinálásnál A feldolgozást a saját csatornában ez nem befolyásolja. CLEARM() törli az összes várakozás jelölőt a csatornában. CLEARM(0) csak a "0" várakozás jelölőt törli. CLEARM megtartja az érvényességét a csatorna Reset és az NC Start után is.		
<ChanNr>:	csatornaszám A saját csatorna számát nem kell megadni.		
	típus:	INT	
<Prog>:	abszolút vagy relatív ág-megadás (**opcionális**) + programok		
	típus:	STRING	
	ág-megadáshoz lásd: **Irodalom** Programozási kézikönyv Munka előkészítés; "Fájl és program kezelés" > "Programtároló" > "Programtároló fájljainak címzése" (Oldal 215)		
<AckMode>:	nyugtázási módus (**opcionális**)		
	típus:	CHAR	
	érték-kek:	"N"	nyugtázás nélkül A program megmunkálás az utasítás kiküldése után folytatódik.. A küldő nem lesz értesítve akkor, ha az utasítás sikeresen nem hajtható végre.
		"S"	szinkron nyugtázás A programfeldolgozás az mindaddig megállítódik, amíg a vevő-komponens az utasítást nem nyugtázta. Pozitív nyugtázásnál a következő utasítás feldolgozása történik Negatív nyugtázásnál egy hibajelentés lesz kiadva.
<MarkNr>:	várakozásjelző száma **Utalás** Egy több-csatornás rendszerben maximum 100 várakozás jelölő áll rendelkezésre (várakozás jelölő 0 ... 99). Egy egy-csatornás rendszerben csak a 0 várakozás jelölő áll rendelkezésre.		
1) A felhasználó-specifikus kommunikációhoz és / vagy a csatornák koordinálásához lehet használni jelölőket SETM / CLEARM segítségével a WAITMC feltételes várakozás utasítás alkalmazása nélkül is. A várakozás jelölők ennél megtartják az értékeiket a csatorna Reset-en és az NC Start-on túl.			

Példák

START csatornanévvel az MD20000-ből

- Paraméterezés

```
MD10280 $MN_PROG_FUNCTION_MASK, bit 1 = TRUE
$MC_CHAN_NAME[ 0 ] = "MEGMUNKÁLÁS"  ; csatorna 1 neve
$MC_CHAN_NAME[ 1 ] = "TÁPLÁLÁS"  ; csatorna 2 neve
```

- Programozás

Programkód	Kommentár
START(MEGMUNKÁLÁS)	; csatorna 1 indítása
STERT(ELLÁTÁS)	; csatorna 2 indítása

START lokális "csatornanevek" és felhasználói változók által

Programkód	Kommentár
DEF INT MASCHINE = 1	; felhasználói változó definíció csatorna 1-re
DEF INT LADER = 2	; felhasználói változó definíció csatorna 2-re
...	
START(MASCHINE)	; csatorna 1 indítása
START(LADER)	; csatorna 2 indítása

START lokális "csatornanevek", felhasználói változók és paraméterezett csatornanevek által

Programkód	Kommentár
DEF INT chanNo1	; felhasználói változó definíció csatorna 1-re
DEF INT chanNo2	; felhasználói változó definíció csatorna 2-re
chanNo1 = CHAN_1	; paraméterezett csatornanév hozzárendelés csatorna 1
chanNo2 = CHAN_2	; paraméterezett csatornanév hozzárendelés csatorna 2
...	
START(chanNo1)	; csatorna 1 indítása
START(chanNo2)	; csatorna 2 indítása

INIT utasítás abszolút ág-megadással

/_N_MPF_DIR/_N_ABSPAN1_MPF program választása csatorna 2-ben

Programkód

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

INIT utasítás programnevekkel

; Szerszám kiválasztása "MYPROG" névvel. A vezérlés a programot a keresési ág alapján keresi.

Programkód

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

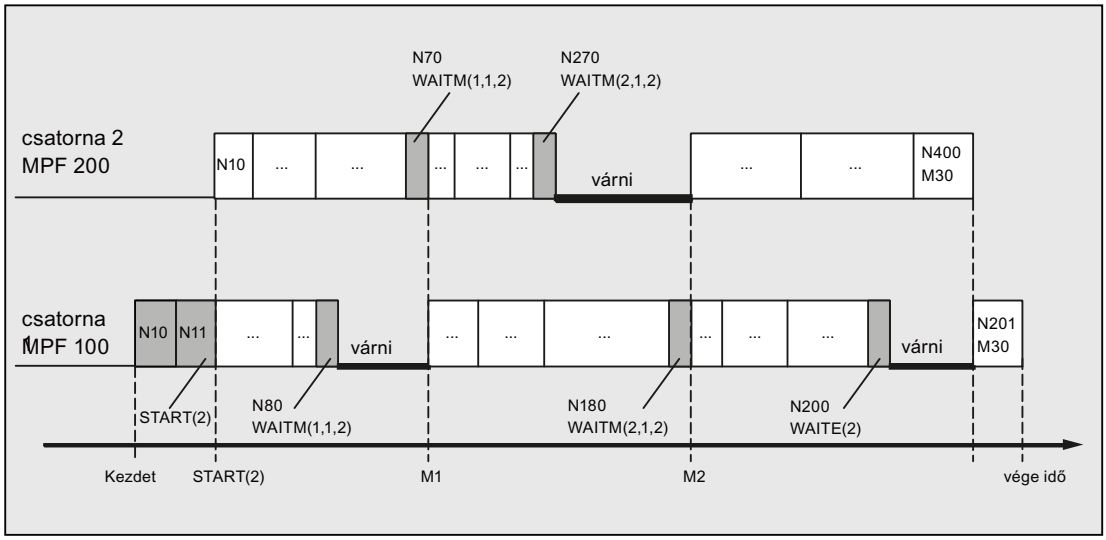
Program koordinálás WAITM-mel

- csatorna 1: Az /_N_MPF_DIR/_N_MPF100_MPF már program ki van választva és elindítva.

Programkód	Kommentár
	; MPF100 program
N10 INIT(2,"MPF200","N")	; MPF200 program kiválasztása, csatorna 2
N11 START(2)	; csatorna 2 indítása
...	
N80 WAITM(1,1,2)	; várakozás Wait-jelölőre 1-re a csatorna 1 és 2-ben
N81 ...	; csatorna 1, N81 és csatorna 2, N71 ; szinkron fognak kezdeni
...	
N180 WAITM(2,1,2)	; várakozás Wait-jelölőre 2-re a csatorna 1 és 2-ben
N181 ...	; csatorna 1, N181 és csatorna 2, N271 ; szinkron fognak kezdeni
...	
N200 WAITE(2)	; várakozás programvégre a csatorna 2-ben
N201 ...	; N201 kezdés csak az ; az MPF200 programvég után a csatorna 2-ben
N201 M30	; programvég csatorna 1

- csatorna 2: A csatorna 1-ben az N10 és N20 mondattal lesz kiválasztva az MPF200_MPF a csatorna 2-re.

Programkód	Kommentár
;$PATH=/_N_MPF_DIR	; MPF200 program
...	
N70 WAITM(1,1,2)	várakozás Wait-jelölőre 1-re a csatorna 1 és 2-ben
N71 ...	; csatorna 1, N81 és csatorna 2, N71 ; szinkron fognak kezdeni
...	
N270 WAITM(2,1,2)	várakozás Wait-jelölőre 2-re a csatorna 1 és 2-ben
N271 ...	; csatorna 1, N181 és csatorna 2, N271 ; szinkron fognak kezdeni
...	
N400 M30	programvég csatorna 2



Peremfeltételek

Követő mondatok feldolgozásának nem szinkron kezdete a WAIT jelölők után

A WAIT jelölőkkel történő csatorna koordinálásnál megtörténhet a követő mondatok feldolgozásának nem szinkron kezdete Ez a viselkedés akkor léphet fel, ha a szinkronizálandó csatornák egyikében közvetlenül a WAIT-jelölő elérése előtt egy olyan akció lesz kiváltva, amelynek következménye egy maradékút törlés közvetett újra-pozicionálással (REPOSA).

Feltétel: aktuális csatorna-hozzárendelés az 1 és 2 csatornákban

- csatorna 1: X1 és U tengelyek
- csatorna 2: X2 tengely

Táblázat 2-2 Időbeni lefutás az 1 és 2 csatornákban

Csatorna 1	Csatorna 2	Leírás
. . .	. . .	tetszőleges megmunkálás a csatorna 1 és 2-ben
`N100 WAITM(20,1,2)`		csatorna 1: eléri a `Wait`-jelölőt és vár a szinkronizáci-óra a csatorna 2-vel
GETD(U) megmunká-lás kezdete: • *tengely-csere* • *maradékút törlés* • *REPOSA* *vége*	`N200 GETD(U)`	csatorna 2: U tengely igénylése a csatorna 1-ből csatorna 1: `GET(U)` feldolgozása a háttérben
	`N210 WAITM(20,1,2)`	csatorna 2: eléri a `WAIT`-jelölőt. ⇒ a csatorna 1 és 2 szinkronizálása ezzel lezárult
	`N220 G0 X2=100`	csatorna 2: N220 feldolgozásának kezdete
`N110 G0 X1=100`		csatorna 1: N110 feldolgozásának idő-eltolt kezdete

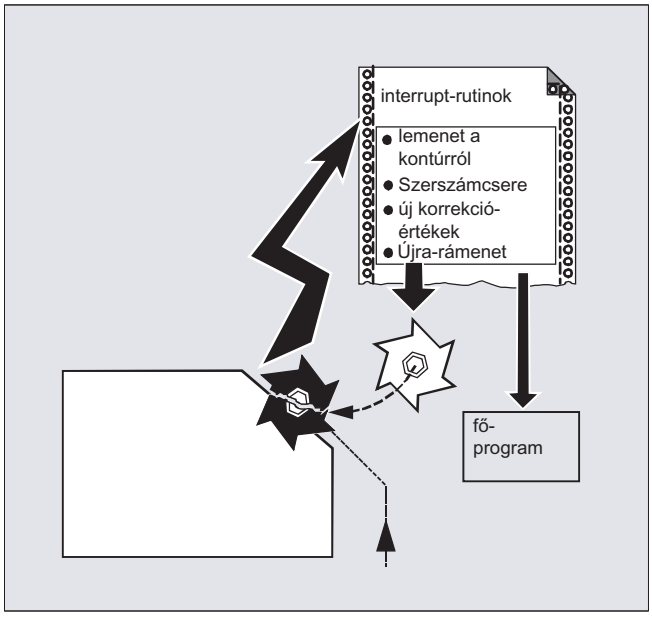
2.14 Interrupt rutinok (ASUP)

2.14.1 Egy interrupt-rutin funkciója

Megjegyzés

A következő leírásban felváltva előforduló "Aszinkron alprogram (ASUP)" és "Interrupt-rutinok" fogalmak ugyanazt a funkcionalitást jelentik.

Egy Interrupt-rutin működését egy tipikus példa magyarázza meg:



A megmunkálás közben eltörik a szerszám. Ezáltal egy jel keletkezik, a mi a futó megmunkálást megállítja és egyidejűleg egy alprogramot - az un. interrupt-rutint - elindítja. Ebben az alprogramban van minden utasítás, amelyeket ebben az esetben végre kell hajtani.

Ha az alprogram végre van hajtva, (és ezáltal az üzemkészség helyreállt), a vezérlés visszaugrik a főprogramba és a megmunkálást - mint a `REPOS` utasítás után - a megszakítási helyen folytatja (lásd " Újra rámenet a kontúrra (Oldal 527) ").

⚠ VIGYÁZAT

Ütközés veszély

Ha az alprogramban nincs `REPOS` utasítás programozva, akkor annak a mondatnak a végpontjára lesz pozícionálás, amelyik a megszakított mondatot követi.

Irodalom

Alap funkciók működési kézikönyv; BAG, csatorna, program üzem, Reset viselkedés (K1),fejezet: "Aszinkron alprogramok (ASUP), Interrupt rutinok"

2.14.2 Interrupt-rutin létrehozása

Interrupt-rutint alprogramként létrehozni

Az interrupt-rutintok a definiálásnál alprogramként lesznek jelölve

Példa:

```
Programkód          Kommentár
PROC ABHEB_Z        ; programnév "ABHEB_Z"
N10 ...             ; Ezután következnek az NC-mondatok.
...
N50 M17             ; Végül programvég és visszatérés a főprogramba.
```

Modális G utasítások mentése (SAVE)

Az interrupt-rutintok a definiálásnál `SAVE`-vel lesznek jelölve.

A `SAVE` jelző hatására a modális G utasítások az interrupt-rutin hívása előtt mentve és az interrupt-rutin vége után reaktiválva lesznek (lásd " Alprogramok SAVE mechanizmussal (SAVE) (Oldal 166) ").

Ezáltal lehetséges a megmunkálást az interrupt-rutin lefutása után a megszakítási helyen folytatni.

Példa:

```
Programkód
PROC ABHEB_Z SAVE
N10 ...
...
N50 M17
```

További interrupt-rutinokat hozzárendelni (SETINT)

Az interrupt-rutinokon belül lehet `SETINT` utasításokat (lásd "Interrupt-rutint hozzárendelni és indítani (SETINT)" (Oldal 126)) programozni és ezáltal további interrupt-rutinokat élesre kapcsolni.. A kiváltás a bemeneten történik.

Irodalom

További információk az alprogramok létrehozásához, lásd "Alprogramok, makró technika"

2.14.3 Interrupt-rutint hozzárendelni és indítani (SETINT, PRIO, BLSYNC)

A vezérlésnek több gyors bemenete (bemenet 1 ... 8) van, amelyek interrupt-ot (1 ... 8) váltanak ki. Minden interrupt-hoz hozzá lehet rendelni egy prioritást és egy interrupt-rutint a `SETINT` utasítással. Ha az interrupt egy gyors bemenet bekapcsolásával lett kiváltva, az aktuális feldolgozás a csatornában megszakad és elindul az interrupt-rutin.

Interrup prioritás

Ha egy munkadarabprogramban több bemenethez vannak interrupt-ok hozzárendelve, az interrupt-okhoz különböző prioritásokat kell hozzárendelni.

Egy interrupt-hoz az 1 ... 128 prioritás-értékeket lehet hozzárendelni. A prioritás-érték 1 a legmagasabb, a 128 a legalacsonyabb prioritásnak felel meg.

Szintaxis

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

Jelentés

`SETINT(<n>)`:	Az <n> interrupt-jelhez az NC program (ASUP) <név> lesz hozzárendelve. A hozzárendelt interrupt-rutin el lesz indítva a bemenet <n> == 1 felismerésekor. **Utalás:** Ha egy <n> interrupt-jelhez egy másik interrupt-rutin lesz hozzárendelve, a meglevő hozzárendelés hatástalanná válik.	
`<n>`:	interrupt-jel száma	
	típus:	INT
	értéktartomány:	1 ... 32
`PRIO=`:	interrupt-ok prioritása (opcionális)	
`<érték>`:	(**opcionális**) prioritás-érték	
	típus:	INT
	értéktartomány:	1 ... 128 (1 ⇒ legmagasabb prioritás)
`<NÉV>`:	NC program (ASUP) neve	
`BLSYNC`:	(**opcionális**) `BLSYNC` hatása az interrupt kiváltása utáni várakozás az aktuális mondat végrehajtásáig. Csak ezután lesz az interrupt-rutin végrehajtva.	
`LIFTFAST`:	(**opcionális**) `LIFTFAST` hatása, hogy az interrupt kiváltása után először egy gyors leemelés következik (lásd "Gyors leemelés a kontúrról (SETINT LIFTFAST, ALF) (Oldal 130)" fejezet). Csak ezután lesz az interrupt-rutin végrehajtva.	

Peremfeltételek

Interrupt szabályok

1. Minden interrupt-ra, amelyet nem lehet azonnal végrehajtani, vagy amelyik már végrehajtás alatt van, egy további interrupt igény lesz tárolva. Az ezen túli interrupt igények erre az interrupt-ra elvesznek.
2. Ha egy interrupt végrehajtás alatt van és egy magasabb prioritású további interrupt lesz kiváltva, az megszakítja az alacsonyabb prioritású interrupt-ot. A magasabb prioritású interrupt lezárása után az alacsonyabb prioritású interrupt folytatva lesz. Ha a magasabb prioritású interrupt alatt további igények érkeznek az alacsonyabb prioritású interrupt-ra, egy igény tárolva lesz. A továbbiak elvesznek.
3. Ha egy interrupt végrehajtás alatt van és egy magasabb prioritású további interrupt lesz kiváltva, az megszakítja az alacsonyabb prioritású interrupt-ot. A magasabb prioritású interrupt végre lesz hajtva. Ha ismét kiváltásra kerül egy magasabb prioritású interrupt, az aktuális interrupt meg lesz szakítva és a magasabb prioritású lesz végrehajtva. Maximálisan hat aktív interrupt szint lehetséges. Egy aktuálisan feldolgozásban levő interrupt szint és további öt várakozó interrupt szint. Minden aktív interrupt szinthez maximum egy további interrupt igény lesz tárolva. Az összes többi interrupt igény elveszik. Szintén elvesznek az interrupt igények, ha azok további interrupt szinteket (interrupt szinteket ≥ 7) igényelnek.

Példák

Példa 1: Interrupt rutint hozzárendelni és prioritást megadni

Programkód	Kommentár
`N20 SETINT(3) PRIO=1 ABHEB_Z`	`; IF bemenet 3 == 1`
	`; THEN interrupt-rutin "ABHEB_Z" indítása`
`N30 SETINT(2) PRIO=2 ABHEB_X`	`; IF bemenet 2 == 1`
	`; THEN interrupt-rutin "ABHEB_X" indítása`

Az interrupt-rutinok a prioritási értékek sorrendjében egymás után lesznek végrehajtva, ha a bemenetek egyidejűleg fennállnak: először "ABHEB_Z", utána "ABHEB_X".

Példa 2: Interrupt-rutint újra hozzárendelni

Programkód	Kommentár
`N20 SETINT(3) PRIO=2 ABHEB_Z`	`; IF bemenet 3 == 1`
	`; THEN interrupt-rutin "ABHEB_Z" indítása`
`...`	
`N80 SETINT(3) PRIO=1 ABHEB_X`	`; IF bemenet 3 == 1`
	`; THEN interrupt-rutin "ABHEB_X" indítása`

2.14.4 Egy interrupt-rutin hozzárendelését deaktiválni/újra-aktiválni (DISABLE, ENABLE)

Egy `SETINT` utasítást a `DISABLE` deaktiválja és a `ENABLE` újra aktiválja, a bemenet → interrupt-rutin hozzárendelés elveszése nélkül.

Szintaxis

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

Jelentés

`DISABLE(<n>)`:	utasítás: **deaktiválja** a bemenet <n> interrupt-rutin hozzárendelését	
`ENABLE(<n>)`:	utasítás: **újra aktiválja** a bemenet <n> interrupt-rutin hozzárendelését	
`<n>`:	paraméter: interrupt-jel száma	
	típus:	INT
	értéktartomány:	1 ... 32

Példa

Programkód	**Kommentár**
`N20 SETINT(3) PRIO=1 ABHEB_Z`	`; Ha a bemenet 3 kapcsol, akkor`
	`; interrupt-rutin "ABHEB_Z" indítása.`
`...`	
`N90 DISABLE(3)`	`; SETINT utasítás az N20-ból deaktiválva lesz.`
`...`	
`N130 ENABLE(3)`	`; SETINT utasítás az N20-ból újra aktiválva lesz.`
`...`	

2.14.5 Egy interrupt-rutin hozzárendelésének törlése (CLRINT)

Egy `SETINT`-tel definiált interrupt-jel hozzárendelést egy NC programhoz (ASUP) az `CLRINT`-tel lehet törölni.

Szintaxis

```
CLRINT(<n>)
```

Jelentés

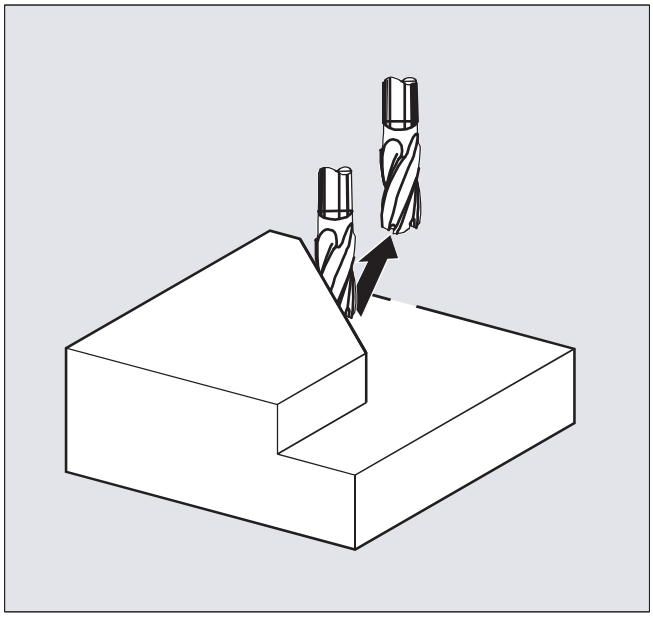
CLRINT(<n>):	utasítás: SETINT-tel definiált interrupt-jel hozzárendelés törlése NC program-hoz (ASUP) <n>	
<n>:	paraméter: interrupt-jel száma	
	típus:	INT
	értéktartomány:	1 ... 32

Példa

Programkód	Kommentár
N20 SETINT(3) PRIO=2 ABHEB_Z	
...	
N50 CLRINT(3)	; A hozzárendelés a bemenet "3" és az "AB-HEB_Z" interrupt-rutin között törölve lesz.

2.14.6 Gyors leemelés a kontúrról (SETINT LIFTFAST, ALF)

Egy SETINT utasításnál LIFTFAST-tal egy bemenet kapcsolásánál a szerszám gyors leemeléssel eltávolodik a munkadarab-kontúrtól.



A további lefutás attól függ, hogy a SETINT utasítás a LIFTFAST mellett egy interrupt-rutint is tartalmaz-e:

Interrupt-rutinnal:	**Utána** a gyors leemelésnek lesz az interrupt-rutin végrehajtva.
Interrupt-rutin nélkül:	A megmunkálás a gyors leemelés után vészjelzéssel leáll.

Szintaxis

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

Jelentés

SETINT(<n>):	utasítás: bemenet <n>-hez egy interrupt-rutint rendelni A hozzárendelt interrupt-rutin elindul, ha az <n> bemenet kapcsol.	
<n>:	paraméter: bemenet száma	
	típus:	INT
	értéktartomány:	1 ... 8
PRIO= :	prioritás megadása	
<érték>:	prioritás érték	
	értéktartomány:	1 ... 128
	A prioritás 1 a legmagasabb prioritás.	
<NÉV>:	alprogram (interrupt-rutin) neve, amelyet végre kell hajtani	
LIFTFAST:	utasítás: gyors leemelés a kontúrról	
ALF=… :	utasítás: programozható lemeneti irány (mozgás-mondatban áll) A programozási lehetőségekhez ALF-fal lásd a " Mozgásirány gyors leemelésnél a kontúrról (Oldal 132) " témát	

Peremfeltételek

Viselkedés aktív frame-nél tükrözéssel

A leemelési irány meghatározásánál megvizsgálásra kerül, hogy aktív-e egy frame tükrözéssel. Ebben az esetben a leemelési iránynál az érintőirányra vonatkoztatva a jobb és a bal meg lesz cserélve. A szerszámirány irány-összetevői nem lesznek tükrözve. Ezt a viselkedést a következő gépadat beállítás aktiválja:

MD21202 $MC_LIFTFAST_WITH_MIRROR = TRUE

Példa

Egy eltörött szerszámot automatikusan egy testvér-szerszámmal kell helyettesíteni. A megmunkálás az új szerszámmal folytatódik.

Főprogram:

Főprogram	**Kommentár**
N10 SETINT(1) PRIO=1 W_WECHS LIFTFAST	; Ha az 1-es bemenet kapcsol, a szerszám azonnal gyors leemeléssel (kód Nr. 7 a G41 szerszámsugár-korrekcióhoz) eltávolodik a kontúrtól. Ezután az "W_WECHS" interrupt-rutin kerül feldolgozásra.
N20 G0 Z100 G17 T1 ALF=7 D1	
N30 G0 X-5 Y-22 Z2 M3 S300	
N40 Z-7	

Főprogram	Kommentár
N50 G41 G1 X16 Y16 F200	
N60 Y35	
N70 X53 Y65	
N90 X71.5 Y16	
N100 X16	
N110 G40 G0 Z100 M30	

Alprogram:

Alprogram	Kommentár
PROC W_WECHS SAVE	; alprogram az aktuális üzemállapot tárolásával
N10 G0 Z100 M5	; szerszámcsere pozíció, orsó-stop
N20 T11 M6 D1 G41	; szerszámcsere
N30 REPOSL RMBBL M3	; rámenet újra a kontúrra és visszaugrás a főprogramba (egy mondatban programozva)

2.14.7 Mozgásirány gyors leemelésnél a kontúrról

Visszahúzás mozgás

A visszahúzási mozgások síkját a következő G utasítások határozzák meg:

- LFTXT
  A sík, amelyikben a visszahúzás végre lesz hajtva, a pálya-érintőből és a szerszámirányból lesz meghatározva (alap-beállítás).
- LFWP
  A visszahúzási mozgások síkja az aktív munkasík, amit G17, G18 vagy G19 G utasításokkal kell kiválasztani. A visszahúzó mozgás iránya független a pályaérintőtől. Ezzel programozható egy tengely-párhuzamos gyors leemelés.
- LFPOS
  Visszahúzás a POLFMASK / POLFMLIN által megadott tengelyekkel a POLF által programozott abszolút tengelypozícióra
  ALF-nak nincs hatása a leemelési irányra több tengelynél és több tengelyre lineáris összefüggésben.
  Irodalom:
  Programozási kézikönyv, Alapok; Fejezet: "Gyors-visszahúzás menetvágás közben"

Programozható lemeneti irány (ALF=…)

A visszahúzási mozgás síkjában az ALF-fal az irány 45 fok diszkrét lépésekben lesz programozva.

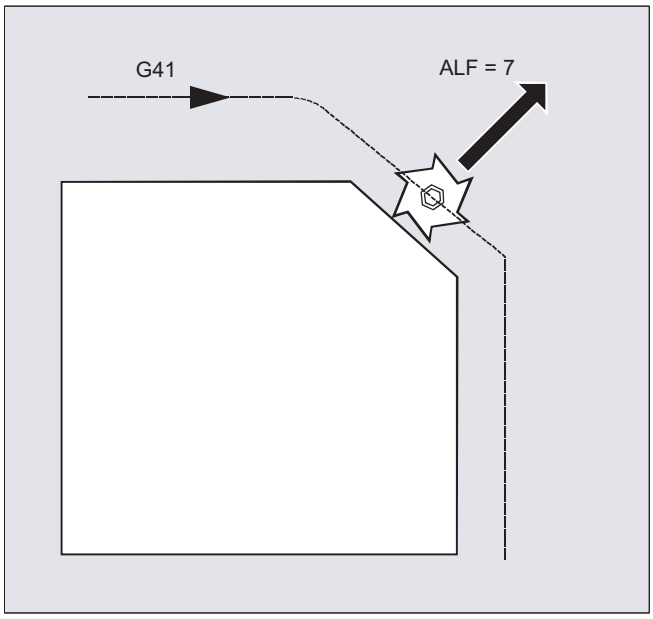
A lehetséges elmozdulási irányok a vezérlésben speciális kódszámokkal vannak tárolva és ezekkel a számokkal felhívhatók

Példa:

Programkód

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

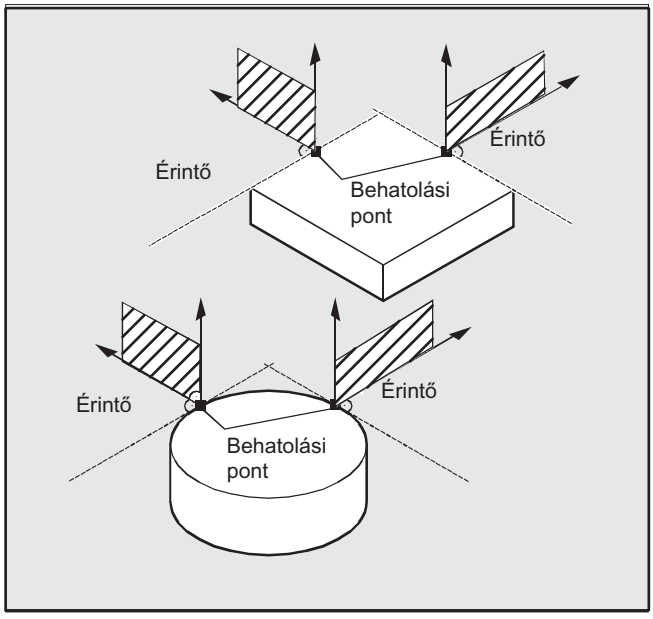
A szerszám bekapcsolt `G41`-nél (megmunkálási irány a kontúrtól balra) merőlegesen eltávolodik a kontúrtól



Vonatkoztatási síkok az elmozgási irányok leírásához LFTXT-nél

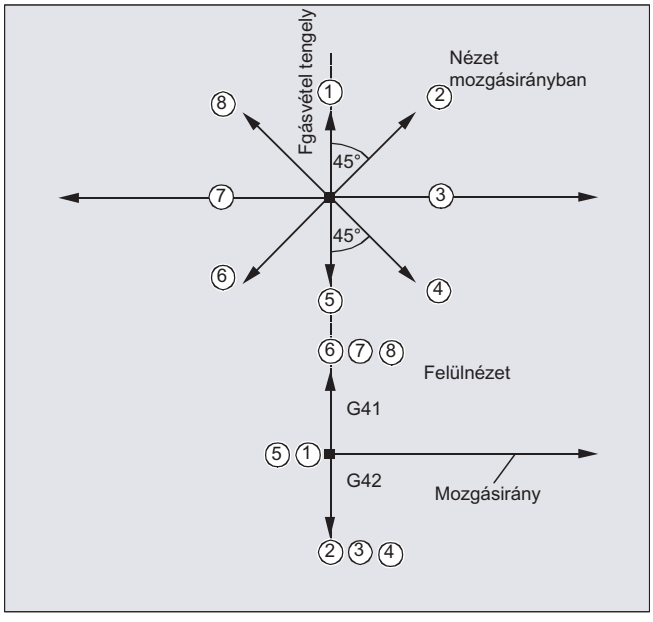
A szerszám érintési pontjánál a programozott kontúrra egy sík lesz felfeszítve, amely vonatkoztatásként szolgál a leemelő mozgás megadásához a megfelelő kódszámmal.

A vonatkoztatási sík a szerszám hossztengelyre (fogásvételi irány) valamint egy erre és a szerszámnak a kontúrt érintő pontja érintőjére merőleges vektorra lesz ráillesztve.



Elmozgási irányok kódszámai LFTXT-nél

A vonatkoztatási síktól kiindulva láthatók a mellékelt ábrán az elmozdulási irányok kódszámai.



Az `ALF=1`-re a visszahúzás szerszámirányban rögzített..

A ALF=0-val a "Gyors leemelés" funkció kikapcsol.

⚠ VIGYÁZAT

Ütközés veszély

Bekapcsolt szerszámsugár-korrekciónál:

- G41-nél a 2, 3, 4 kódszámokat
- G42-nél a 6, 7, 8 kódszámokat

nem szabad alkalmazni mert ezekben az esetekben a szerszám a kontúr felé mozogna és a munkadarabbal ütközne.

Elmozgási irányok kódszámai LFWP-nél

Az LFWP-nél az irány a munkasíkban a következő hozzárendelés szerint adódik:

- G17: X/Y sík
  ALF=1: visszahúzás X irányban
  ALF=3: visszahúzás Y irányban
- G18: Z/X sík
  ALF=1: visszahúzás Z irányban
  ALF=3: visszahúzás X irányban
- G19: Y/Z sík
  ALF=1: visszahúzás Y irányban
  ALF=3: visszahúzás Z irányban

2.14.8 Mozgás-lefutások interrupt-rutinoknál

Interrupt-rutin LIFTFAST nélkül

A tengely-mozgások a pályán a nyugalomig lesznek fékezve Ez után indul az interrupt-rutin.

A nyugalmi pozíció megszakítási pozícióként tárolva lesz és REPOS-nál RMIBL-lel az interrupt-rutin végén fel lesz véve.

Interrupt-rutin LIFTFAST-tal

A tengely-mozgások a pályán le lesznek fékezve. Egyidejűleg a LIFTFAST-mozgás fölérendelt mozgásként lesz végrehajtva. Ha a pályamozgás és a LIFTFAST-mozgás nyugalomba került, az interrupt-rutin el lesz indítva.

Megszakítási pozícióként az a pozíció lesz tárolva, amelyiknél a LIFTFAST-mozgás el lett indítva és ezáltal a pálya el lett hagyva.

Az interrupt-rutin `LIFTFAST`-tal és `ALF=0`-val úgy viselkedik mint az interrupt-rutin `LIFTFAST` nélkül.

Megjegyzés

A távolság, amivel a geometriatengely a gyors leemelésnél a kontúrtól eltávolodik, gépadatban beállítható.

2.15 Tengely-csere, orsó-csere (RELEASE, GET, GETD)

Egy vagy több tengelyt vagy orsót egyszerre csak egy csatornában lehet interpolálni. Ha egy tengely felváltva két különböző csatornában kell dolgozzon (pl. palettacserélő), akkor azt először az aktuális csatornában fel kell szabadítani és azután a másik csatornába átvenni. A tengely a csatornák között kicserélésre kerül.

Tengely-csere bővítménye

Egy tengelyt/orsót előrefutás-állj-jal és a főfutás és előrefutás közötti szinkronizációval vagy alternatívaként előrefutás-állj nélkül is lehet cseréni. Ezenkívül a tengelycsere lehetséges a következőkkel is

- AXCTSWE ill. AXCTWED tengelykonténer forgatás implicit `GET/GETD`-vel.
- frame forgatással, ha ez a tengely más tengelyekkel csatolva van
- szinkronakciók, lásd mozgásszinkron akciók, "tengelycsere `RELEASE`, `GET`".

Gépgyártó

Vegyük ehhez figyelembe a gépgyártó tájékoztatásait. A beállítható gépadatokkal egy tengely a tengelycseréhez minden csatornában egyértelműen definiálva kell legyen és a tengelycsere viselkedése gépadatokkal beállítható is.

Szintaxis

`RELEASE(tengelynév, tengelynév, ...)` vagy `RELEASE(S1)`

`GET(tengelynév, tengelynév, ...)` vagy `GET(S2)`

`GETD(tengelynév, tengelynév, ...)` vagy `GETD(S3)`

A GETD (GET Directly) közvetlenül elhoz egy tengelyt egy másik csatornából. Ez azt jelenti, hogy ehhez a GETD-hez nem kell legyen megfelelő RELEASE programozva egy másik csatornában. De azt is jelenti, hogy most egy másik csatorna-kommunikációt kell felépíteni (pl. várakozás-jelzők).

Jelentés

`RELEASE (tengelynév, tengelynév, …)`:	tengely(ek) felszabadítása
`GET (tengelynév, tengelynév, …)`:	tengely(ek) átvétele
`GETD (tengelynév, tengelynév, …)`:	tengely(ek) közvetlen átvétele
`tengelynév`:	tengely-hozzárendelés a rendszerben: AX1, AX2, … vagy a géptengely-nevek megadása
`RELEASE(S1)`:	S1, S2, …orsók felszabadítása
`GET(S2)`:	S1, S2, …orsók átvétele
`GETD(S3)`:	S1, S2, …orsók közvetlen átvétele

GET igény előrefutás állj nélkül

Ha egy **előrefutás állj nélküli** GET igény után a tengely `RELEASE(tengely)`-lyel vagy `WAITP(tengely)`-lyel ismét engedélyezve lesz, akkor egy következő `GET` egy `GET-et okoz` **előrefutás állj-jal**.

> ⚠ VIGYÁZAT
>
> **Tengely hozzárendelés megváltozott**
>
> Egy GET-tel átvett tengely ill. orsó egy billentyű- vagy program-RESET után is ehhez a csatornához rendelve marad.
>
> Egy új program Startnál a megcserélt tengelyek ill. orsók hozzárendelése program-technikailag kell történjen, ha a tengelyre a saját alapcsatornájában van szükség.
>
> POWER ON-nál a gépadatban megadott csatornához lesz hozzárendelve.

Példák

Példa 1: Tengely-csere két csatorna között

A 6 tengelyből az 1-es csatornában használva vannak: 1., 2., 3. és 4. tengely.
5. és 6. 2-es csatornában a munkadarab-cseréhez vannak felhasználva.

A 2. tengelyt a két csatorna között kell cserélni és POWER ON után az 1-es csatornához kell legyen hozzárendelve.

Program "MAIN" csatorna 1-ben:

Programkód	Kommentár
`INIT (2,"TAUSCH2")`	`; TAUSCH2 programot a 2-es csatornában kiválasztani.`
`N… START (2)`	`; Programot a 2-es csatornában elindítani.`
`N… GET (AX2)`	`; AX2 tengelyt átvenni.`
`...`	
`N… RELEASE (AX2)`	`; AX2 tengelyt felszabadítani.`
`N… WAITM (1,1,2)`	`; Várakozás Wait-jelölőre az 1-es és a 2-es csatornában a szinkronizációhoz mindkét csatornában.`
`...`	`; További lefutás tengelycsere után.`
`N… M30`	

Program "TAUSCH2" csatorna 2-ben:

Programozás	Kommentár
`N… RELEASE (AX2)`	
`N160 WAITM(1,1,2)`	`; Várakozás Wait-jelölőre az 1-es és a 2-es csatornában a szinkronizációhoz mindkét csatornában.`
`N150 GET(AX2)`	`; AX2 tengelyt átvenni.`
`...`	`; További lefutás tengelycsere után.`
`N… M30`	

Példa 2: Tengely-csere szinkronizálás nélkül

Ha a tengelyt nem kell szinkronizálni, a GET nem okoz előrefutás álljt.

Programozás	Kommentár
N01 G0 X0	
N02 RELEASE(AX5)	
N03 G64 X10	
N04 X20	
N05 GET(AX5)	; Ha nem szükséges szinkronizáció, ez nem lesz végrehajtható mondat.
N06 G01 F5000	; Nem végrehajtható mondat.
N07 X20	; Nem végrehajtható mondat, mert az X pozíció azonos az N04-gyel.
N08 X30	; Az első végrehajtható mondat az N05 után.
...	

Példa 3: Tengelycsere aktiválása előrefutás állj nélkül

Előfeltétel: A tengelycserét előrefutás állj nélkül egy gépadattal be kell állítani.

Programozás	Kommentár
N010 M4 S100	
N011 G4 F2	
N020 M5	
N021 SPOS=0	
N022 POS[B]=1	
N023 WAITP(B)	; B tengely semleges tengely lesz.
N030 X1 F10	
N031 X100 F500	
N032 X200	
N040 M3 S500	; Tengely nem vált ki előrefutás álljt / REORG-ot.
N041 G4 F2	
N050 M5	
N099 M30	

Ha a B tengely ill. orsó közvetlenül a az N023 mondat után **PLC tengelyként** pl. 180 fokra és vissza 1 fokra lesz mozgatva, akkor ez a tengely ismét semleges tengellyé válik és az `N40` mondatban nem vált ki előrefutás álljt.

További információ

Előfeltételek a tengely-cseréhez

- A tengely gépadatokkal minden csatornában definiált kell legyen, amelyik használni akarja.
- A **tengely**-specifikus gépadattal meg kell legyen adva, hogy a tengely POWER ON után melyik csatornához legyen hozzárendelve.

Leírás

Tengely felszabadítás: RELEASE

A tengely felszabadításnál figyelembe kell venni:

1. A tengely nem vehet részt transzformációban.
2. Tengely-csatolásoknál (érintő-vezérlés) a szövetség minden tengelye fel kell legyen szabadítva.
3. Konkuráló pozicionáló-tengelyt ebben az állapotban nem lehet cserélni.
4. Gantry mester-tengelynél minden követő-tengely is cserélve lesz.
5. Tengely-csatolásoknál (magával visz, vezetőérték csatolás, elektronikus hajtómű) csak az egyesülés vezetőtengelyét lehet felszabadítani.

Tengelyt átvenni: GET

Ezzel az utasítással lesz a tulajdonképpeni tengelycsere végrehajtva. A felelősség a tengelyért teljesen annál a csatornánál van, amelyben az utasítás programozva lett.

GET hatásai:

Tengely-csere szinkronizálással:

Egy tengelyt mindig akkor kell szinkronizálni, ha időközben egy másik csatornához vagy a PLC-hez volt hozzárendelve, és a GET előtt nem volt szinkronizálás "WAITP"-vel, G74-gyel vagy maradékút törléssel.

- Előre-futás állj következik be (mint STOPRE-nél).
- A megmunkálás addig megszakad, amíg a csere teljesen végre nincs hajtva.

Automatikus "GET"

Ha egy tengely elvileg a csatornában hozzáférhető, de adott időpontban nem "csatorna-tengely", automatikusan egy GET utasítás lesz végrehajtva. Ha a tengely(ek) már szinkronizált(ak), nem keletkezik előrefutás állj.

Tengelycsere viselkedését változtathatóra beállítani

A tengelyek leadási időpontja egy gépadattal a következők szerint beállítható:

- Az automatikus tengelycsere két csatorna között akkor is megtörténik, ha a tengely egy WAITP-vel egy semleges állapotba lett hozva (eddigi viselkedés).
- Egy tengelykonténer-forgatás igénynél a tengelykonténernek az összes, a végrehajtandó csatornához rendelt tengelye implicit GET-tel ill. GETD-vel a csatornába lesz hozva. Ezt követően tengelycsere csak a tengelykonténer-forgatás lezárása után ismét megengedett.

- A főfutamban egy beszúrt mondat után meg lesz vizsgálva, hogy szükséges-e egy reorganizáció. Csak ha ennek a mondatnak a tengelyállapotai az aktuális tengelyállapotokkal **nem** egyeznek, szükséges a reorganizáció.
- Egy GET-mondat helyett előrefutás állj-jal és szinkronizációval az előrefutás és a főfutás között történhet egy tengelycsere előrefutás-állj nélkül is. Ekkor csak egy közbenső mondat lesz létrehozva a GET igénnyel. A főfutásban ennek a mondatnak a végrehajtásánál meg lesz vizsgálva, hogy a tengely állapotai a mondatban az aktuális tengelyállapotokkal megegyeznek-e.

További információk a tengely vagy orsó csere funkciókhoz, lásd
Bővítő funkciók működési kézikönyv; BAG, csatornák, tengely csere (K5).

2.16 Tengelyt egy másik csatornának átadni (AXTOCHAN)

Az AXTOCHAN utasítással fel lehet szólítani egy tengelyt ennek a tengelynek az átadására egy másik csatornának. A tengelyt lehet NC munkadarabprogramból és egy szinkronakcióból is a megfelelő csatornába áthozni.

Szintaxis

```
AXTOCHAN(tengelynév,csatornaszám[,tengelynév,csatornaszám[,...]])
```

Jelentés

Elem	Leírás
AXTOCHAN:	tengelyt egy megadott csatornának igényelni
tengelynév:	tengely-hozzárendelés a rendszerben: X, Y, … vagy az érintett géptengely-nevek megadása A végrehajtandó csatorna nem kell a saját csatorna legyen és nem kell az a csatorna se legyen, amelyik aktuálisan birtokolja az interpolációs jogokat a tengelyhez
csatornaszám:	csatorna száma, amelyhez a tengelyt hozzá kell rendelni

Megjegyzés

Konkuráló pozícionáló-tengelyek és kizárólag PLC vezérelte tengely

Egy PLC-tengely konkuráló pozícionáló-tengelyként nem válthat csatornát. Egy kizárólag PLC vezérelte tengelyt nem lehet az NC-programhoz rendelni.

Irodalom

Bővítő funkciók működési kézikönyv; Pozícionáló tengelyek (P2)

Példa

AXTOCHAN az NC-programban

Az X és Y tengelyek az 1. csatornában és a 2. csatornában ismertek. Aktuálisan a csatorna 1-nek van interpolációs joga és a csatorna 1-ben a következő program lesz elindítva:

Programkód	Kommentár
N110 AXTOCHAN(Y,2)	; Y tengelyt a 2. csatornába tolni.
N111 M0	
N120 AXTOCHAN(Y,1)	; Y-tengelyt ismét visszahozni (semleges).
N121 M0	
N130 AXTOCHAN(Y,2,X,2)	; Y-tengelyt és X-tengelyt a 2. csatornába tolni (tengely semleges).
N131 M0	
N140 AXTOCHAN(Y,2)	; Y tengelyt a 2. csatornába tolni (NC program).
N141 M0	

További információk

AXTOCHAN az NC-programban

Ennél csak a tengely igénylésénél az NC-program számára a saját csatornában lesz egy `GET` végrehajtva és ezzel várakozás lesz a tényleges állapotváltozásra. Ha a tengely egy másik csatorna számára lesz igényelve vagy a saját csatornában semleges tengellyé kellene váljon, akkor az igény megfelelően el lesz utasítva.

AXTOCHAN egy szinkronakcióból

Ha egy tengely a saját csatorna számára van igényelve, akkor az `AXTOCHAN` egy szinkronakcióból le lesz képezve egy `GET` -re egy szinkronakcióból.. Ebben az esetben a tengely az első igénynél a saját tengely számára semleges tengellyé válik. A második igénynél a tengely az NC-programhoz lesz hozzárendelve úgy, mint a GET igénynél az NC-programban. A GET igényhez egy szinkronakcióból lásd a "Mozgásszinkron akciók" fejetetet.

2.17 Gépadatokat hatásossá tenni (NEWCONF)

A NEWCONF utasítással gépadat hatásossá válik. A funkció lehet aktiválni a HMI kezelőfelületen az "MD-t hatásossá tenni" softkey működtetésével.

A "NEWCONF" funkció végrehajtásánál egy implicit előrefutás állj történik, azaz a pályamozgás megszakad.

Szintaxis

```
NEWCONF
```

Jelentés

NEWCONF:	utasítás az összes "NEW_CONFIG" hatásossági fokozatú gépadat hatásossá tételére

NEWCONF csatornákat átfogó végrehajtása a munkadarabprogramból

Ha tengely gépadatok a munkadarabprogramból változtatva, majd után NEWCONF-fal aktiválva lesznek, akkor a NEWCONF csak azokat a gépadatokat aktiválja, amelyek a munkadarabprogram csatornájában változásokat okoznak.

Megjegyzés

Az összes változás hatásossá tétele céljából a NEWCONF utasítást minden csatornában végre kell hajtani, amelyekben a gépadatok által megváltoztatott tengelyek vagy funkciók aktuálisan számításra kerülnek..

A NEWCONF -nál a tengely gépadatok nem lesznek hatásossá téve.

A PLC vezérlésű tengelyekre egy tengely RESET-et kell végrehajtani.

Példa

Maró-megmunkálás: furat-pozíciókat különböző technológiákkal megmunkálni

Programkód	Kommentár
N10 $MA_CONTOUR_TOL[AX]=1.0	; Gépadatot változtatni.
N20 NEWCONF	; Gépadatokat hatásossá tenni.
...	

2.18 Fájl írás (WRITE)

A `WRITE` utasítással mondatokat/adatokat lehet írni az NC programból a passzív fájlrendszerben vagy egy külső programtárolón található fájl (jegyzőkönyv fájl) végére . Ez lehet az éppen feldolgozás alatt levő program is.

Megjegyzés

Egy `WRITE` utasítással írandó fájl újként létre lesz hozva, ha a programtárolóban nem létezik.

Előfeltétel

Az aktuálisan beállított védelmi fokozat nagyobb vagy egyenlő kell legyen a fájl WRITE jogával. Ha ez nem így van, a hozzáférés hibajelzéssel (hibaváltozó visszaadási értéke =13) el lesz utasítva.

Szintaxis

```
DEF INT <hiba>
...
WRITE(<hiba>,"<fájlnév>"/"<ExtG>","<mondat/adatok>")
```

Jelentés

`WRITE`:	utasítás egy mondat ill. adatok hozzáadására a megadott fájl végéhez		
`<hiba>`:	**Paraméter 1:** változó a hibaérték visszaadására		
	típus:	INT	
	érték:	0	nincs hiba
		1	ág nem megengedett
		2	ágat nem találta
		3	fájlt nem találta
		4	helytelen fájl-típus
		10	fájl tele
		11	fájl használatban
		12	nincs szabad erőforrás
		13	nincs hozzáférési jog
		14	hiányzó vagy nem sikeres EXTOPEN a kiadási készülékre
		15	Hiba írásnál a külső készülékre
		16	érvénytelen külső ág van programozva
`<fájlnév>`:	**paraméter 2:** A fájl neve, amelybe a megadott mondatot ill. adatokat be kell illeszteni.		
	típus:	STRING	
	A tényleges fájlnév előtt be lehet adni az abszolút ágat. Ha nincs ág megadva, a fájl az aktuális könyvtárban (= a kiválasztott program könyvtára) lesz keresve. Szabályok az ág-megadáshoz lásd "Programtároló fájljainak címzése (Oldal 215)".		

<ExtG>:	Ha az adatokat a "Process DataShare" funkcióval egy külső készülékre/fájlba kell írni, a fájlnév helyett a megnyitandó külső készülék/fájl szimbolikus jelölőjét kell megadni.	
	típus:	STRING
	További információkat lásd "Process DataShare - Kiadás egy külső készülékre/fájlba (EXTOPEN, WRITE, EXTCLOSE) (Oldal 657)". **Utalás:** A jelölő azonos kell legyen az `EXTOPEN` utasításban megadott jelölővel.	
<mondat/adatok>:	**Paraméter 3:** Mondat ill. adatok,amelyeket a megadott fájlba be kell illeszteni.	
	típus:	STRING

Megjegyzés

A passzív fájlrendszerbe vagy egy külső programtárolóba írásnál a `WRITE` utasítás egy "LF" (LINE FEED = új sor) karaktert illeszt a kiadási string végére.

A kiadásnál egy külső készülékre/fájlba a "Process DataShare" funkcióval ez a viselkedés nem érvényes. Ha ki kell adni egy "LF"-et, akkor azt külön meg kell adni a kiadási string végén.

→ Lásd ehhez a 3. példát: Implicit/explicit "LF"!

Peremfeltételek

- **Maximális fájlméret (→ gépgyártó!)**
  A passzív fájlrendszerben a jegyzőkönyv-fájlok maximális hosszát a következő gépadattal lehet megadni:
  MD11420 $MN_LEN_PROTOCOL_FILE
  A maximális fájlméret minden fájlra érvényes, amelyek a `WRITE` utasítással lesznek létrehozva a passzív fájlrendszerben. A túllépésénél hibajelzés kerül kiadásra és a mondat ill. az adatok nem lesz tárolva. Ha a tároló elegendő, egy új fájlt lehet létrehozni.

Példák

Példa 1: WRITE utasítás a passzív fájlrendszerben abszolút ág-megadás nélkül

Programkód	Kommentár
`N10 DEF INT ERROR`	`; Mező-változók definiálása.`
`N20 WRITE(ERROR,"PROT","PROTOKOLL VOM 7.2.97")`	`; A "PROTOKOLL VOM 7.2.97" szöveg írása a _N_PROT_MPF fájlba.`
`N30 IF ERROR`	`; Hiba kiértékelés.`
`N40 MSG ("Hiba a WRITE utasításnál:" << ERROR)`	
`N50 M0`	
`N60 ENDIF`	
`...`	

Példa 2: WRITE utasítás a passzív fájlrendszerben abszolút ág-megadással

Programkód

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

Példa 3: Implicit/explicit "LF"

a, Írás a passzív fájlrendszerbe implicit létrehozott "LF"-fel

Programkód

```
...
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
```

Kiadási eredmény:

MY_STRING

MY_STRING

b, Írás külső fájlba implicit létrehozott "LF" nélkül

Programkód

```
...
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
```

Kiadási eredmény:

MY_STRINGMY_STRING

c, Írás külső fájlba explicit programozott "LF"-fel

Az a-val azonos eredmény eléréséhez a következőt kell programozni:

Programkód

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

Programkód

```
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
```

Kiadási eredmény:

MY_STRING

MY_STRING

2.19 Fájl törlés (DELETE)

A DELETE utasítással minden fájlt lehet törölni, mindegy, hogy WRITE utasítással lett létrehozva vagy nem. Azokat a fájlokat is lehet DELETE-tel törölni, amelyek magasabb hozzáférési fokozattal lettek létrehozva.

Szintaxis

```
DEF INT <hiba>
DELETE(<hiba>,"<fájlnév>")
```

Jelentés

DELETE:	utasítás a megadott fájl törlésére		
<hiba>:	változó a hibaérték visszaadására		
	típus	INT	
	érték:	0	nincs hiba
		1	ág nem megengedett
		2	ágat nem találta
		3	fájlt nem találta
		4	helytelen fájl-típus
		11	fájl használatban
		12	nincs szabad erőforrás
		20	egyéb hiba
<fájlnév>:	törlendő fájl neve		
	típus:	STRING	
	A tényleges fájlnév előtt be lehet adni az abszolút ágat. Ha nincs ág megadva, a fájl az aktuális könyvtárban (= a kiválasztott program könyvtára) lesz keresve. Szabályok az ág-megadáshoz lásd "Programtároló fájljainak címzése (Oldal 215)".		

Példa

Programkód	Kommentár
N10 DEF INT ERROR	; Mező-változók definiálása.
N15 STOPRE	; Előrefutás-állj.
N20 DELETE(ERROR,"/_N_SPF_DIR/_N_TEST1_SPF")	; TEST1 fájl törlése az alprogram-könyvtárban.
N30 IF ERROR	; Hiba kiértékelés.
N40 MSG("Hiba a DELETE utasításnál:" <<ERROR)	
N50 M0	
N60 ENDIF	

2.20 Sorok olvasása fájlban (READ)

A `READ` utasítás a megadott fájlban olvas egy vagy több sort és az olvasott információt egy STRING típusú mezőben teszi le. Minden olvasott sort egy mezőelemet foglal el ebben a mezőben.

Előfeltétel

Az aktuálisan beállított védelmi fokozat nagyobb vagy egyenlő kell legyen a fájl READ jogával. Ha ez nem így van, a hozzáférés hibajelzéssel (hibaváltozó visszaadási értéke =13) el lesz utasítva.

Szintaxis

```
DEF INT <hiba>
DEF STRING[<string hossz>] <eredmény>[<n>,<m>]
READ(<hiba>,"<fájlnév>",<kezdősor>,<sorok száma>,<eredmény>)
```

Jelentés

`READ`:	utasítás a megadott fájlban sorokat olvasni és ezeket a sorokat elhelyezni egy változó-mezőben		
`<hiba>`:	változó a hibaérték visszaadására (Call-By-Reference paraméter)		
	típus	INT	
	érték:	0	nincs hiba
		1	ág nem megengedett
		2	ágat nem találta
		3	fájlt nem találta
		4	helytelen fájl-típus
		11	fájl használatban
		13	hozzáférési jog nem elegendő
		21	sor nem létezik (`<kezdősor>` vagy `<sorok száma>` paraméter nagyobb a sorok számánál a megadott fájlban
		22	eredmény-változó (`<eredmény>`) mezőhossza túl kicsi
		23	sor-tartomány túl nagy (`<sorok száma>` paraméter olyan nagyra lett választava, hogy a fájl végén túl kellene olvasni)
`<fájlnév>`:	olvasandó fájl neve (Call-By-Value paraméter)		
	típus:	STRING	
	A tényleges fájlnév előtt be lehet adni az abszolút ágat. Ha nincs ág megadva, a fájl az aktuális könyvtárban (= a kiválasztott program könyvtára) lesz keresve. Szabályok az ág-megadáshoz lásd "Programtároló fájljainak címzése (Oldal 215)".		

<kezdősor>:	olvasandó fájl kezdősora (Call-By-Value paraméter)		
	típus:	INT	
	érték:	0	A <sorok száma> paraméterrel megadott számú sor lesz olvasva a fájl vége előttről.
		1 … n	az első olvasandó sor száma
<sorok száma>:	olvasandó sorok száma (Call-By-Value paraméter)		
	típus:	INT	
<eredmény>:	eredmény változó (Call-By-Reference paraméter) változó-mező, amelybe az olvasott szöveg lesz letéve		
	típus:	STRING (max. hossz: 255)	
	Ha a <sorok száma> paraméterben kevesebb sor van megadva, mint az eredmény-változó [<n>,<m>] mezőhossza, akkor a maradék mezőelemek nem változnak. A sor lezárása a "LF" (Line Feed) vagy "CR LF" (Carrige Return Line Feed) vezérlőjelekkel **nem** lesz az eredmény-változóba letéve. A beolvasott sorok levágásra kerülnek, ha a sor hosszabb a definiált string-hossznál. Hibajelzés nincs.		

Megjegyzés

Bináris fájlokat nem lehet beolvasni. A helytelen fájltípus hiba (hiba-változó visszaadási értéke = 4) kerül kiadásra. A következő fájltípusok nem olvashatók: _BIN, _EXE, _OBJ, _LIB, _BOT, _TRC, _ACC, _CYC, _NCK.

Példa

Programkód	**Kommentár**
N10 DEF INT ERROR	; Mező-változók definiálása.
N20 DEF STRING[255] RESULT[5]	; Az eredmény-változók definiálása.
N30 READ(ERROR,"/_N_CST_DIR/_N_TESTFILE_MPF", 1,5,RESULT)	; Fájlnév tartomány- és fájl-jelölővel és ág-megadással.
N40 IF ERROR <>0	; Hiba kiértékelés.
N50 MSG("FEHLER"<<ERROR<<"BEI READ-BEFEHL")	
N60 M0	
N70 ENDIF	
...	

2.21 Egy fájl létezését megvizsgálni (ISFILE)

Az `ISFILE` utasítással megvizsgálhatjuk, hogy egy fájl a programtárolóban létezik-e.

Szintaxis

```
<eredmény>=ISFILE("<fájlnév>")
```

Jelentés

`ISFILE`:	Utasítás egye fájl meglétének ellenőrzésére		
`<fájlnév>`:	A fájl neve, amelynek a létezését meg kell vizsgálni.		
	típus:	STRING	
	A tényleges fájlnév előtt be lehet adni az abszolút ágat. Ha nincs ág megadva, a fájl az aktuális könyvtárban (= a kiválasztott program könyvtára) lesz keresve. Szabályok az ág-megadáshoz lásd "Programtároló fájljainak címzése (Oldal 215)".		
`<eredmény>`:	eredmény-változó a vizsgálat eredményének felvételére		
	típus	BOOL	
	érték:	TRUE	fájl létezik
		FALSE	fájl nem létezik

Példák

Példa 1

Programkód	Kommentár
`N10 DEF BOOL RESULT`	`; Az eredmény-változók definiálása.`
`N20 RESULT=ISFILE("TESTFILE")`	
`N30 IF(RESULT==FALSE)`	
`N40  MSG("FÁJL NEM LÉTEZIK")`	
`N50  M0`	
`N60 ENDIF`	
`...`	

Példa 2

Programkód	Kommentár
`N10 DEF BOOL RESULT`	`; Az eredmény-változók definiálása.`
`N20 RESULT=ISFILE("TESTFILE")`	
`N30 IF(NOT ISFILE("TESTFILE"))`	
`N40  MSG("FÁJL NEM LÉTEZIK")`	
`N50  M0`	
`N60 ENDIF`	
`...`	

2.22 Fájl információkat kiolvasni (FILEDATE, FILETIME, FILESIZE, FILESTAT, FILEINFO)

A `FILEDATE`, `FILETIME`, `FILESIZE`, `FILESTAT` és `FILEINFO` utasításokkal kiolvashatók bizonyos fájl információk, mint az utolsó író hozzáférés dátuma / ideje, aktuális fájlméret, fájl állapot vagy ezen információk összege.

Előfeltétel

Az aktuálisan beállított védelmi fokozat nagyobb vagy egyenlő kell legyen a fölérendelt könyvtárak megtekintési jogával. Ha ez nem így van, a hozzáférés hibajelzéssel (hibaváltozó visszaadási értéke =13) el lesz utasítva.

Szintaxis

```
FILE....(<hiba>,"<fájlnév>",<eredmény>)
```

Jelentés

`FILEDATE`:	Egy fájl utolsó írás hozzáférésének **dátumát** adja		
`FILETIME`:	Egy fájl utolsó írás hozzáférésének **időpontját** adja		
`FILESIZE`:	Egy fájl **aktuális méretét** adja		
`FILESTAT`:	Egy fájlra vonatkozó **jogok** aktuális **állapotát** adja: • olvasás (r: read) • írás (w: write) • végrehajtás (x: execute) • kijelzés (s: show) • törlés (d: delete) **Utalás:** Ezek a védelmi fokok a passzív fájlrendszer speciális tulajdonságai. A külső programtárolóhoz hozzáférésnél a `FILESTAT` csak az alap hozzáférési jogokat adja (77777).		
`FILEINFO`:	Megadja egy fájlra az **információk összegét**, amelyek `FILEDATE`, `FILETIME`, `FILESIZE` és `FILESTAT` által kiolvashatók		
`<hiba>`:	változó a hibaérték visszaadására (Call-By-Reference paraméter)		
	típus	VAR INT	
	érték:	0	nincs hiba
		1	ág nem megengedett
		2	ágat nem találta
		3	fájlt nem találta
		4	helytelen fájl-típus
		13	hozzáférési jog nem elegendő
		22	eredmény-változó (`<eredmény>`) string-hossza túl kicsi

<fájlnév>:	fájl neve, amelyről a fájl-információ(ka)t ki kell olvasni			
	típus:	CHAR[160]		
	A tényleges fájlnév előtt be lehet adni az abszolút ágat. Ha nincs ág megadva, a fájl az aktuális könyvtárban (= a kiválasztott program könyvtára) lesz keresve. Szabályok az ág-megadáshoz lásd "Programtároló fájljainak címzése (Oldal 215)".			
<eredmény>:	eredmény-változó (Call-By-Reference paraméter) változó, amelybe az igényelt fájl-információ el lesz letéve			
	típus:	VAR CHAR[8]	következő-nél	FILEDATE "dd.mm.yy" formátum
		VAR CHAR[8]	következő-nél	FILETIME "hh:mm.ss" formátum
		VAR INT	következő-nél	FILESIZE A fájl mérete bájtban lesz kiadva.
		VAR CHAR[5]	következő-nél	FILESTAT "rwxsd" formátum (r: **r**ead, w: **w**rite, x: e**x**ecute, s: **s**how, d: **d**elete)
		VAR CHAR[32]	következő-nél	FILEINFO formátum: "rwxsd nnnnnnnn dd.mm.yy hh:mm:ss"

Példa

Programkód	**Kommentár**
`N10 DEF INT ERROR`	`; Hiba változók definiálása.`
`N20 STRING[32] RESULT`	`; Az eredmény-változók definiálása.`
`N30 FILEINFO(ERROR,"/_N_MPF_DIR/_N_TESTFILE_MPF",RESULT)`	`; Fájlnév tartomány- és fájljelölővel és ág-megadás.`
`N40 IF ERROR <> 0`	`; hiba kiértékelés`
`N50  MSG("FEHLER"<<ERROR<<"BEI FILEINFO-BEFEHL")`	
`N60  M0`	
`N70 ENDIF`	
`...`	

A példa az RESULT eredmény-változóban pl. a következő eredményt adhatja:

"77777 12345678 26.05.00 13:51:30"

2.23 Felkerekítés (ROUNDUP)

A "ROUNDUP" funkcióval a REAL típusú (törtszámok tizedesponttal) beadási értékek a következő nagyobb egész-számra lehet felkerekíteni.

Szintaxis

```
ROUNDUP(<érték>)
```

Jelentés

ROUNDUP:	utasítás egy beadási érték felkerekítésére
<érték>:	REAL típusú beadási érték

Megjegyzés

Az INTEGER típusú beadási értékek (egészszám) változatlanul lesznek visszaadva.

Példák

Példa 1: Különböző beadási értékek és azok kerekítési eredményei

Példa	Kerekítési eredmény
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

Példa 2: ROUNDUP az NC programban

Programkód

```
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 Alprogram-technika

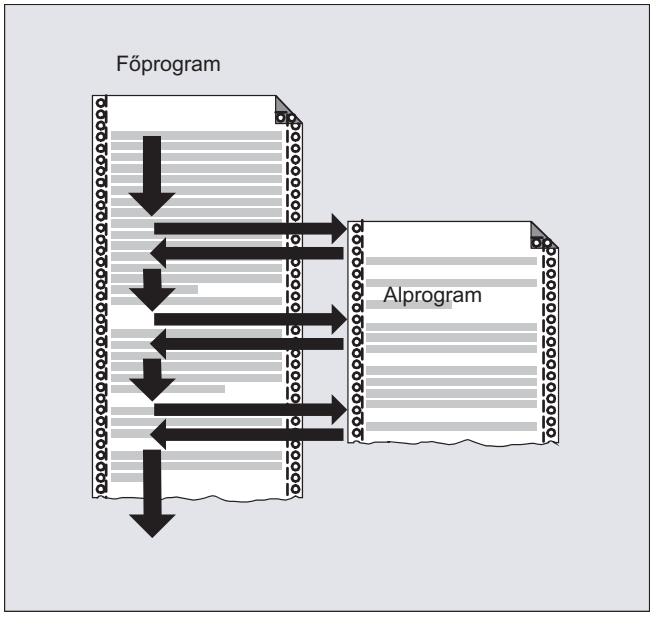
2.24.1 Általános

2.24.1.1 Alprogram

Az "alprogram" megnevezés még abból az időből származik, amikor a munkadarabprogramok fel voltak osztva fő- és alprogramokra. A főprogramok voltak a munkadarabprogramok, amelyeket a vezérlésen kiválasztottak végrehajtásra és utána elindították. Az alprogramok voltak a munkadarabprogramok, amelyeket a főprogram hívott fel.

Ez a fix felosztás a mai SINUMERIK NC-nyelvben már nincs meg. Minden munkadarabprogramot lehet elvileg főprogramként kiválasztani és indítani vagy alprogramként egy másik munkadarabprogramból felhívni.

A továbbiakban egy munkadarabprogramot akkor nevezünk alprogramnak, ha egy másik munkadarabprogramból lesz felhívva.



Alkalmazás

Mint az összes magas szintű nyelvben az NC-nyelvben is az alprogramok a többször használt programrészeknek egy önálló, önmagában zárt programba való kitárolására lesznek használva.

Az alprogramok a következő előnyöket nyújtják:

- Növelik a programok áttekinthetőségét és olvashatóságát.
- Emelik a minőséget a tesztelt programrészek újra alkalmazásával.

- Lehetővé teszik speciális megmunkálási könyvtárak létrehozását.
- Tárolóhelyet takarítanak meg

2.24.1.2 Alprogramnevek

Elnevezési szabályok

Az alprogramnév az alábbi feltételek betartásával szabadon választható:

- Megengedett karakterek:
  - betűk: A ... Z, a ... z
  - számok: 0 ... 9
  - alsóvonal: _
- Az első két karakter két betű vagy egy alsóvonal után egy betű kell legyen.

  Megjegyzés

  Ha ez a feltétel tejesül, egy NC programot a programnév megadásával egy másik programból alprogramként lehet hívni. Ha azonban a programnév számmal kezdődik, akkor az alprogram hívás csak CALL utasítással lehetséges.

- maximális hossz 24 karakter

Megjegyzés
Nagy- / kisbetűk

A SINUMERIK NC nyelvben **nincs** különbség téve a nagy- és a kisbetűk között.

Megjegyzés
Nem megengedett programnevek

A problémák elkerülésére a Windows alkalmazásokkal a következő programneveket **nem** szabad használni:

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

Vezérlés belső bővítések

Az program létrehozásánál megadott alprogramneveket a vezérlés belül kibővíti egy elő- és utótaggal:

- előtag: `_N_`
- Utótag: `_SPF`

Programnevek használata

A programnevek használatánál, pl. egy alprogram-hívásnál az előtag, programnév és utótag minden kombinációja lehetséges.

Példa:

A `SUB_PROG` programnevű alprogramot a következő jelölőkkel lehet felhívni:

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

Fő- és alprogramok név-azonossága

Ha egy főprogram (.MPF) és egy alprogram (.SPF) azonos prograrmnévvel léteznek, a programnevek használatánál az NC programban meg kell adni a megfelelő fájl-bővítményt az egyértelmű azonosításhoz. Különben az a program lesz használva, amelyik a keresési ág mentén először található a megadott programnévvel.

2.24.1.3 Alprogramok egymásba skatulyázása

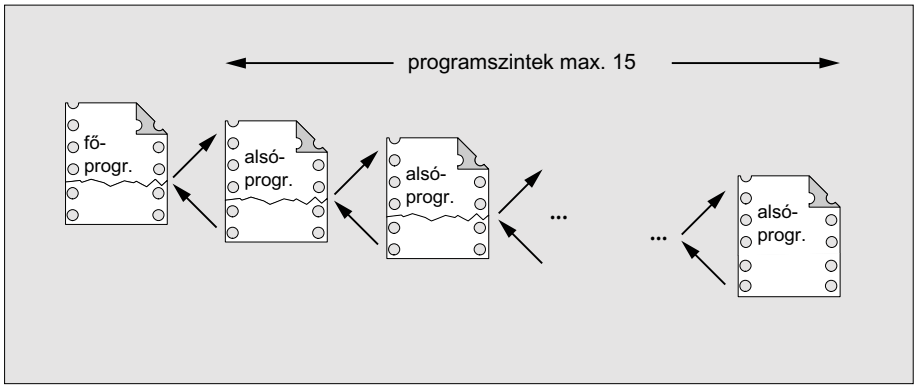
Egy főprogram fel tud hívni alprogramokat, amelyek ismét alprogramokat tudnak hívni. A programok lefutása így egymásba skatulyázható. Minden program egy saját programszinten fut.

Skatulyázási mélység

Az NC-nyelv aktuálisan 16 programszintet bocsájt rendelkezésre. A főprogram mindig a legfelső 0 programszinten fut. Egy alprogram mindig a felhívást követő alacsonyabb programszinten fut. A programszint 1 ezzel az első alprogramszint.

Programszintek felosztása:

- programszint 0: főprogramszintek
- programszint 1 - 15: alprogramszint 1 - 15



Interrupt-rutinok (ASUP)

Ha egy interrupt-rutin keretében egy alprogram lesz felhívva, akkor az nem a csatorna aktív programszintjén, hanem a következő alacsonyabb programszinten (n+1) lesz feldolgozva. Hogy ez még a legalsó programszinten is lehetséges legyen, az interrupt-rutinokkal összefüggésben 2 további programszint (16 és 17) áll rendelkezésre.

Ha kettőnél több programszintre van szükség, akkor azt külön figyelembe kell venni a csatornában végrehajtásra kerülő munkadarabprogram strukturálásában. Vagyis maximum annyi programszintet szabad igénybe venni, hogy az interrupt feldolgozáshoz még elegendő programszint álljon rendelkezésre.

Ha például az interrupt feldolgozás 4 programszintet igényel, akkor a munkadarabprogramot úgy kell strukturálni, hogy maximum 13 programszintet foglaljon el. Ha ekkor egy interrupt lép fel, rendelkezésére áll a szükséges 4 programszint (14 ... 17).

Siemens ciklusok

A Siemens ciklusok 3 szintet igényelnek Egy Siemens ciklus felhívása ezért legkésőbb meg kell történjen:

- munkadarabprogram feldolgozás programszint 12
- interrupt-rutinok: programszint 14

2.24.1.4 Keresési ág

Egy alprogram ág-megadás nélküli felhívásánál a vezérlés egy előre megadott keresési sorrend szerint (lásd "Kereső-ág alprogramhívásnál (Oldal 219)") keresi végig a programtárolót.

2.24.1.5 Formális és aktuális paraméterek

A formális és az aktuális paraméterekről az alprogramoknak paraméter-átadásos felhívásával és ennek definíciójával kapcsolatban beszélünk.

Formális paraméter

Egy alprogram definíciójánál az alprogramnak átadandó paraméterek, az úgynevezett formális paraméterek típussal és paraméternévvel definiáltak kell legyenek.

A formális paraméterek ezzel az alprogram interfészét definiálják.

Példa:

Programkód	Kommentár
PROC KONTUR (REAL X, REAL Y)	; Formális paraméterek: X és Y egyaránt REAL típusú
N20 X1=X Y1=Y	; X1 tengely X pozícióra mozog és az Y1 tengely az Y pozícióra
...	
N100 RET	

Aktuális paraméter

Az alprogram felhívásánál az alprogramnak abszolút értékeket vagy változókat, az úgynevezett aktuális paramétereket kell átadni.

Az aktuális paraméterek ezzel a felhívásnál az alprogram interfészét aktuális értékekkel töltik fel.

Példa:

Programkód	Kommentár
N10 DEF REAL BREITE	; változó definíció
N20 BREITE=20.0	; változó hozzárendelés
N30 KONTUR(5.5, BREITE)	; alprogram felhívása aktuális paraméterekkel: 5.5 és BREITE
...	
N100 M30	

2.24.1.6 Paraméter-átadás

Egy alprogram definíciója paraméter-átadással

Egy paraméter-átadásos alprogram definíciója a `PROC` kulcsszóval és az alprogram által elvárt paraméterek teljes felsorolásával történik.

Nem teljes paraméter-átadás

Az alprogram hívásánál nem kell mindig az alprogram-interfésznél definiált összes paramétert közvetlenül átadni. Ha egy paraméter elmarad, erre a paraméterre a "0" érték lesz átadva.

A paraméterek sorrendjének egyértelmű jelöléséhez azonban a vesszőt, mint a paraméterek elválasztójelét mindig meg kell adni. Kivételt képez az utolsó paraméter. Ha ez elmarad a felhívásnál, az utolsó vessző is elmaradhat.

Példa:

Alprogram:

Programkód	Kommentár
PROC SUB_PROG (REAL X, REAL Y, REAL Z)	; Formális paraméterek: X, Y és Z
...	
N100 RET	

Főprogram:

Programkód	Kommentár
PROC MAIN_PROG	
...	
N30 SUB_PROG(1.0,2.0,3.0)	; alprogram felhívása teljes paraméter-átadással: X=1.0, Y=2.0, Z=3.0

Programkód	Kommentár
…	
N100 M30	

Példák az alprogram felhívására az N30-ban nem teljes paraméter-átadással:

```
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
```

FIGYELEM

Paraméter-átadás Call-By-Reference

A paramétereket, amely Call-by-Reference által kerülnek átadásra, az alprogram felhívásánál nem szabad elhagyni.

FIGYELEM

AXIS adattípus

Az AXIS adattípusú paramétereket az alprogram felhívásánál nem szabad elhagyni.

Átadási paraméterek megvizsgálása

A $P_SUBPAR [n] rendszerváltozóval az n = 1, 2, ... értékkel meg lehet vizsgálni az alprogramban,hogy egy paraméter közvetlenül át lett-e adva vagy el lett hagyva. Az n index a formális paraméterek sorrendjére vonatkozik. Az index n = 1 az 1. formális paraméterre vonatkozik, az n =2 a 2. formális paraméterre stb

A következő programrészlet például az 1. formális paraméterre mutatja be, hogyan lehet egy vizsgálatot elvégezni:

Programozás	Kommentár
PROC SUB_PROG (REAL X, REAL Y, REAL Z)	; Formális paraméterek: X, Y és Z
N20 IF $P_SUBPAR[1]==TRUE	; 1. formális paraméter X vizsgálata.
...	; Ezek az akciók akkor lesznek végre-hajtva, ha az X formális paraméter köz-vetlenül lett átadva.
N40 ELSE	
...	; Ezek az akciók akkor lesznek végre-hajtva, ha az X formális paraméter nem lett átadva.
N60 ENDIF	
...	; Általános akciók
N100 RET	

2.24.2 Egy alprogram definíciója

2.24.2.1 Alprogram paraméter-átadás nélkül

Egy paraméter-átadás nélküli alprogram definíciójánál a definíciós sor a program elején elmaradhat.

Szintaxis

```
[PROC <programnév>]
...
```

Jelentés

PROC:	definíciós utasítás egy program elején
<programnév>:	egy program neve

Példa

Példa 1: Alprogram PROC utasítással

Programkód	Kommentár
PROC SUB_PROG	; definíciós sor
N10 G01 G90 G64 F1000	
N20 X10 Y20	
...	
N100 RET	; alprogram-visszaugrás

Példa 2: Alprogram PROC utasítás nélkül

Programkód	Kommentár
N10 G01 G90 G64 F1000	
N20 X10 Y20	
...	
N100 RET	; alprogram-visszaugrás

Lásd még

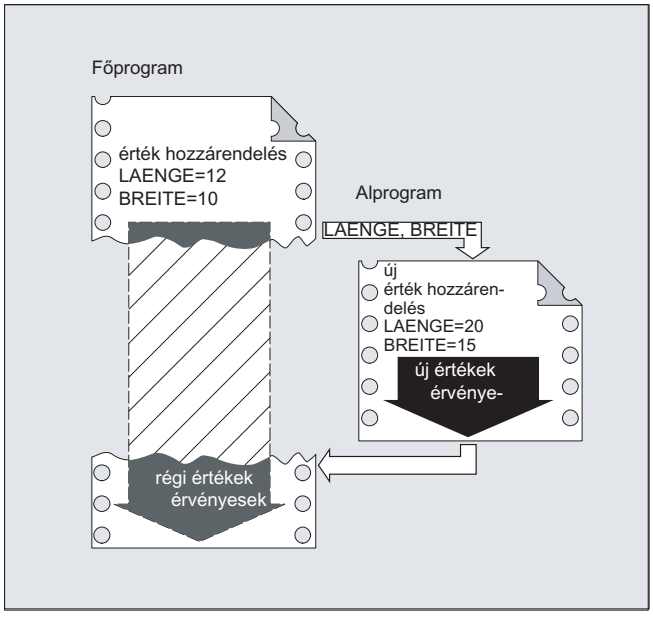
Alprogram hívása paraméter-átadás nélkül (Oldal 188)

2.24.2.2 Alprogram Call-by-Value paraméter-átadással (PROC)

Egy Call-by-Value paraméter-átadásos alprogram definíciója a PROC kulcsszóval, utána a programnévvel és az összes paraméter teljes felsorolásával típussal és névvel történik. A definíciós utasítás az első programsorban kell álljon.

Call-By-Value

A felhívó program Call-By-Value paraméter-átadásnál az alprogramnak csak egy változó értékét adja át. Ezzel az alprogram nem kap közvetlen hozzáférést a változóhoz. Ezzel a paraméter-érték változtatásánál csak az alprogramban látható érték változik. A felhívó programban definiált változók értéke változatlan marad. A Call-by-Value paraméter-átadásnak így nincs visszahatása a felhívó programra.



Szintaxis

```
PROC <programnév> (<paraméter típus> <paraméter
név>=<Init_érték>, ...)
```

Megjegyzés

Maximum 127 paramétert lehet átadni.

Jelentés

PROC:	definíciós utasítás egy program elején
<programnév>:	egy program neve
<paraméter-típus>:	paraméter adattípusa (pl. REAL, INT, BOOL)
<paraméter-név>:	paraméter neve
<inic_érték>:	opcionális érték a paraméter inicializálásához (**opcionális**) Ha az alprogram felhívásánál nincs megadva paraméter, a paramé-terhez az inicializálási érték lesz hozzárendelve.

Példák

Példa 1

Egy SUB_PROG alprogram definíciója három REAL típusú paraméterrel alapértékekkel:

Programkód

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

Példa 2

Különféle felfutási változatok

Programkód

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

Lásd még

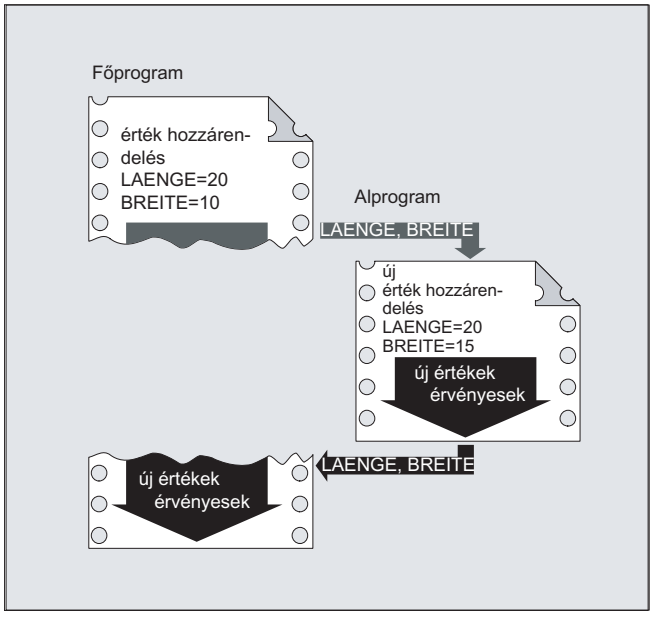
Alprogram felhívása paraméter-átadással (EXTERN) (Oldal 190)

2.24.2.3 Alprogram Call-by-Reference paraméter-átadással (PROC, VAR)

Egy Call-by-Reference paraméter-átadásos alprogram definíciója a `PROC` kulcsszóval, utána a programnévvel és az összes paraméter teljes felsorolásával `VAR` kulcsszóval, típussal és névvel történik. A definíciós utasítás az első programsorban kell álljon. Paraméterként át lehet adni referenciákat is mezőkre.

Call-By-Reference

A felhívó program Call-By-Reference paraméter-átadásnál az alprogramnak nem egy változó értékét adja át, hanem egy referenciát (mutatót) a változóra. Ezzel az alprogram közvetlen hozzáférést kap a változóhoz. Ezzel a paraméter-érték változtatásánál nem csak az alprogramban látható érték változik, hanem a felhívó programban definiált változók értéke is. A Call-by-Reference paraméter-átadásnak ezzel az alprogram befejezése után is visszahatása van a felhívó programra.



Megjegyzés

A Call-by-Reference paraméter-átadás csak akkor szükséges, ha az átadandó változó a helyileg egy felhívó programban lett definiálva (LUD). Csatorna-globális és NC-globális változókat nem kell átadni, azokhoz az alprogramból közvetlenül hozzá lehet férni.

Szintaxis

```
PROC <programnév> (VAR <paraméter-típus> <paraméternév>, ...)
PROC <programnév> (VAR <mező-típus> <mezőnév> [<m>,<n>,<o>], ...)
```

Megjegyzés

Maximum 127 paramétert lehet átadni.

Jelentés

PROC:	definíciós utasítás egy program elején
VAR:	kulcsszó paraméter-átadáshoz referenciával
<programnév>:	egy program neve
<paraméter-típus>:	paraméter adattípusa (pl. REAL, INT, BOOL)
<paraméternév>:	paraméter neve
<mező-típus>:	mezőelem adattípusa (pl. REAL, INT, BOOL)
<mezőnév>:	egy mező neve

[<m>,<n>,<o>]:	mező-méret aktuálisan maximum 3-dimenziős mezők lehetségesek:	
	<m>:	mezőméret az 1. dimenzióhoz
	<n>:	mezőméret az 2. dimenzióhoz
	<o>:	mezőméret az 3. dimenzióhoz

Megjegyzés

- A PROC kulcsszó után megadott programnév meg kell egyezzen a kezelőfelületen megadott programnévvel.
- A nem meghatározott mező-hosszúságú mezőkkel formális paraméterként az alprogramok változtatható mező-hosszúságú mezőkkel tudnak dolgozni. Ehhez pl. egy két-dimenziós mező definíciójánál az 1. dimenzió hossza formális paraméterként nem lesz megadva. A vesszőt azonban ki kell írni.
  Példa: PROC <programnév> (VAR REAL FELD[,5])

Példa

Egy alprogram definíciója két paraméterrel referenciaként REAL típusra :

Programkód

```
; paraméter 1: referencia típusra: REAL, név: LAENGE
; paraméter 2: referencia típusra: REAL, név: BREITE
PROC SUB_PROG(VAR REAL LAENGE, VAR REAL BREITE)
```

Lásd még

Alprogram felhívása paraméter-átadással (EXTERN) (Oldal 190)

2.24.2.4 Modális G-funkciókat menteni (SAVE)

A SAVE beállítással az alprogram felhívása előtt aktív modális G utasítások mentve lesznek és az alprogram vége után ismét aktiválva.

FIGYELEM
Pályavezérlő-üzem megszakítása
Ha aktív pályavezérlő-üzemnél egy alprogram SAVE beállítással fel lesz hívva, a pályavezérlő-üzem az alprogram végénél (visszaugrás) meg lesz szakítva.

Szintaxis

```
PROC <alprogramnév> SAVE
```

Jelentés

SAVE:	Modális G utasítások mentése az alprogram felhívása előtt és helyreállítása az alprogram vége után

Példa

A KONTUR alprogramban a G91 (abszolút méret) modális G utasítás hat. A főprogramban a G90 (abszolút méret) globális G utasítás hat. Az alprogram definíciója SAVE-vel biztosítja, hogy az alprogram vége után a főprogramban ismét a G90 hat.

Alprogram definíció:

Programkód	Kommentár
PROC KONTUR (REAL WERT1) SAVE	; alprogram definíció SAVE paraméterrel
N10 G91 ...	; G91 modális G-funkció: láncméret
N100 M17	; alprogramvég

Főprogram:

Programkód	Kommentár
N10 G0 X... Y... G90	; G90 modális G-funkció: Abszolút méret
N20 ...	
...	
N50 KONTUR (12.4)	; alprogramhívás
N60 X... Y...	; G90 modális G utasítás SAVE-vel újra aktiválva

Peremfeltételek

Frame-ek

A frame-ek viselkedése a SAVE tulajdonágú alprogramok vonatkozásában függ a frame típusától és gépadatban beállítható.

Irodalom

Alapfunkciók működési kézikönyv; Tengelyek, koordináta-rendszerek, Frame-k (K2), fejezet: "Alprogram visszaugrás SAVE-vel"

2.24.2.5 Egyes-mondat feldolgozás elnyomás (SBLOF, SBLON)

Egyes-mondat elnyomás a teljes programra

Az SBLOF-fal megjelölt programok aktív egyes-mondat feldolgozásnál egy mondatként kompletten lesznek végrehajtva, vagyis a teljes programra el lesz nyomva az egyes-mondat feldolgozás.

Az SBLOF a PROC-sorban áll és az alprogram végéig vagy megszakadásáig érvényes. A visszaugrás utasítással lesz eldöntve, hogy az alprogram végén legyen megállás vagy ne.

Visszaugrás M17-tel:	állj az alprogram végén
Visszaugrás RET-tel:	nincs állj az alprogram végén

Egyes-mondat elnyomás a programon belül

Az SBLOF egyedül kell álljon a mondatban. Ettől a mondattól kezdve az egyes-mondat ki lesz kapcsolva:

- a következő SBLON-ig
  vagy
- az aktív alprogram-szint végéig

Szintaxis

Egyes-mondat elnyomás a teljes programra:

```
PROC ... SBLOF
```

Egyes-mondat elnyomás a programon belül:

```
SBLOF
...
SBLON
```

Jelentés

PROC:	egy program első utasítása
SBLOF:	utasítás az egyes-mondat megmunkálás kikapcsolására Az SBLOF állhat egyPROC-mondatban vagy egyedül egy mondatban.
SBLON:	utasítás az egyes-mondat megmunkálás bekapcsolására Az SBLON egy önálló mondatban kell legyen.

Peremfeltételek

- **Egyes-mondat elnyomás és mondat-kijelző**
  Az aktuális mondat kijelzését a ciklusokban/alprogramokban A `DISPLOF`-fal el lehet nyomni. Ha a `DISPLOF` az `SBLOF`-fal együtt van programozva, akkor egyes-mondat álljnál a cikluson/alprogramon belül, akárcsak a ciklus/alprogram-hívás előtt, lesz kijelzés.
- **Egyes-mondat elnyomás rendszer-ASUP-ban vagy alkalmazói-ASUP-ban**
  Ha az egyes-mondat állj a rendszer-ASUP-ban vagy az alkalmazói-ASUP-ban az MD10702 IGNORE_SINGLEBLOCK_MASK-ban el van nyomva (bit0 = 1 ill. bit1 = 1), az egyes-mondat álljt ismét aktiválni lehet az ASUP-ban az `SBLON` programozásával.
  Ha az egyes-mondat állj az alkalmazói-ASUP-ban az MD20117 $MC_IGNORE_SINGLEBLOCK_ASUP-pal el van nyomva, akkor az egyes-mondat álljt az`SBLON` programozásával az ASUP **nem** lehet ismét aktiválni.
- **Az egyes-mondat elnyomás különlegességei a különböző egyes-mondat megmunkálási típusoknál**
  Az aktív SBL2 egyes-mondat megmunkálásnál (állj minden munkadarabprogram mondat után) az `SBLON` mondatban **nem** lesz megállás, ha az MD10702 $MN_IGNORE_SINGLEBLOCK_MASK (egyes-mondat álljt megakadályozni) bit 12 "1"-re van állítva.
  Az aktív SBL3 egyes-mondat megmunkálásnál (állj minden munkadarabprogram mondat után a ciklusban is) az `SBLOF` el lesz nyomva.

Példák

Példa 1: Egyes-mondat elnyomás egy programon belül

Programkód	Kommentár
`N10 G1 X100 F1000`	
`N20 SBLOF`	`; egyes-mondatot kikapcsolni`
`N30 Y20`	
`N40 M100`	
`N50 R10=90`	
`N60 SBLON`	`; egyes-mondatot újra bekapcsolni`
`N70 M110`	
`N80 ...`	

Az `N20` és `N60` közötti tartomány az egyes-mondat üzemben egy lépésként lesz feldolgozva.

Példa 2: Ciklus a felhasználó számára egy utasításként kell hasson

Főprogram:

Programkód
`N10 G1 X10 G90 F200`
`N20 X-4 Y6`
`N30 CYCLE1`
`N40 G1 X0`
`N50 M30`

CYCLE1 ciklus:

Programkód	Kommentár
N100 PROC CYCLE1 DISPLOF SBLOF	; egyes-mondat elnyomás
N110 R10=3*SIN(R20)+5	
N120 IF (R11 <= 0)	
N130 SETAL(61000)	
N140 ENDIF	
N150 G1 G91 Z=R10 F=R11	
N160 M17	

A CYCLE1 ciklus aktív egyes-mondatnál végre lesz hajtva, vagyis a CYCLE1 végrehajtásához a Start-billentyűt egyszer meg kell nyomni.

Példa 3: Egy PLC által indított ASUP a megváltozott nullaponteltolás és szerszámkorrekciók aktiválásához ne kell látható legyen

```
Programkód
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=$P_TOOLNO
N130 RET
N140 _G54: G55 D=$P_TOOL T=$P_TOOLNO
N150 RET
N160 _G56: G56 D=$P_TOOL T=$P_TOOLNO
N170 RET
N180 _G57: G57 D=$P_TOOL T=$P_TOOLNO
N190 RET
N200 END: D=$P_TOOL T=$P_TOOLNO
N210 RET
```

Példa 4: MD10702 bit 12 = 1 esetén nem lesz megállás

Kiinduló helyzet:

- egyes-program megmunkálás aktív
- MD10702 $MN_IGNORE_SINGLEBLOCK_MASK Bit12 = 1

Főprogram:

Programkód	Kommentár
N10 G0 X0	; Ebben a program-sorban megállás.
N20 X10	; Ebben a program-sorban megállás.
N30 CYCLE	; Ciklus által generált mozgásmondat.
N50 G90 X20	; Ebben a program-sorban megállás.
M30	

CYCLE ciklus:

Programkód	Kommentár
PROC CYCLE SBLOF	; Egyes-mondat álljt elnyomni
N100 R0 = 1	
N110 SBLON	; MD10702 Bit12=1 miatt ebben a programsorban nincs megállás.
N120 X1	; Ebben a program-sorban lesz megállás.
N140 SBLOF	
N150 R0 = 2	
RET	

Példa 5: Egyes-mondat elnyomás program-skatulyázásánál

Kiinduló helyzet:

egyes-program megmunkálás aktív

Program-skatulyázás:

Programkód	Kommentár
N10 X0 F1000	; Ebben a mondatban lesz megállás.
N20 UP1(0)	
PROC UP1(INT _NR) SBLOF	; Egyes-mondat álljt elnyomni.
N100 X10	
N110 UP2(0)	
PROC UP2(INT _NR)	
N200 X20	
N210 SBLON	; Egyes-mondat álljt bekapcsolni.
N220 X22	; Ebben a mondatban lesz megállás.
N230 UP3(0)	
PROC UP3(INT _NR)	
N300 SBLOF	; Egyes-mondat álljt elnyomni.
N305 X30	
N310 SBLON	; Egyes-mondat álljt bekapcsolni.
N320 X32	; Ebben a mondatban lesz megállás.

Programkód	Kommentár
N330 SBLOF	; Egyes-mondat álljt elnyomni.
N340 X34	
N350 M17	; SBLOF aktív.
N240 X24	; Ebben a mondatban lesz megállás. SBLON aktív
N250 M17	; Ebben a mondatban lesz megállás. SBLON aktív
N120 X12	
N130 M17	; Ebben a visszaugrás mondatban lesz megállás. SBLOF PROC utasításból aktív
N30 X0	; Ebben a mondatban lesz megállás.
N40 M30	; Ebben a mondatban lesz megállás.

További információk

Egyes-mondat tiltás aszinkron alprogramokhoz

Egy ASUP egyes-mondatban egy lépésben feldolgozható, ha az ASUP-ban egy PROC utasítás van programozva SBLOF-fal. Ez a "szerkeszthető rendszer-ASUP"-ra is érvényes (MD11610 $MN_ASUP_EDITABLE).

Példa szerkeszthető rendszer-ASUP-ra:

Programkód	Kommentár
N10 PROC ASUP1 SBLOF DISPLOF	
N20 IF $AC_ASUP=='H200'	
N30 RET	; Nincs REPOS üzemmód váltásnál.
N40 ELSE	
N50 REPOSA	; REPOS minden egyéb esetben.
N60 ENDIF	

Program-befolyásolások egyes-mondatban

Az egyes-mondat funkcióban a felhasználó a munkadarabprogramot mondatonként fel tudja dolgozni. A következő beállítási módok vannak:

- SBL1: IPO egyes-mondat megállással minden gépfunkció-mondat után
- SBL2: egyes-mondat megállással minden mondat után
- SBL3: állj a ciklusban (az SBL3 kiválasztásával az SBLOF utasítás el lesz nyomva).

Egyes-mondat elnyomás program-skatulyázásánál

Ha egy alprogramban SBLOF lett programozva a PROC utasításban, akkor megállás lesz az M17 alprogram-visszaugrásnál. Ezzel meg lesz akadályozva, hogy a felhívó programban már a következő mondat végre legyen hajtva. Ha egy alprogramban SBLOF-fal, SBLOF nélkül a PROC utasításban, egy egyes-mondat elnyomás lesz aktiválva, csak a felhívó program következő gépfunkció-mondata után lesz megállás. Ha ez nem kívánatos, az alprogramban még a visszaugrás (M17) előtt ismét SBLON-t kell programozni. Egy RET visszaugrásnál a fölérendelt programban nem lesz megállás.

2.24.2.6 Aktuális mondatkijelzés elnyomása (DISPLOF, DISPLON, ACTBLOCNO)

A mondatkijelzőn alapesetben az aktuális programmondat van kijelezve. A ciklusokban ill. alprogramokban az aktuális mondat kijelzését a DISPLOF utasítással el lehet nyomni.. Az aktuális mondat helyett a ciklus vagy az alprogram hívása lesz kijelezve. A DISPLON utasítással a mondatkijelzés elnyomását ismét meg lehet szüntetni..

A DISPLOF ill. DISPLON a PROC utasítás sorában van programozva és a teljes alprogramban hat és közvetve az összes, ebből az alprogramból felhívott alprogramban is, amelyek nem tartalmaznak DISPLON ill. DISPLOF utasítást. Ez a viselkedés érvényes ASUP-okra is.

Szintaxis

```
PROC … DISPLOF
PROC … DISPLOF ACTBLOCNO
PROC … DISPLON
```

Jelentés

DISPLOF:	utasítás az aktuális mondatkijelzést elnyomására	
	elhelyezés:	PROC utasítást tartalmazó programsor végén
	hatásosság:	az alprogramból történő visszaugrásig vagy a program végéig
	Utalás: Ha a DISPLOF utasítást tartalmazó alprogramból további alprogramok vannak felhívva, akkor az aktuális mondatkijelzés ezekben is el van nyomva, hacsak ezekben nincs DISPLON programozva.	
DISPLON:	utasítás az aktuális mondatkijelzést elnyomás megszüntetésére	
	elhelyezés:	PROC utasítást tartalmazó programsor végén
	hatásosság:	az alprogramból történő visszaugrásig vagy a program végéig
	Utalás: Ha a DISPLOF utasítást tartalmazó alprogramból további alprogramok vannak felhívva, akkor az aktuális mondatkijelzés ezekben is el van nyomva, hacsak ezekben nincs DISPLON programozva.	
ACTBLOCNO:	A DISPLOF együtt az ACTBLOCNO tulajdonsággal azt eredményezi, hogy egy vészjelzés fellépése esetén az aktuális mondat száma lesz kiadva, amelyikben a vészjelzés fellépett. Ez akkor is érvényes, ha egy alacsonyabb programszinten csak DISPLOF van programozva. A DISPLOF-nál ACTBLOCNO nélkül ellenben a ciklus ill. alprogram felhívás mondatszáma lesz kijelezve az utolsó, nem DISPLOF megjelölésű programszintről.	

Példák

Példa 1: Aktuális mondatkijelzést a ciklusban elnyomni

Programkód	Kommentár
PROC CYCLE(AXIS TOMOV, REAL POSITION) SAVE DISPLOF	; Aktuális mondat kijelzést elnyomni. helyette legyen kijelezve a ciklus hívása, pl.: CYCLE(X,100.0)
DEF REAL DIFF	; ciklus tartalom
G01 …	

Programkód	Kommentár
...	
RET	; Alprogram visszaugrás. A mondatkijelzőn a ciklus hívását követő mondat lesz kijelezve.

Példa 2: Mondatkijelzés a vészjelzés kiadásánál

SUBPROG1 alprogram (ACTBLOCNO-val):

Programkód	Kommentár
PROC SUBPROG1 DISPLOF ACTBLOC-NO	
N8000 R10 = R33 + R44	
...	
N9040 R10 = 66 X100	; vészjelzés 12080 kiváltása
...	
N10000 M17	

SUBPROG2 alprogram (ACTBLOCNO nélkül):

Programkód	Kommentár
PROC SUBPROG2 DISPLOF	
N5000 R10 = R33 + R44	
...	
N6040 R10 = 66 X100	; vészjelzés 12080 kiváltása
...	
N7000 M17	

Főprogram:

Programkód	Kommentár
N1000 G0 X0 Y0 Z0	
N1010 ...	
...	
N2050 SUBPROG1	; vészjelzés kiadása = "12080 csatorna K1 mondat N9040 szintakszis-hiba szövegnél R10="
N2060 ...	
N2350 SUBPROG2	; vészjelzés kiadása = "12080 csatorna K1 mondat N2350 szintakszis-hiba szövegnél R10="
...	
N3000 M30	

Példa 3: Aktuális mondat kijelzést elnyomást megszüntetni

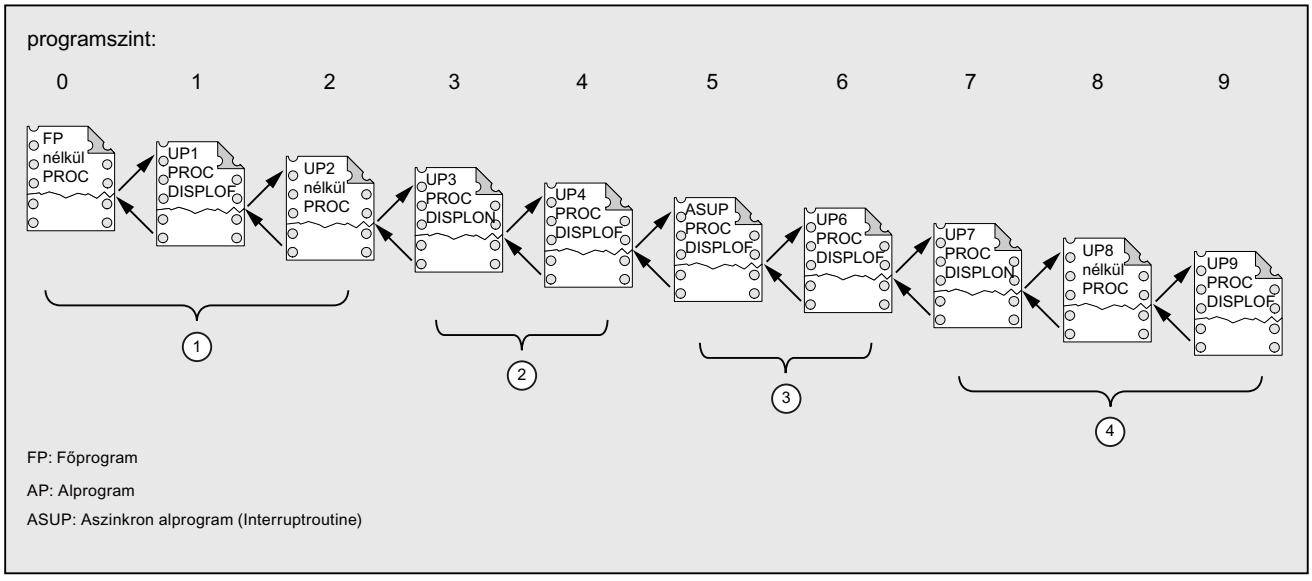
SUB1 alprogram elnyomással:

Programkód	Kommentár
PROC SUB1 DISPLOF	; Aktuális mondatkijelzést SUB1 alprogramban elnyomni. Ehelyett a SUB1 felhívást tartalmazó mondatot kell kijelezni.
...	
N300 SUB2	; SUB2 alprogram hívása.
...	
N500 M17	

SUB2 alprogram elnyomás nélkül:

Programkód	Kommentár
PROC SUB2 DISPLON	; Aktuális mondatkijelzés elnyomását a SUB2 alprogramban megszüntetni.
...	
N200 M17	; Visszaugrás a SUB1 alprogramba. SUB1-ben az aktuális mondatkijelzés ismét el lesz nyomva.

Példa 4: Kijelzés viselkedése különböző DISPLON/DISPLOF kombinációknál



① Az aktuális mondatkijelzőben a munkadarabprogram sorai a programszint 0-ból lesznek kijelezve.

② Az aktuális mondatkijelzőben a munkadarabprogram sorai a programszint 3-ból lesznek kijelezve.

③ Az aktuális mondatkijelzőben a munkadarabprogram sorai a programszint 3-ból lesznek kijelezve.

④ Az aktuális mondatkijelzőben a munkadarabprogram sorai a programszint 7/8-ból lesznek kijelezve.

2.24.2.7 Alprogramokat előkészítéssel megjelölni (PREPRO)

A PREPRO kulcsszóval a felfutásnál a PROC utasítássor végén minden fájlt meg lehet jelölni.

Megjegyzés

A program-előkészítés ezen módja a megfelelően beállított gépadattól függ. Kérjük, vegye ehhez figyelembe e gépgyártó tájékoztatásait.

Irodalom:
Különleges funkciók működési kézikönyv; Elő-feldolgozás (V2)

Szintaxis

```
PROC … PREPRO
```

Jelentés

PREPRO:	kulcsszó az összes, a felfutásnál előkészített fájl- és a ciklus-könyvtárakban levő NC-programok megjelöléséhez

Alprogramokat előkészítéssel beolvasni és alprogram-hívás

A felfutásnál előkészített alprogramok paraméterekkel és az alprogram-hívásnál is a cikluskönyvtárak azonos sorrendben lesznek kezelve

1. _N_CUS_DIR alkalmazói ciklusok
2. _N_CMA_DIR gyártói ciklusok
3. _N_CST_DIR szabvány ciklusok

Az azonos nevű NC-programok különböző változatainál az először megtalált PROCutasítás lesz aktiválva és a többi PROC utasítás vészjelzés nélkül figyelmen kívül lesz hagyva.

2.24.2.8 Alprogram-visszaugrás M17

Egy alprogram végén van az M17 visszaugrás utasítás (ill. az M30 alprogramvég utasítás). Ez visszaugrást okoz a felhívó programba a munkadarab-programnak az alprogram-hívás utáni programsorára.

Megjegyzés

M17 és M30 az NC-nyelvben azonos értékűnek van kezelve.

Szintaxis

```
PROC <programnév>
...
M17/M30
```

Peremfeltételek

Alprogram-visszaugrás hatása a pályavezérlő-üzemre

Ha az M17 (ill. M30) egyedül áll a munkadarabprogramban, akkor megszakítja a csatornában aktív pályavezérlő-üzemet.

A pályavezérlő-üzem megszakításának elkerülésére az M17-et (ill. M30-at) az utolsó mozgás-mondatba kell írni. Ezen kívül a következő gépadatot "0"-ba kell állítani:

MD20800 $MC_SPF_END_TO_VDI = 0 (nincs M30/M17 kiadás az NC/PLC interfészen)

Példa

1. Alprogram M17-tel külön mondatban

Programkód	Kommentár
N10 G64 F2000 G91 X10 Y10	
N20 X10 Z10	
N30 M17	; Visszaugrás a pályavezérlő-üzem megszakításával.

2. Alprogram M17-tel az utolsó mozgás-mondatban

Programkód	Kommentár
N10 G64 F2000 G91 X10 Y10	
N20 X10 Z10 M17	; Visszaugrás a pályavezérlő-üzem megszakítása nélkül.

2.24.2.9 Alprogram-visszaugrás RET

Az M17 visszaugrási utasítás helyezz az alprogramban lehet a RET utasítást is használni. A RET egy önálló munkadarabprogramban kell legyen programozva. Akár az M17 a RET is visszaugrást okoz a felhívó programba a munkadarabprogramnak az alprogram-hívás utáni programsorára.

Megjegyzés

Paraméterek programozásával meg lehet változtatni a RET visszaugrás viselkedését (lásd "Paraméterezhető alprogram-visszaugrás (RET...) (Oldal 178)").

Alkalmazás

A RET utasítást akkor kell használni, ha G64 pályavezérlő-üzemet (G641 ... G645) a visszaugrással nem akarjuk megszakítani.

Előfeltétel

A RET utasítást csak olyan alprogramokban szabad használni, amelyek nem a SAVE tulajdonsággal lettek definiálva.

Szintaxis

```
PROC <programnév>
...
RET
```

Példa

Főprogram:

Programkód	Kommentár
PROC MAIN_PROGRAM	; programkezdet
...	
N50 SUB_PROG	; alprogramhívás: SUB_PROG
N60 ...	
...	
N100 M30	; programvég

Alprogram:

Programkód	Kommentár
PROC SUB_PROG	
...	
N100 RET	; Visszaugrás az N60 mondatra a főprogramban.

2.24.2.10 Paraméterezhető alprogram-visszaugrás (RET...)

Általában a `RET` utasítással egy alprogramból történik visszaugrás a felhívó programba. A feldolgozás ezután az alprogramhívást követő programsorral folytatódik. Ha a program feldolgozását egy másik helyen kell folytatni, a következő lehetőségek állnak rendelkezésre:

- A programfeldolgozás folytatása leforgácsoló ciklusok felhívása után ISO-dialektus módban (a kontúrleírás után).
- Visszaugrás a főprogramba egy tetszőleges alprogram-szintről (ASUP után is) hiba-kezelésnél.
- Visszaugrás több programszinten át speciális alkalmazások számára Compile-ciklusokban és ISO-dialektus módban

Ehhez a `RET` utasítást további paraméterekkel kell programozni.

Keresési irány

A `<célmondat>` paraméter megadásánál a visszaugrás először a felhívó mondat utáni mondatra történik. Ezután a cél annak a programnak a program**vége** irányában történik, ahova a visszaugrás lesz. Ha a keresés nem sikeres, a következőként a keresés a program kezdete irányában történik.

Szintaxis

```
RET("<célmondat>")
RET("<célmondat>",<mondat a célmondat után>)
RET("<célmondat>",<mondat a célmondat után>,<visszaugrási szintek
száma>)
RET("<célmondat>", ,<visszaugrási szintek száma>)
RET("<célmondat>",<mondat a célmondat után>,<visszaugrási szintek
száma>),
<visszaugrás programkezdetre>)
RET( , ,<visszaugrási szintek száma>,<visszaugrás programkezdetre>)
```

Jelentés

RET:	Alprogramvég	
<célmondat>:	A paraméterben az ugráscélként az a mondat lesz megadva, ahol a programfeldolgozást folytatni kell Ha a <visszaugrási szintek száma> paraméter nincs programozva, akkor az ugráscél abban a programban van, amelyikből az aktuális alprogram fel lett hívva. Lehetséges megadások:	
	<mondatszám>	Célmondat száma A mondatszám keresése a programban, ahova vissza lesz ugorva, először a programvég irányában történik.
	<ugrás-jelölő>	Ugrás-jelölő, amelyik jelen kell legyen abban a programban, ahova vissza lesz ugorva. Az ugrás-jelölő keresése a programban, ahova vissza lesz ugorva, először a programvég irányában történik.
	<karakterlánc>	Karakterlánc, amelyik jelen kell legyen abban a programban, ahova vissza lesz ugorva (pl. program vagy változó név). A karakterlánc keresése a programban, ahova vissza lesz ugorva, először a programvég irányában történik. A karakterlánc programozására a célmondatban a következő szabályok érvényesek: • **Üres jel a végén** (megkülönböztetés az ugrás-jelelölőhöz, amelyik egy ":"-vel a végén van megjelölve). • **karakterlánc előtt** csak egy mondatszám és/vagy egy ugrás-jelölő lehet, **program utasítások nem**.

`<célmondat utáni mondat>`:	A paraméterben meg lesz adva, hogy a program feldolgozása a <célmondat> paraméterben megadott mondatnál vagy az utána következő mondatnál legyen folytatva.		
	típus:	INT	
	érték:	0	A visszaugrás a <célmondat> paraméterben megadott mondatnál folytatódik.
		> 0	A visszaugrás a <célmondat> paraméterben megadott következő mondatnál folytatódik.
`<visszaugrási szintek száma>`:	A paraméterben megadásra kerül a programszintek száma, ahova vissza lesz ugorva, ott a célmondatot keresni és a program feldolgozását folytatni.		
	típus:	INT	
	érték:	1	a program az "aktuális programszint - 1"-en lesz folytatva (mint `RET` paraméter nélkül).
		2	a program az "aktuális programszint - 2"-en lesz folytatva, vagyis egy szint át lesz ugorva
		3	a program az "aktuális programszint - 3"-en lesz folytatva, vagyis két szint át lesz ugorva
		...	
	értéktartomány:	1 ... 15	
`<visszaugrás programkezdetre>`:	A paraméterben meg lesz adva, hogy a főprogramba visszaugrásnál és ott aktív **ISO dialektus mód** esetén a program a programkezdetnél legyen folytatva.		
	típus:	BOOL	
	érték:	1	Ha a visszaugrás a főprogramba történik és ott egy **ISO-dialektus modus** aktív, elágazás lesz a programkezdetre.

Megjegyzés

Egy alprogram-visszaugrásnál karakterlánccal a célmondat-keresés megadásaként a felhívó mondatban mindig először egy ugrás-jelölő után lesz keresve.

Ha az ugráscélt egy karakterlánccal egyértelműen kell definiálni, akkor a karakterlánc nem egyezhet egy ugrás-jelölő nevével, mert különben az alprogram-visszaugrás mindig az ugrás-jelölőre és nem a karakterláncra lesz végrehajtva (lásd példa 2).

Peremfeltételek

A visszaugrásnál több programszinten át az egyes programszintek `SAVE` utasításai ki lesznek értékelve.

Ha egy visszaugrásnál több programszinten át egy modális alprogram aktív és az átugrott alprogramok egyikében az `MCALL` kikapcsoló utasítás van programozva, a modális alprogram továbbra is aktív marad.

FIGYELEM
Programozási hiba
Egy visszaugrásnál több programszinten át kizárólag a felhasználóé felelősség arról gondoskodni, hogy a szükséges modális beállítások folytatódjanak. Ez pl. egy megfelelő főmondat programozásával érhető el.

Példák

Példa 1: Újra rámenet a főprogramban ASUP-feldolgozás után

```
Programozás                          Kommentár
N10010 CALL "UP1"                    ; programszint 0 (főprogram)
  N11000 PROC UP1                    ; programszint 1
  N11010 CALL "UP2"
  N12000 PROC UP2                    ; programszint 2
  ...
  N19000 PROC ASUP                   ; programszint 3 (ASUP feldolgozás)
  ...
  N19100 RET("N10900", ,$P_STACK)    ; alprogram-visszaugrás a főprogramba
                                     ; $P_STACK: aktuális programszint
N10900                               ; célmondat a főprogramban
N10910 MCALL                         ; modális alprogram hívást kikapcsolni
N10920 G0 G60 G40 M5                 ; további modális beállításokat inicializálni
```

Példa 2: Karakterlánc (<String>) a célmondat keresés megadására

Főprogram:

```
Programkód                           Kommentár
PROC MAIN_PROGRAM
N1000 DEF INT iVar1=1, iVar2=4
N1010 ...
N1200 subProg1                       ; "subProg1" alprogram felhívása
N1210 M2 S1000 X10 F1000
N1220 ......
N1400 subProg2                       ; "subProg2" alprogram felhívása
N1410 M3 S500 Y20
N1420 ..
N1500 lab1: iVar1=R10*44
N1510 F500 X5
```

Programkód	Kommentár
N1520 ...	
N1550 subprog1: G1 X30	; "subProg1" itt ugrás-jelölőként van definiálva
N1560 ...	
N1600 subProg3	; "subProg3" alprogram felhívása
N1610 ...	
N1900 M30	

subProg1 alprogram:

Programkód	Kommentár
PROC subProg1	
N2000 R10=R20+100	
N2010 ...	
N2200 RET("subProg2")	; visszaugrás a főprogramba az N1400 mondatra

subProg2 alprogram:

Programkód	Kommentár
PROC subProg2	
N2000 R10=R20+100	
N2010 ...	
N2200 RET("iVar1")	; visszaugrás a főprogramba az N1500 mondatra

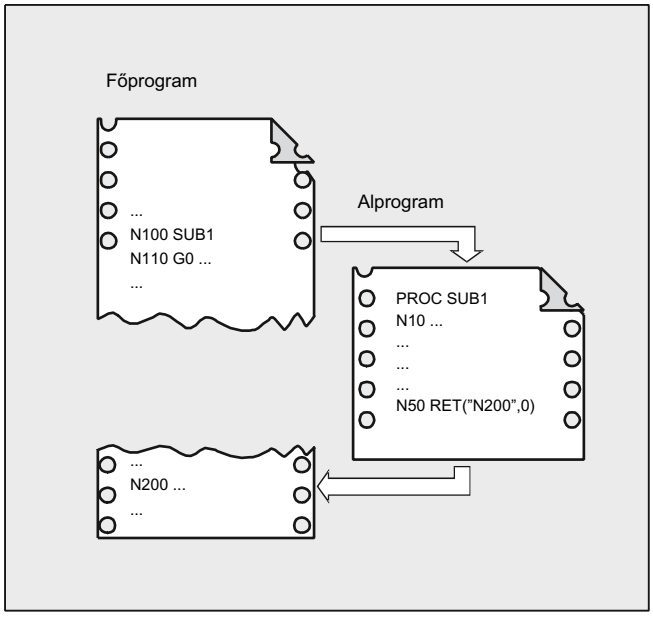
subProg3 alprogram:

Programkód	Kommentár
PROC subProg3	
N2000 R10=R20+100	
N2010 ...	
N2200 RET("subProg1")	; visszaugrás a főprogramba az N1550 mondatra

További információk

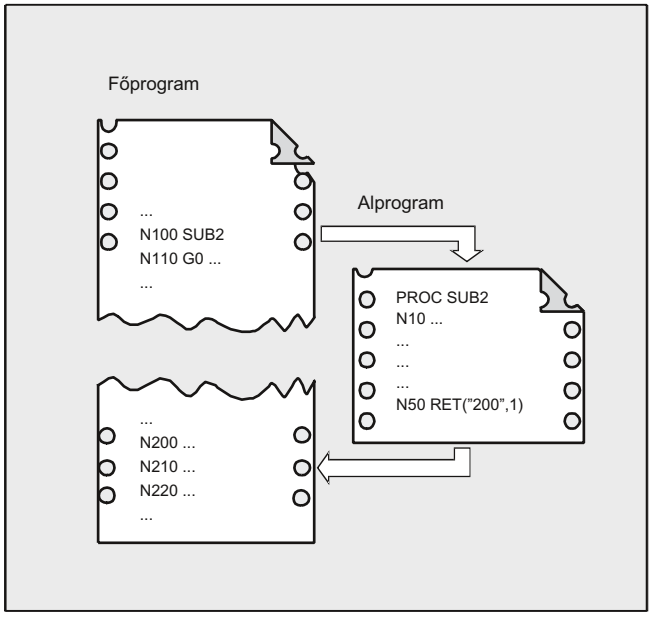
A következő ábrák bemutatják a visszaugrási-paraméterek különböző hatásait

1. <célmondat> = "N200", <célmondat utáni mondat> = 0



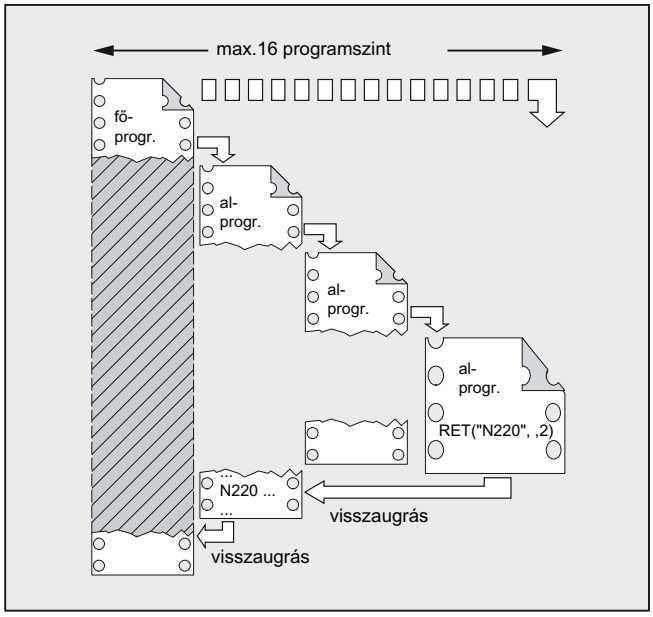
A `RET` utasítás után a program feldolgozás a főprogram `N200` mondatában lesz folytatva.

2. <célmondat> = "N200", <célmondat utáni mondat> = 1



A `RET` utasítás után a program feldolgozás az (`N210`) mondattal lesz folytatva, ami a főprogramban az `N200` mondat után következik.

3. <célmondat> = "N220", <visszaugrási szintek száma> = 2



A `RET` utasítás után visszaugrás történik két programszinttel és a program feldolgozás az `N220` mondatban lesz folytatva.

2.24.2.11 Paraméterezhető alprogram-visszaugrás (RETB ...)

Általában a `RETB` utasítással egy alprogramból történik visszaugrás a felhívó programba. A feldolgozás ezután az alprogramhívást követő programsorral folytatódik. Ha a program feldolgozását egy másik helyen kell folytatni, a következő lehetőségek állnak rendelkezésre:

- programfeldolgozás folytatása leforgácsoló ciklusok felhívása után ISO-dialektus módban, a kontúrleírás után
- visszaugrás a főprogramba egy tetszőleges alprogram-szintről (ASUP után is) hiba-kezelésnél
- visszaugrás több programszinten át speciális alkalmazások számára Compile-ciklusokban és ISO-dialektus módban

Ehhez a `RETB` utasítást további paraméterekkel kell programozni.

Keresési irány

A `<célmondat>` paraméter megadásánál a visszaugrás először a felhívó mondat utáni mondatra történik. Ezután a cél annak a programnak a program**kezdete** irányában lesz keresve, ahova a visszaugrás lesz. Ha a keresés nem sikeres, a következőként a keresés a program vége irányában történik.

Szintaxis

```
RETB("<célmondat>")
RETB("<célmondat>",<célmondat utáni mondat>)
RET("<célmondat>",<mondat a célmondat után>,<visszaugrási szintek
száma>)
```

```
RETB("<célmondat>", ,<visszaugrási szintek száma>)
RETB("<célmondat>",<mondat a célmondat után>,<visszaugrási szintek
száma>,
<visszaugrás programkezdetre>)
RETB( , ,<visszaugrási szintek száma>,<visszaugrás programkezdetre>)
```

Jelentés

RETB:	Alprogramvég		
<célmondat>:	A paraméterben az ugráscélként az a mondat lesz megadva, ahol a programfeldolgozást folytatni kell Ha a <visszaugrási szintek száma> paraméter nincs programozva, akkor az ugráscél abban a programban van, amelyikből az aktuális alprogram fel lett hívva. Lehetséges megadások:		
	<mondatszám>	Célmondat száma A mondatszám keresése a programban, ahova vissza lesz ugorva, először a programkezdet irányában történik.	
	<ugrás-jelölő>	Ugrás-jelölő, amelyik jelen kell legyen abban a programban, ahova vissza lesz ugorva. Az ugrás-jelölő keresése a programban, ahova vissza lesz ugorva, először a programkezdet irányában történik.	
	<karakterlánc>	Karakterlánc, amelyik jelen kell legyen abban a programban, ahova vissza lesz ugorva (pl. program vagy változó név). A karakterlánc keresése a programban, ahova vissza lesz ugorva, először a programkezdet irányában történik. A karakterlánc programozására a célmondatban a következő szabályok érvényesek: • **Üres jel a végén** (megkülönböztetés az ugrás-jelelölőhöz, amelyik egy ":"-vel a végén van megjelölve). • **karakterlánc előtt** csak egy mondatszám és/vagy egy ugrás-jelölő lehet, **program utasítások nem**.	
<célmondat utáni mondat>:	A paraméterben meg lesz adva, hogy a program feldolgozása a <célmondat> paraméterben megadott mondatnál vagy az utána következő mondatnál legyen folytatva.		
	típus:	INT	
	érték:	0	A visszaugrás a <célmondat> paraméterben megadott mondatnál folytatódik.
		> 0	A visszaugrás a <célmondat> paraméterben megadott következő mondatnál folytatódik.

<visszaugrási szintek száma>:	A paraméterben megadásra kerül a programszintek száma, ahova vissza lesz ugorva, ott a célmondatot keresni és a program feldolgozását folytatni.		
	típus:	INT	
	érték:	1	a program az "aktuális programszint - 1"-en lesz folytatva (mint RET paraméter nélkül).
		2	a program az "aktuális programszint - 2"-en lesz folytatva, vagyis egy szint át lesz ugorva
		3	a program az "aktuális programszint - 3"-en lesz folytatva, vagyis két szint át lesz ugorva
		...	
	értéktartomány:	1 ... 15	
<visszaugrás programkezdetre>:	A paraméterben meg lesz adva, hogy a főprogramba visszaugrásnál és ott aktív **ISO dialektus mód** esetén a program a programkezdetnél legyen folytatva.		
	típus:	BOOL	
	érték:	1	Ha a visszaugrás a főprogramba történik és ott egy **ISO-dialektus modus** aktív, elágazás lesz a programkezdetre.

Megjegyzés

Egy alprogram-visszaugrásnál karakterlánccal a célmondat-keresés megadásaként a felhívó mondatban mindig először egy ugrás-jelölő után lesz keresve.

Ha az ugráscélt egy karakterlánccal egyértelműen kell definiálni, akkor a karakterlánc nem egyezhet egy ugrás-jelölő nevével, mert különben az alprogram-visszaugrás mindig az ugrás-jelölőre és nem a karakterláncra lesz végrehajtva (lásd példa 2).

Peremfeltételek

A visszaugrásnál több programszinten át az egyes programszintek SAVE utasításai ki lesznek értékelve.

Ha egy visszaugrásnál több programszinten át egy modális alprogram aktív és az átugrott alprogramok egyikében az MCALL kikapcsoló utasítás van programozva, a modális alprogram továbbra is aktív marad.

FIGYELEM
Programozási hiba
Egy visszaugrásnál több programszinten át kizárólag a felhasználóé felelősség arról gondoskodni, hogy a szükséges modális beállítások folytatódjanak. Ez pl. egy megfelelő főmondat programozásával érhető el.

Példa

Programkód	Kommentár
BEISPIEL.MPF	
…	
N3000 START_CYC(parm1, param2, …)	
N3010 TECH_CYC1(param1, param2, …)	
N3020 TECH_CYC2(param1, param2, …)	
N3030 TECH_CYC3(param1, param2, …)	
N3040 END_CYC(param1, param2, …)	
N3040 **END_CYC**(param1, param2, …)	
N3050 …	
N4500 START_CYC(param11, param12, …)	
N4510 …	
N4590 END_CYC(param11, param12, ..)	
N5000 …	
…	
N6000 M30	

Programkód	Kommentár
PROC END_CYC(…)	; felhívás a főprogramban sor N3040
N10000 …	
N15000 if status == 1	
N15010 RETB("START_CYC")	; visszaugrás a felhívó programba BEISPIEL.MPF
	; karakterlánc "START_CYC" keresése
	; keresési irány vissza a programkezdet irányába
	; program feldolgozás folytatása az N3000 sorral
N15020 endif	
N15030 if status == 0	
N15040 RET	; visszaugrás a felhívó programba BEISPIEL.MPF
	; program feldolgozás folytatása az N3050 sorral
N15050 endif	
N16000 RET("START_CYC")	; visszaugrás a felhívó programba BEISPIEL.MPF
	; karakterlánc "START_CYC" keresése
	; keresési irány előre programvég irányába
	; program feldolgozás folytatása az N4500 sorral
N17060 RETB	; visszaugrás a felhívó programba BEISPIEL.MPF
	; program feldolgozás folytatása az N3050 sorral
	; RETB paraméter nélkül azonos RET-tel

2.24.3 Egy alprogram felhívása

2.24.3.1 Alprogram hívása paraméter-átadás nélkül

Egy alprogram felhívása vagy az L címmel és az alprogramszámmal vagy egy programnév megadásával történik.

Egy főprogramot is fel lehet hívni alprogramként. A főprogramban levő `M2` vagy `M30` programvég ebben az esetben `M17`-ként (programvég visszaugrással a felhívó programba) lesz kiértékelve.

Megjegyzés

Természetesen egy alprogramot is el lehet indítani főprogramként.

Keresési stratégia a vezérlésben:

Létezik *_MPF?

Létezik *_SPF?

Ebből következik: Ha a felhívandó alprogram neve egyezik a főprogram nevével, akkor a felhívó főprogram lesz ismét felhívva. Ezt az általában nem kívánatos hatást az alprogramok és főprogramok egyértelmű névadásával el kell kerülni.

Megjegyzés

Alprogramokat, amelyek nem igényelnek paraméter-átadást, fel lehet hívni egy inicializálási fájlból is.

Szintaxis

```
L<szám>/<programnév>
```

Megjegyzés

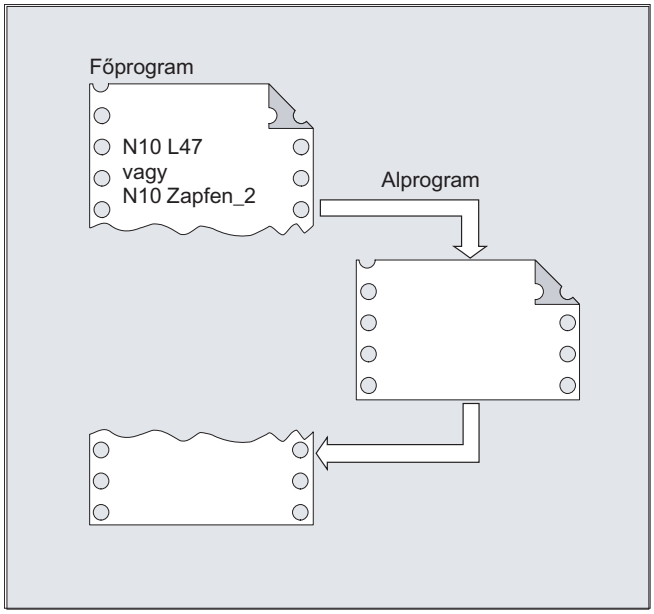
Egy alprogram felhívását egy külön NC-mondatban kell programozni..

Jelentés

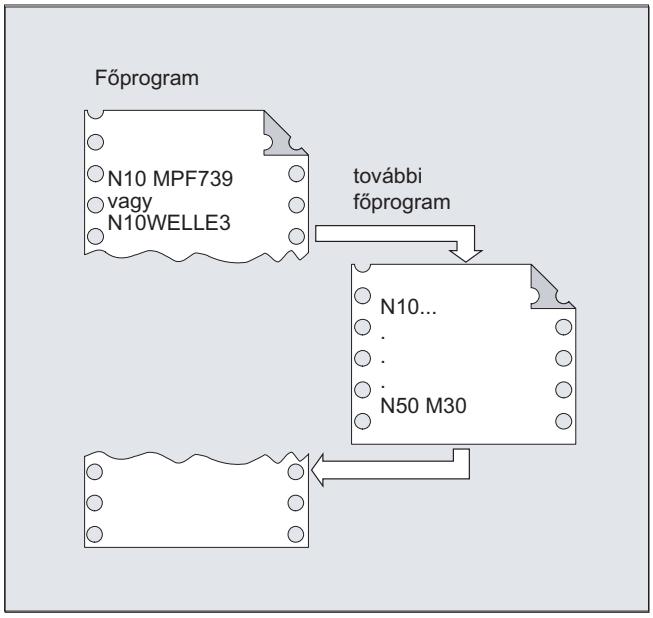
`L`:	cím az alprogram felhívásához	
`<szám>`:	alprogram száma	
	típus:	INT
	érték:	maximum 7 decimális hely **Figyelem:** A vezető nulláknak a névadásnál van jelentősége (⇒ L123, L0123 és L00123 három különböző alprogram).
`<programnév>`:	alprogram (vagy főprogram) neve	

Példák

Példa 1: Egy alprogram felhívása paraméter-átadás nélkül



Példa 2: Egy főprogram felhívása alprogramként



Lásd még

Alprogram paraméter-átadás nélkül (Oldal 162)

2.24.3.2 Alprogram felhívása paraméter-átadással (EXTERN)

Az alprogram felhívásánál paraméter-átadással a változókat vagy értékeket közvetlenül át lehet adni (VAR paramétereknél nem).

Alprogramokat paraméter átadással a felhívásuk előtt a főprogramban EXTERN-nel ismerté kell tenni (pl. a program elején). Meg kell adni az alprogram nevét és a változótípusokat az átadás sorrendjében.

FIGYELEM
Elcserélés veszély A változó-típusok és az átadás sorrendje meg kell egyezzen az alprogramban PROC-cal megadottakkal. A paraméternevek a fő-és alprogramban lehetnek különbözőek.

Szintaxis

```
EXTERN <programnév>(<Typ_Par1>,<Typ_Par2>,<Typ_Par3>)
...
<programnév>(<Wert_Par1>,<Wert_Par2>,<Wert_Par3>)
```

Megjegyzés

Az alprogram felhívását egy külön NC-mondatban kell programozni..

Jelentés

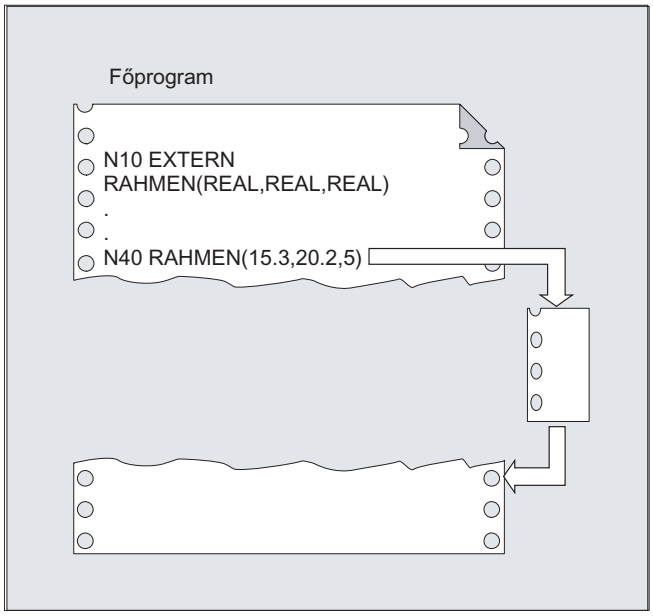
<programnév>:	alprogram neve
EXTERN:	kulcsszó egy paraméter-átadásos alprogram ismertté tételéhez **Utalás:** EXTERN-t csak akkor kell megadni, ha az alprogram a munkadarab- vagy a globális alprogram-könyvtárban van. A ciklusokat nem kell EXTERN-ként megadni..
<Typ_Par1>,<Typ_Par2>,<Typ_Par3>:	átadandó paraméterek változó-típusai az átadás sorrendjében
<érték_Par1>,<érték_Par2>,<érték_Par3>:	változó-értékek az átadandó paraméterekre

Példák

Példa 1: Alprogram felhívás előző megismertetéssel

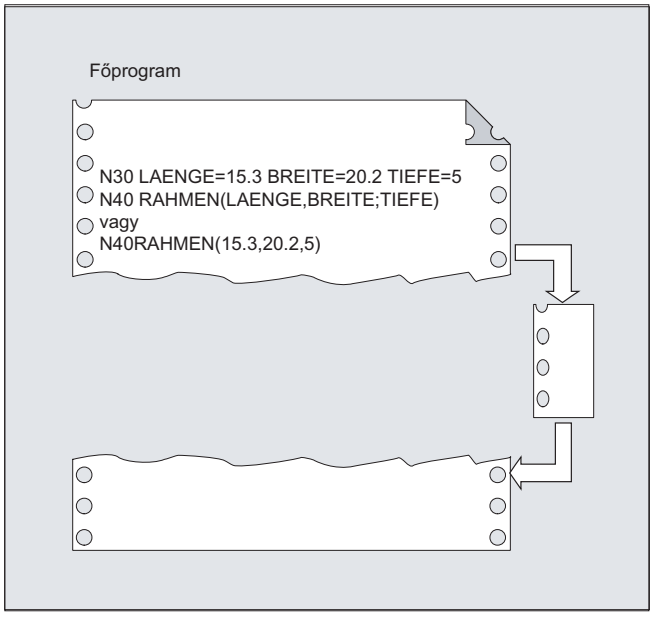
Programkód	Kommentár
N10 EXTERN RAHMEN(REAL,REAL,REAL)	; alprogram megadása
...	

Programkód	Kommentár
N40 RAHMEN(15.3,20.2,5)	; alprogram felhívása paraméter-átadással



Példa 2: Alprogram felhívás megismertetés nélkül

Programkód	Kommentár
N10 DEF REAL LAENGE, BREITE, TIEFE	
N20 …	
N30 LAENGE=15.3 BREITE=20.2 TIEFE=5	
N40 RAHMEN(LAENGE,BREITE,TIEFE)	; vagy: N40 RAHMEN(15.3,20.2,5)



Lásd még

Alprogram Call-by-Value paraméter-átadással (PROC) (Oldal 162)

Alprogram Call-by-Reference paraméter-átadással (PROC, VAR) (Oldal 164)

2.24.3.3 Program ismétlések száma (P)

Ha egy alprogramot többször egymás után fel kell dolgozni, az alprogram hívását tartalmazó mondatban a `P` címmel programozható a program ismétlések kívánt száma.

⚠ VIGYÁZAT
Alprogram felhívása program ismétléssel és paraméter-átadással A paraméterek csak a program hívásnál ill. az első átfutásnál kerülnek átadásra. Minden további ismétlésnél a paraméterek változatlanul maradnak. Ha a program ismétléseknél a paramétereket változtatni akarjuk, az alprogramokban kell a megfelelő intézkedéseket megtenni.

Szintaxis

```
<programnév> P<érték>
```

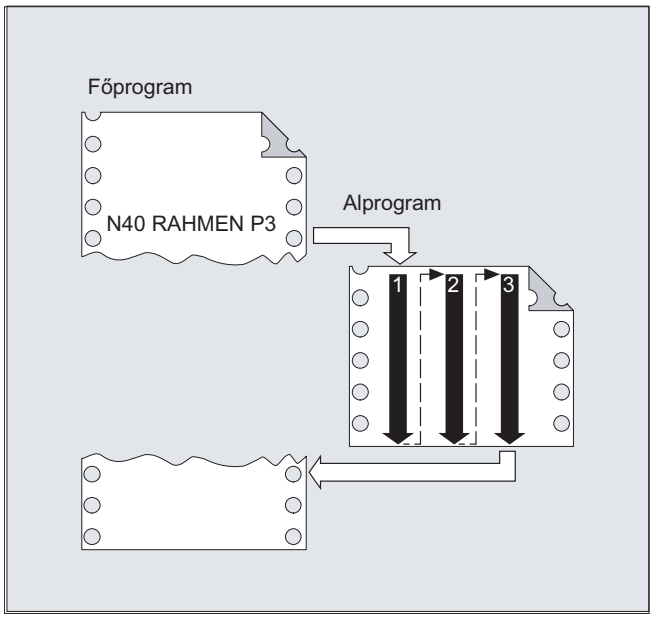
Jelentés

`<programnév>`:	alprogramhívás
`P`:	cím a program ismétlések programozására

<érték>:	program ismétlések száma	
	típus:	INT
	értéktartomány:	1 … 9999 (előjel nélkül)

Példa

```
Programkód              Kommentár
...
N40 RAHMEN P3           ; A RAHMEN alprogramot háromszor egymás után kell végrehajtani.
...
```



2.24.3.4 Modális alprogram-hívás (MCALL)

Az `MCALL(<programnév>)` modális alprogramhívással a megadott alprogram nem lesz rögtön felhívva. Ehelyett a felhívás ettől az időponttól a munkadarab-programban automatikus történik minden pályamozgásos mondat után. A programszinteken túllépve is.

Megjegyzés

Egy program lefutásban mindig csak az utolsó `MCALL(<programnév)` hat. Az aktuális modális alprogramhívás pótolja az eddig aktívat.

Ha paraméterek lesznek átadva az alprogramnak, ez a paraméter átadás csak az `MCALL(<programnév(Par1, Par2, ...))` felhívásánál történik meg.

FIGYELEM
Modális alprogram-hívás pályamozgások nélkül A modális alprogram a következő helyzetekben egy pályamozgás programozása nélkül is felhívásra kerül: • Az `S` vagy `F` címek programozásánál, ha `G0` vagy `G1` aktív. • Ha `G0` vagy `G1` egyedül áll a mondatban vagy további G utasításokkal lett programozva.

Szintaxis

```
MCALL <programnév>
...
MCALL
```

Jelentés

`MCALL <programnév>`:	"modális alprogram-hívás" funkció bekapcsolása
`<programnév>`:	alprogram neve
`MCALL`:	`MCALL`-lal egy programnév megadása nélkül a "modális alprogram-hívás" funkció ki lesz kapcsolva.

Peremfeltételek

ASUP

Ha egy munkadarab-program megmunkálását egy ASUP (lásd "Interrupt rutinok (ASUP) (Oldal 125)" fejezetben) megszakítja, ebben az ASUP -ban nem lesznek végrehajtva modális alprogramok.

Ha egy ASUP "Reset" csatorna állapotban lesz elindítva, ez a modális alprogramhívások vonatkozásában úgy viselkedik, mint egy normál munkadarabprogram.

Szerszámcsere ciklus

Ha a szerszámcsere ciklusban a "Modális alprogram hívás" funkció ki lesz kapcsolva, figyelembe kell venni, hogy a szerszámcsere ciklus esetleg a mondat keresés után közvetetten az ASUP kereséssel vagy kézzel áttárolással fel lesz hívva. Ebben a helyzetben nem szabad kikapcsolni a "Modális alprogram hívás" funkciót, mert különben a keresés eredménye hamis lesz. Ezért javasolt a "Modális alprogram-hívás" funkció kikapcsolását a szerszámcsere ciklusban a következők szerint programozni:

```
Programkód                Kommentár
...
  IF $AC_ASUP == 0        ; A felhívás nem ASUP kereséssel vagy áttárolással történik.
    MCALL                 ; "Modális alprogram hívás" funkciót kikapcsolni.
  ENDIF
```

```
Programkód                Kommentár
...
```

Példák

Példa 1

```
Programkód                Kommentár
N10 G0 X0 Y0
N20 MCALL L70             ; Modális alprogramhívást L70-re bekapcsolni.
N30 X10 Y10               ; X10 Y10 felvétele, utána L70 felhívása.
N40 X20 Y20               ; X20 Y20 felvétele, utána L70 felhívása.
...
N100 MCALL                ; "Modális alprogram hívás" funkciót kikapcsolni.
N110 X0 Y0                ; X0 Y0 felvétele, utána L70 nem lesz felhívva.
```

Példa 2

```
Programkód
N10 G0 X0 Y0
N20 MCALL L70
N30 L80
```

Ebben a példában a következő NC-mondatok a programozott pályatengelyekkel az L80 alprogramban vannak. Az L70-t az L80 hívja fel.

2.24.3.5 Közvetett alprogram-hívás (CALL)

Az adott feltételektől függően egy adott helyen különböző alprogramokat lehet felhívni. Ehhez az alprogram neve egy STRING típusú változóban lesz megadva. Az alprogram hívás `CALL`-lal és a változónévvel történik.

Megjegyzés

A közvetett alprogram-hívás csak a paraméter átadás nélküli alprogramoknál lehetséges. Egy alprogram közvetlen hívásához a nevét egy STRING-állandóban adjuk meg.

Szintaxis

```
CALL <programnév>
```

Jelentés

`CALL`:	cím a közvetett alprogram-hívásához	
`<programnév>`:	alprogram neve (változó vagy állandó)	
	típus:	STRING

Példa

Közvetlen hívás STRING állandóval:

Programkód	Kommentár
…	
CALL "/_N_WKS_DIR/_N_SUBPROG_WPD/_N_TEIL1_SPF"	; TEIL1 alprogram közvetlen hívása
…	

Közvetett felhívás változóval:

Programkód	Kommentár
…	
DEF STRING[100] PROGNAME	; változót definiálni
PROGNAME="/_N_WKS_DIR/_N_SUBPROG_WPD/_N_TEIL1_SPF"	; TEIL1 alprogramot a PROGNAME változóhoz rendelni
CALL PROGNAME	; TEIL1 alprogramot a PROGNAME változóval közvetetten felhívni
…	

2.24.3.6 Közvetett alprogram-hívás a végrehajtandó programrész megadásával (CALL BLOCK ... TO ...)

A CALL-lal és a BLOCK ... TO kulcs-kombinációval egy alprogram közvetetten fel lesz hívva és a kezdet és vég-jelölővel megjelölt programrész végre lesz hajtva.

Szintaxis

```
CALL <programnév> BLOCK <kezdet-jelölő> TO <vég-jelölő>
CALL BLOCK <kezdet-jelölő> TO <vég-jelölő>
```

Jelentés

CALL:	cím a közvetett alprogram-hívásához	
<programnév>:	alprogram neve (változó vagy állandó), amely a feldolgozandó programrészt tartalmazza (**megadás opcionális**).	
	típus:	STRING
	Utalás: Ha nincs <programnév> programozva, a <kezdet-jelölő> és <vég-jelölő> által megjelölt programrész az aktuális programban lesz keresve és végrehajtva.	
BLOCK ... TO ... :	kulcs-kombináció közvetett programrész végrehajtásra	
<kezdet-jelölő>:	változó, amely a feldolgozandó programrész kezdetére utal	
	típus:	STRING
<vég-jelölő>:	változó, amely a feldolgozandó programrész végére utal	
	típus:	STRING

Példa

Főprogram:

Programkód	Kommentár
...	
DEF STRING[20] STARTLABEL, ENDLABEL	; Változó definíció a kezdet- és a vég-jelölőre.
STARTLABEL="LABEL_1"	
ENDLABEL="LABEL_2"	
...	
CALL "CONTUR_1" BLOCK STARTLABEL TO ENDLA-BEL	; Közvetett alprogram-hívás és a végre-hajtandó programrész megadása.
...	

Alprogram:

Programkód	Kommentár
PROC CONTUR_1 ...	
LABEL_1	; kezdet-jelölő: programrész végrehajtás kezdete
N1000 G1 ...	
...	
LABEL_2	; vég-jelölő: programrész végrehajtás vége
...	

2.24.3.7 Egy ISO-nyelven programozott program közvetett felhívása (ISOCALL)

Az `ISOCALL` közvetett programhívással fel lehet hívni egy ISO-nyelven programozott programot. Ennél aktiválva lesz a gépadatban beállított ISO-mód. A program végén ismét az eredeti megmunkálási mód lesz hatásos. Ha a gépadatban nincs ISO-mód beállítva, az alprogram-hívás Siemens-módban történik.

További információk az ISO-módhoz, lásd:
Irodalom:
ISO-dialektusok működési leírás

Szintaxis

```
ISOCALL <programnév>
```

Jelentés

`ISOCALL`:	kulcsszó közvetett alprogram-hívásra, amivel a gépadatban beállított ISO-mód aktiválva lesz
`<programnév>`:	ISO-nyelven programozott program neve (STRING típusú változó vagy állandó)

Példa: Kontúr ciklusprogramozással ISO-módból felhívva

Programkód	Kommentár
0122_SPF	; kontúrleírás ISO-módban
N1010 G1 X10 Z20	
N1020 X30 R5	
N1030 Z50 C10	
N1040 X50	
N1050 M99	
N0010 DEF STRING[5] PROGNAME = "0122"	; Siemens munkadarabprogram (-ciklus)
...	
N2000 R11 = $AA_IW[X]	
N2010 ISOCALL PROGNAME	
N2020 R10 = R10+1	; 0122.spf programot ISO-módban feldolgozni
...	
N2400 M30	

2.24.3.8 Alprogramot ág-megadással és paraméterekkel felhívni (PCALL)

A PCALL-lal az alprogramokat abszolút ág-megadással és paraméter átadással lehet felhívni.

Szintaxis

```
PCALL <ág/programnév>(<paraméter 1>,…,<paraméter n>)
```

Jelentés

PCALL:	kulcsszó alprogram híváshoz abszolút ág-megadással
<ág/programnév>:	Abszolút ág-megadás alprogramnevet beleértve. Szabályok az ág-megadáshoz lásd "Programtároló fájljainak címzése (Oldal 215)". Ha nincs abszolút ág megadva, a PCALL úgy viselkedik, mint egy szabványos alprogram hívás programjelölővel. A programnév előtag és fájl-jelölő nélkül lesz megadva. Ha a programnevet előtaggal és fájl-jelölővel kell programozni, akkor kifejezetten előtaggal és fájl-jelölővel az EXTERN utasítással kell megadni.
<paraméter 1>, ...:	aktuális-paraméter az alprogram PROC utasításának megfelelően

Példa

Programkód
PCALL/_N_WKS_DIR/_N_WELLE_WPD/WELLE(parameter1,parameter2,…)

2.24.3.9 Keresőágat alprogram-hívásnál bővíteni (CALLPATH)

A CALLPATH utasítással lehet bővíteni a keresőágat az alprogram-hívásokhoz. Ezzel fel lehet hívni alprogramokat egy nem kiválasztott munkadarab-könyvtárból is, az alprogram teljes, abszolút ágnevének megadása nélkül is.

Egy további alkalmazási lehetőség adódik az "EES GDIR" nélkül EES üzemmódban, ha egy könyvtár egy külső programtárolón globális alprogramok tárolására lesz használva. Ebben az esetben a keresési ágat CALLPATH-szal ki lehet bővíteni ezzel az alprogram könyvtárral.

A keresőág bővítése az alkalmazói ciklusok megadása előtt történik (_N_CUS-DIR).

A keresőág bővítését a következő események kapcsolják ki:

- CALLPATH üres-jellel
- CALLPATH paraméter nélkül
- munkadarabprogram vége
- Reset

Szintaxis

```
CALLPATH("<ágnév>")
```

Jelentés

CALLPATH:	kulcsszó a programozható keresőág bővítéshez Egy külön programsorban kell programozni.
<ágnév>:	STRING típusú állandó vagy változó Egy könyvtár abszolút ág-megadását tartalmazza, amivel a kereső-ágat bővíteni kell. Szabályok az ág-megadáshoz lásd "Programtároló fájljainak címzése (Oldal 215)".

Példa

A kereső-ágat egy adott munkadarab-könyvtárral kell bővíteni:

Programkód

```
...
CALLPATH("/_N_WKS_DIR/_N_MYWPD_WPD")
...
```

Ezzel a következő keresőág lesz beállítva (5. pozíció új):

1. aktuális könyvtár/*név*
2. aktuális könyvtár/*név_SPF*
3. aktuális könyvtár/*név_MPF*
4. //NC:/_N_SPF_DIR/*név_SPF*
5. /_N_WKS_DIR/_N_MYWPD_WPD/név_SPF

6. /_N_CUS_DIR/*név_SPF*
7. /_N_CMA_DIR/*név_SPF*
8. /_N_CST_DIR/*név_SPF*

Peremfeltételek

- A `CALLPATH` megvizsgálja, hogy a programozott ágnév ténylegesen létezik-e. Hiba esetén a munkadarabprogram feldolgozása 14009 korrekciós-mondat vészjelzéssel meg lesz szakítva.
- A `CALLPATH` programozható INI-fájlokban is. Ekkor a az INI-fájl feldolgozási ideje alatt hatásos (WPD-INI-fájl vagy inicializálási program NC-aktív adatokra, pl. frame-kre a 1. csatornára _N_CH1_UFR_INI). Utána az inicializálási program ismét vissza lesz állítva

2.24.3.10 Külső alprogram feldolgozás (840D sl) (EXTCALL)

Az `EXTCALL` utasítással egy külső munkadarabprogramot lehet egy külső tárolóból utántölteni és feldolgozni.

Külső tárolóként a következők állnak rendelkezésre:

- helyi meghajtó
- hálózati meghajtó
- USB meghajtó

Megjegyzés

Egy USB meghajtón található külső program feldolgozásához interfészként csak a kezelőhelyen ill. TCU-n található USB interfész használható.

FIGYELEM

Szerszám / munkadarab károsodás USB-FlashDrive miatt

Egy külső alprogram feldolgozásához nem ajánlott egy USB-FlashDrive használata. A kommunikáció megszakadása a USB-FlashDrive-val a munkadarabprogram feldolgozása közben érintkezési problémák, kiesés, letörés meglökés miatt vagy véletlen kihúzás a megmunkálás azonnali megállását okozza. A munkadarab és/vagy szerszám ezáltal megsérülhet.

A külső programág beállítása

A külső program-könyvtár ágát előre be lehet állítani a következő beállítási adattal:

SD42700 $SC_EXT_PROG_PATH

Az `EXTCALL` hívásnál megadott program-ággal ill. -jelölővel együtt ebből adódik ki a felhívandó program teljes ága.

Megjegyzés

Paraméter

Egy külső program felhívásánál annak nem lehet paramétert átadni.

Szintaxis

```
EXTCALL("<ág/><programnév>")
```

Jelentés

EXTCALL:	utasítás egy külső alprogram felhívásához	
"<ág/><programnév>":	STRING típusú állandó/változó	
	<ág/>:	abszolút vagy relatív ágmegadás (**opcionális**)
	<programnév>:	A programnév az "_N_" előtag nélkül lesz megadva. A fájl-bővítményt ("MPF", "SPF") a "_" vagy "." karakterrel lehet a programnévhez hozzátoldani (**opcionális**). Példa: "WELLE" "WELLE_SPF" "WELLE.SPF"

Ág-megadás: rövid jelölés

Az ág-megadásnál a következő rövid jelöléseket lehet használni:

- helyi meghajtó: "**LOCAL_DRIVE:**"
- CF kártya: "**CF_CARD:**"
- USB meghajtó (kezelőhely előlap): "**USB:**"

A "CF_Card" és "LOCAL_DRIVE" rövidítéseket alternatívaként lehet használni.

Példa

Feldolgozás helyi meghajtóról

A "MAIN.MPF" főprogram az NC-tárolóban van és feldolgozásra ki van választva.

Alprogram "SP_1"

Az "SP_1.SPF" ill. "SP_1.MPF" külső alprogramok a helyi meghajtón "/user/sinumerik/data/prog/WKS.DIR/WST1.WPD" könyvtárban találhatóak.

A külső programkönyvtár ága következőképpen állítandó be:

SD42700 $SC_EXT_PROG_PATH = LOCAL_DRIVE:WKS.DIR/WST1.WPD

Megjegyzés

Ág-megadás a külső alprogram hívásához:

- Az elő-beállítás használata nélkül: "LOCAL_DRIVE:WKS.DIR/WST1.WPD/SP_1"
- Az elő-beállítás használatával: "SP_1"

"SP_2" alprogram

Az "SP_2.SPF" ill. "SP_2.MPF" alprogramok az USB meghajtó /WKS.DIR/WST1.WPD könyvtárában találhatók. A külső program-könyvtár ág elő-beállítása az "SP_1" alprogram ágához lesz használva és a főprogramban nem lesz átírva. Ezért az "SP_2" alprogram felhívásához a teljes ágat meg kell adni.

Főprogram "MAIN"

Programkód

```
N010 PROC MAIN
N020 ...
N030 EXTCALL("SP_1")
N030 EXTCALL("USB:WKS.DIR/WST1.WPD/SP_2")
N050 ...
N060 M30
```

További információk

EXTCALL hívás abszolút ág-megadással

Ha az alprogram a megadott ág alatt létezik, akkor az EXTCALL hívással végre lesz hajtva. Ha az alprogram a megadott ág alatt nem létezik, akkor az EXTCALL hívásra a program végrehajtás megszakad.

EXTCALL hívás relatív ág-megadással / ág-megadás nélkül

Egy EXTCALL hívás relatív ág-megadással / ág-megadás nélkül a létező programtárolók a következő minta szerint lesznek végig keresve:

1. Ha az SD42700 $SC_EXT_PROG_PATH-ban egy ág-megadás be van állítva, akkor először ebből az ágból kiindulva EXTCALL hívásban megadottak szerint (programnév esetleg relatív ág- megadással) lesz keresve. Az abszolút ág a következő karakterláncolással adódik:
   - Elő-beállított ág megadás az SD42700 $SC_EXT_PROG_PATH-ban
   - elválasztó jel "/"
   - Ág megadás és alprogramok az EXTCALL utasításban.
2. Ha az alprogram az 1. alatt nem található, a felhasználó tároló könyvtárai lesznek végig keresve.

A keresés az alprogram első megtalálásával végződik. Ha az alprogram nem létezik, akkor az EXTCALL hívásra a program végrehajtás megszakad.

Beállítható utántöltés-tároló (FIFO-puffer)

Egy külső alprogram feldolgozásához szükséges egy utántöltés tároló. Az utántöltés tároló nagysága 30 kbájtra van beállítva és mint a többi tároló-vonatkozású gépadatot, csak a gépgyártó változtathatja meg szükség esetén (MD18360 MM_EXT_PROG_BUFFER_SIZE).

Megjegyzés

Alprogramok ugrás utasításokkal

Ugrás utasításokat tartalmazó külső alprogramoknál (`GOTOF`, `GOTOB`, `CASE`, `FOR`, `LOOP`, `WHILE`, `REPEAT`, `IF`, `ELSE`, `ENDIF` stb.) az ugráscélok az utántöltő-tárolón belül kell legyenek.

Megjegyzés

ShopMill/ShopTurn programok

A ShopMill/ShopTurn programok a fájl végén beillesztett kontúr-leírás miatt teljesen az utántöltő-tárolón belül kell legyenek.

Párhuzamosan feldolgozott külső alprogramokhoz mind kell saját utántöltés tároló.

Reset / Programvég / POWER ON

A Reset és POWER ON által a külső alprogram-hívások meg lesznek szakítva és az utántöltési tároló törölve lesz.

A "Feldolgozás kívülről"-re kiválasztott alprogram a Reset, programvég vagy POWER ON után is "Feldolgozás kívülről"-re kiválasztva marad. A viselkedés nem különbözik a belül kiválasztott programokétól, mivel a külső programtároló továbbra is rendelkezésre áll.

Irodalom

További információk a "Feldolgozás kívülről"-höz, lásd:

Alapfunkciók működési kézikönyv; BAG, Csatorna, Program üzem, Reset viselkedés (K1)

2.24.3.11 Külső alprogramot feldolgozni (828D) (EXTCALL)

Az `EXTCALL` utasítással egy külső munkadarabprogramot lehet egy külső tárolóból utántölteni és feldolgozni.

Külső tárolóként a következők állnak rendelkezésre:

- felhasználói CF kártya
- hálózati meghajtók
- USB meghajtó

Megjegyzés

Egy külső USB meghajtón található programhoz interfészként csak a kezelőhely előlap (PPU) USB interfészét szabad használni.

FIGYELEM

Szerszám / munkadarab károsodás USB-FlashDrive miatt

Egy külső alprogram feldolgozásához nem ajánlott USB FlashDrive használata. A kommunikáció megszakadása a USB-FlashDrive-val a munkadarabprogram feldolgozása közben érintkezési problémák, kiesés, letörés meglökés miatt vagy véletlen kihúzás a megmunkálás azonnali megállását okozza. A munkadarab és/vagy szerszám ezáltal megsérülhet.

A külső programág beállítása

A külső program-könyvtár ágát előre be lehet állítani a következő beállítási adattal:

SD42700 $SC_EXT_PROG_PATH

Az `EXTCALL` hívásnál megadott program-ággal ill. -jelölővel együtt ebből adódik ki a felhívandó program teljes ága.

Megjegyzés

Paraméterek

Egy külső program felhívásánál annak nem lehet paramétert átadni.

Szintaxis

```
EXTCALL("<ág/><programnév>")
```

Jelentés

EXTCALL:	utasítás egy külső alprogram felhívásához	
"<ág/><programnév>":	STRING típusú állandó/változó	
	<ág/>:	abszolút vagy relatív ágmegadás (**opcionális**)
	<programnév>:	A programnév az "_N_" előtag nélkül lesz megadva. A fájl-bővítményt ("MPF", "SPF") a "_" vagy "." karakterrel lehet a programnévhez hozzátoldani (**opcionális**). Példa: "WELLE" "WELLE_SPF" "WELLE.SPF"

Ág-megadás: Rövid jelölés

Az ág-megadásnál a következő rövid jelöléseket lehet használni:

- felhasználói CF kártya: "**CF_CARD:**"
- USB meghajtó (kezelőhely előlap): "**USB:**"

Példa

A "MAIN.MPF" főprogram az NC-tárolóban van és feldolgozásra ki van választva.

"SP_1" alprogram

Az "SP_1.SPF" ill. "SP_1.MPF" alprogramok a felhasználói CF-kártyán az "/WKS.DIR/WST1.WPD" könyvtárban találhatók.

A külső programkönyvtár ága következőképpen állítandó be:

SD42700 $SC_EXT_PROG_PATH = CF_CARD:WKS.DIR/WST1.WPD

Megjegyzés

Ág megadása a külső alprogram megadásához:

- elő-beállítás alkalmazása nélkül: "CF_CARD:WKS.DIR/WST1.WPD/SP_1"
- elő-beállítás alkalmazásával: "SP_1"

"SP_2" alprogram

Az "SP_2.SPF" ill. "SP_2.MPF" alprogramok az USB meghajtó /WKS.DIR/WST1.WPD könyvtárában találhatók. A külső program-könyvtár ág elő-beállítása az "SP_1" alprogram ágához lesz használva és a főprogramban nem lesz átírva. Ezért az "SP_2" alprogram felhívásához a teljes ágat meg kell adni.

Főprogram "MAIN"

Programkód

```
N010 PROC MAIN
N020 ...
N030 EXTCALL("SP_1")
N030 EXTCALL("USB:WKS.DIR/WST1.WPD/SP_2")
N050 ...
N060 M30
```

További információk

EXTCALL hívás abszolút ág-megadással

Ha az alprogram a megadott ág alatt létezik, akkor az `EXTCALL` hívással végre lesz hajtva. Ha az alprogram a megadott ág alatt nem létezik, akkor az `EXTCALL` hívásra a program végrehajtás megszakad.

EXTCALL hívás relatív ág-megadással / ág-megadás nélkül

Egy `EXTCALL` hívás relatív ág-megadással / ág-megadás nélkül a létező programtárolók a következő minta szerint lesznek végig keresve:

1. Ha az SD42700 $SC_EXT_PROG_PATH-ban egy ág-megadás be van állítva, akkor először ebből az ágból kiindulva `EXTCALL` hívásban megadottak szerint (programnév esetleg relatív ág- megadással) lesz keresve. Az abszolút ág a következő karakterláncolással adódik:
   - Elő-beállított ág megadás az SD42700 $SC_EXT_PROG_PATH-ban
   - elválasztó jel "/"
   - Ág megadás és alprogramok az `EXTCALL` utasításban.
2. Ha az alprogram az 1. alatt nem található, a felhasználó tároló könyvtárai lesznek végig keresve.

A keresés az alprogram első megtalálásával végződik. Ha az alprogram nem létezik, akkor az `EXTCALL` hívásra a program végrehajtás megszakad.

Beállítható utántöltés-tároló (FIFO-puffer)

Egy külső alprogram feldolgozásához szükséges egy utántöltés tároló. Az utántöltő-tároló nagysága előre be van állítva (lásd MD18360 MM_EXT_PROG_BUFFER_SIZE).

Megjegyzés

Alprogramok ugrás utasításokkal

Ugrás utasításokat tartalmazó külső alprogramoknál (`GOTOF`, `GOTOB`, `CASE`, `FOR`, `LOOP`, `WHILE`, `REPEAT`, `IF`, `ELSE`, `ENDIF` stb.) az ugráscélok az utántöltő-tárolón belül kell legyenek.

Megjegyzés

ShopMill/ShopTurn programok

A ShopMill/ShopTurn programok a fájl végén beillesztett kontúr-leírás miatt teljesen az utántöltő-tárolón belül kell legyenek.

Párhuzamosan feldolgozott külső alprogramokhoz mind kell saját utántöltés tároló.

Reset / Programvég / POWER ON

A Reset és POWER ON által a külső alprogram-hívások meg lesznek szakítva és az aktuális utántöltés tároló törölve lesz.

A "Feldolgozás kívülről"-re kiválasztott alprogram a Reset, programvég vagy POWER ON után is "Feldolgozás kívülről"-re kiválasztva marad. A viselkedés nem különbözik a belül kiválasztott programokétól, mivel a külső programtároló továbbra is rendelkezésre áll.

Irodalom

További információk a "Feldolgozás kívülről"-höz, lásd:

Alapfunkciók Működési kézikönyv; BAG, csatorna, program-üzem, Reset viselkedés (K1)

2.25 Makró-technika (DEFINE...AS)

FIGYELEM
A makró-technika növeli a programozás komplexitását
A makró-technikával a vezérlés programozási nyelvét erősen meg lehet változtatni. A makró-technikát csak nagy gondossággal lehet alkalmazni.

Makrónak az egyes utasítások összefoglalását nevezzük egy új össz-utasítássá saját néven. A program lefutásnál a makró felhívásakor a makró neve alatt programozott utasítások egymás után végrehajtásra kerülnek.

Az érvényességi tartománynak, vagyis annak a tartománynak, amelyben a makró definíció hatásos, megfelelően a következő makró kategóriák vannak:

- lokális makrók
  A lokális makrók olyan makrók, amelyek egy olyan NC programban vannak definiálva, amely a feldolgozás időpontjában nem a főprogram. Ezek az NC program felhívásánál lesznek létrehozva és a program-vége Reset-tel ill. a következő vezérlés-felfutásnál törölve lesznek. A helyi makrókhoz csak az az NC program tud hozzáférni, amelyikben definiálva lett.
- program-globális makrók
  A program-globális makrók olyan makrók, amelyek egy olyan NC programban vannak definiálva, amely a feldolgozás időpontjában nem a főprogram. Ezek az NC program felhívásánál lesznek létrehozva és a program-vége Reset-tel ill. a következő vezérlés-felfutásnál törölve lesznek. A program-globális makrókhoz a főprogramban és az összes alprogramban hozzá lehet férni.

  Megjegyzés

  Program-globális makrók rendelkezésre állása

  A főprogramban definiált program-globális makrók csak akkor állnak rendelkezésre az alprogramokban is, ha a következő gépadat be van állítva:

  MD11120 $MN_LUD_EXTENDED_SCOPE = 1

  Az MD11120 = 0 értékével főprogramban definiált program-globális makrók csak a főprogramban állnak rendelkezésre.

- Globális makrók
  A globális makrók NC- ill. csatorna-globális makrók, amelyek egy definíciós fájlban (makró fájl) vannak definiálva és megmaradnak a program-vége Reset ill. a következő vezérlés felfutás után is. A globális makrókat bármelyik fő- vagy alprogramban fel lehet hívni és feldolgozni.

  Megjegyzés

  Egy **külső** makró-fájl makróinak használatához az NC programban a makró-fájl be kell legyen töltve az NC-be.

A makrókat az alkalmazásuk előtt definiálni kell. A következő szabályokat kell figyelembe venni:

- A makróban tetszőleges jelölőket, G, M, H funkciókat és L alprogramneveket lehet definiálni.
- A makró definíciója történhet a programkezdetnél vagy egy saját definíciós fájlban (makró fájl).
- A lokális és program-globális makrók a programkezdetnél vannak definiálva.
- A globális makrókat egy makró fájlban, pl. _N_DEF_DIR/_N_UMAC_DEF, kell definiálni.
- A G utasítás makrókat csak globális makrókként lehet definiálni.
- A H és L funkciók 2-jegyűen programozhatók.
- Az M és G utasításokat 3-jegyűen lehet programozni.

Megjegyzés

A kulcsszavakat és a foglalt neveket szem szabad átírni makró definícióval.

Szintaxis

Makró definíció:
```
DEFINE <Macro_Name> AS <Operation_1> <Operation_2> ...
```

Felhívás az NC-programban
```
<Macro_Name>
```

Jelentés

`DEFINE ... AS`:	kulcsszó egy makró definiálásához
`<Macro_Name>`:	a makró neve Makrónévként csak jelölők megengedettek. A makró a makrónévvel lesz felhívva az NC programból.
`<Operation_1>`:	első program-utasítás a makróban
`<Operation_2>`:	második program-utasítás a makróban

Peremfeltételek

Skatulyázási mélység

A makrók egymásba skatulyázása nem lehetséges.

Példák

Példa 1: Makró definíció a program elején

Programkód	Kommentár
`DEFINE LINIE AS G1 G94 F300`	`; makró-definíció`
`...`	

Programkód	Kommentár
N70 LINIE X10 Y20	; makró felhívása
...	

Példa 2: Makró definíció egy makró-fájlban

Programkód	Kommentár
DEFINE M6 AS L6	; A szerszámváltásnál egy alprogram kerül felhívásra, ami a szükséges adatátvitel elvégzi. Az alprogramban lesz a tulajdonképpeni szerszámváltás M-funkció kiadva (pl. M106).
DEFINE G81 AS DRILL(81)	; DIN G utasítás utánzása:
DEFINE G33 AS M333 G333	; A menetvágásnál szükséges a szinkronizáció a PLC-vel. Az eredeti G33 G utasítás MD gépadatban G333-ra lett átnevezve, a programozás a felhasználónak azonos marad.

Példa 3: Külső makró-fájl

A külső makró-fájlnak a vezérlésbe beolvasása után a makró-fájlt be kell tölteni az NC-be. Csak ezután lehet a makrókat az NC programban használni.

Programkód	Kommentár
%_N_UMAC_DEF	
;$PATH=/_N_DEF_DIR	; vevő-specifikus makrók
DEFINE PI AS 3.14	
DEFINE TC1 AS M3 S1000	
DEFINE M13 AS M3 M7	; orsó jobbra, hűtőszer be
DEFINE M14 AS M4 M7	; orsó balra, hűtőszer ki
DEFINE M15 AS M5 M9	; orsó állj, hűtőszer ki
DEFINE M6 AS L6	; szerszámcsere-program felhívása
DEFINE G80 AS MCALL	; fúróciklus kiiktatása
M30	

Fájl- és program-kezelés

3

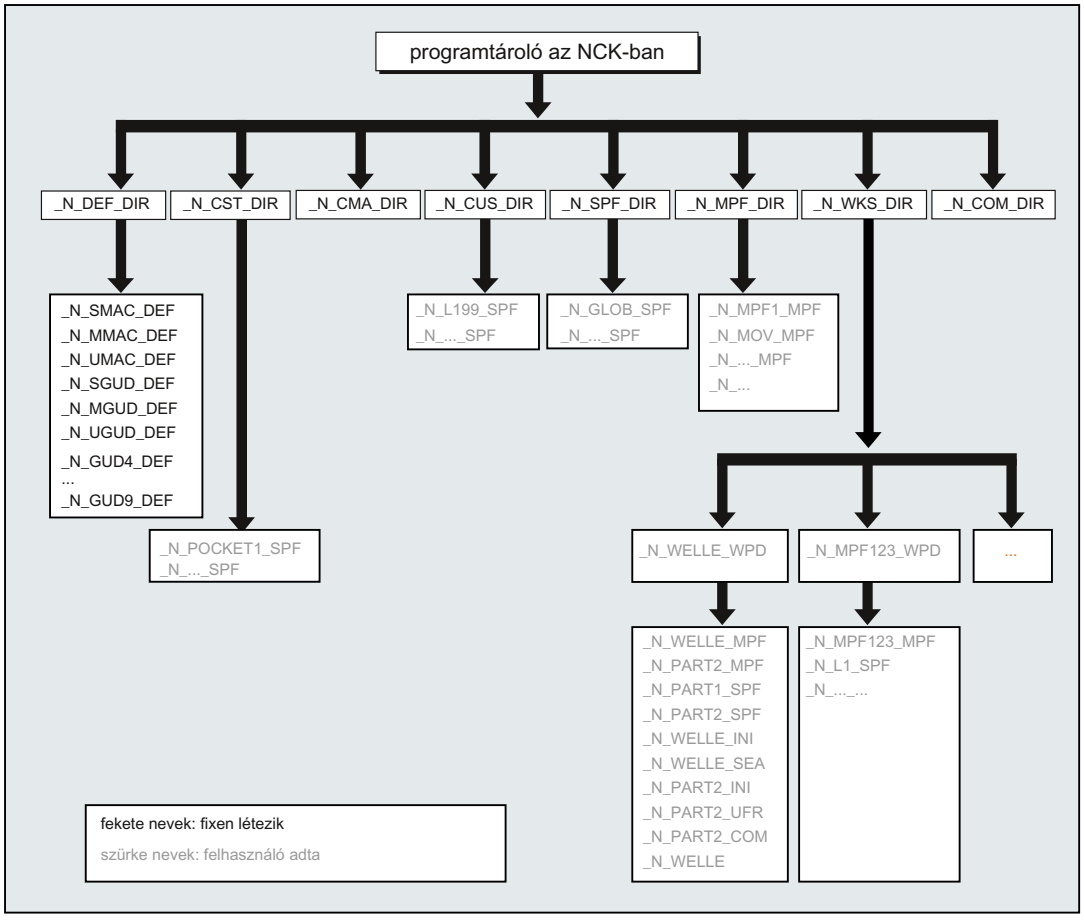
3.1 Programtároló

3.1.1 Programtároló az NC-ben

A programtárolóban vannak a fájlok és programok (pl. fő- és alprogramok, makró-definíciók) tartósan elhelyezve (→passzív fájlrendszer).

Irodalom:
Bővítő funkciók működési kézikönyv; Tároló konfiguráció (S7)

Emellett vannak olyan fájltípusok, amelyeket itt lehet átmenetileg tárolni és szükség esetén (pl. egy adott munkadarab megmunkálásánál) a munkatárolóba át lehet vinni (pl. inicializálási célokra).



Szabványos-könyvtárak

A következő könyvtárak szabványosan léteznek:

Könyvtár	Tartalom
_N_DEF_DIR	Adat-modulok és makró-modulok
_N_CST_DIR	Szabvány ciklusok
_N_CMA_DIR	Gyártói ciklusok
_N_CUS_DIR	Felhasználói ciklusok
_N_WKS_DIR	Munkadarabok
_N_SPF_DIR	Globális alprogramok
_N_MPF_DIR	Főprogramok
_N_COM_DIR	Kommentárok

Fájltípusok

A programtárolóban a következő fájl típusok lehetnek bevive:

Fájltípus	Leírás
<név>_MPF	Főprogram
<név>_SPF	Alprogram
<név>_TEA	Gépadatok
<név>_SEA	Beállítási-adatok
<név>_TOA	Szerszámkorrekciók
<név>_UFR	Nullaponteltolások/frame-ek
<név>_INI	Inicializálási fájlok
<név>_GUD	Globális felhasználói adatok
<név>_RPA	R-paraméterek
<név>_COM	Kommentár
<név>_DEF	Definíciók a globális felhasználói adatokhoz és makrókhoz

Munkadarab-főkönyvtár (_N_WKS_DIR)

A munkadarab-főkönyvtár alaphelyzetben az _N_WKS_DIR jelölővel van a programtárolóban létrehozva. A munkadarab-főkönyvtár minden programozott munkadarabhoz tartalmazza a megfelelő munkadarab-könyvtárakat.

Munkadarab-könyvtárak (..._WPD)

Egy munkadarab-könyvtár minden fájlt tartalmaz, ami egy munkadarab megmunkálásához szükséges. Ezek lehetnek főprogramok, alprogramok, tetszőleges inicializálási-programok és kommentár-fájlok.

Az inicializálási programok a program-kiválasztás után az első munkadarabprogram indításával egyszer lesznek végrehajtva (az MD11280 $MN_WPD_INI_MODE gépadatnak megfelelően).

Példa:

Az _N_WELLE_WPD munkadarab-könyvtár, ami a WELLE munkadarabhoz lett létrehozva, a következő fájlokat tartalmazza:

Fájl	Leírás
_N_WELLE_MPF	Főprogram
_N_PART2_MPF	Főprogram
_N_PART1_SPF	Alprogram
_N_PART2_SPF	Alprogram
_N_WELLE_INI	Általános inicializáló program a munkadarab adataihoz
_N_WELLE_SEA	Beállítási adatok inicializáló program
_N_PART2_INI	Általános inicializáló program a PART2 program adataihoz
_N_PART2_UFR	Inicializáló program a PART2 program frame-adataihoz
_N_WELLE_COM	Kommentár fájl

Egy munkadarab-könyvtárban további adatokat is lehet tárolni, amelyek nem szükségesek közvetlenül az NC általi feldolgozáshoz. Ezek lehetnek az ASCII fájlok mellett bináris fájlok is, mint pl. képek JPG formátumban vagy leírások PDF formátumban. Hogy ezeket az NC bináris fájlokként értelmezni tudja, a fájl bővítményeknek ismertek kell legyenek az NC-ben (beállítás az üzembehelyezésnél az MD17000 $MN_ EXTENSIONS_OF_BIN_FILES által; az alap beállításban a következő fájl bővítmények vannak megadva: JPG, GIF, PNG, BMP, PDF, ICO, HTM).

Munkadarabot megmunkálásra kiválasztani

Egy munkadarab-könyvtárat ki lehet választani megmunkáláshoz egy csatornában. Ha ebben a könyvtárban egy **azonos nevű** főprogram vagy csak egyetlen főprogram (_MPF), található, akkor ez automatikusan ki lesz választva megmunkálásra.

Irodalom:
Kezelési kézikönyv

3.1.2 Külső programtároló

A passzív fájlrendszer mellett az NC-ben egy gépen rendelkezésre állhatnak külső programtárolók is (pl. a helyi meghajtón vagy egy hálózati meghajtón).

A "Feldolgozás kívülről" vagy az "EES (Execution from External Storage)" funkciókkal a munkadarab-programokat **közvetlenül** külső programtárolóból lehet feldolgozni.

Irodalom:
Működési kézikönyv Alapfunkciók; K1: BAG, csatorna, program-üzem, Reset viselkedés

Globális munkadarabprogram-tároló (GDIR)

A meghajtók megadásánál az egyik meghajtót ki lehet jelölni globális munkadarabprogram-tárolónak (GDIR).

Irodalom:
Kezelési kézikönyv; Fejezet: "Programok kezelése" > "Meghajtók létrehozása"

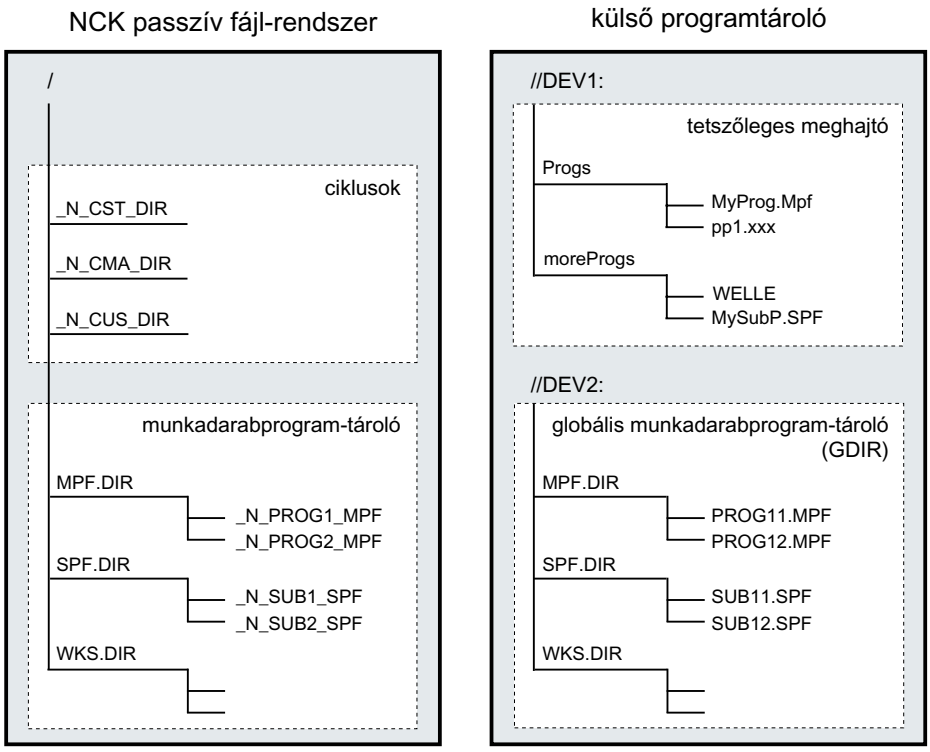
A rendszer ekkor automatikus létrehozza az MPF.DIR, SPF.DIR és WKS.DIR könyvtárak a meghajtón. Ez a három könyvtár alkotja a GDIR-t.

A GDIR kizárólag az EES funkciónál játszik szerepet. A meghajtók konfigurációjától függően a GDIR helyettesíti vagy pótolja az NC munkadarabprogram-tárolót. Egy GDIR létrehozása az EES üzemhez nem feltétlenül szükséges.

A GDIR könyvtárait és fájljait a munkadarab-programban ugyanúgy kell címezni, mint a passzív fájl-rendszerben. Ezzel lehetséges egy ág-megadásos NC program kompatibilis áthelyezése a passzív fájl-rendszerből a GDIR-be. A GDIR SPF.DIR könyvtára szerepel a keresési-ágban az alprogramokra.

Program szervezés

A következő ábra szemlélteti a program szervezést a külső programtárolón:



Case-insensitive fájl-rendszer

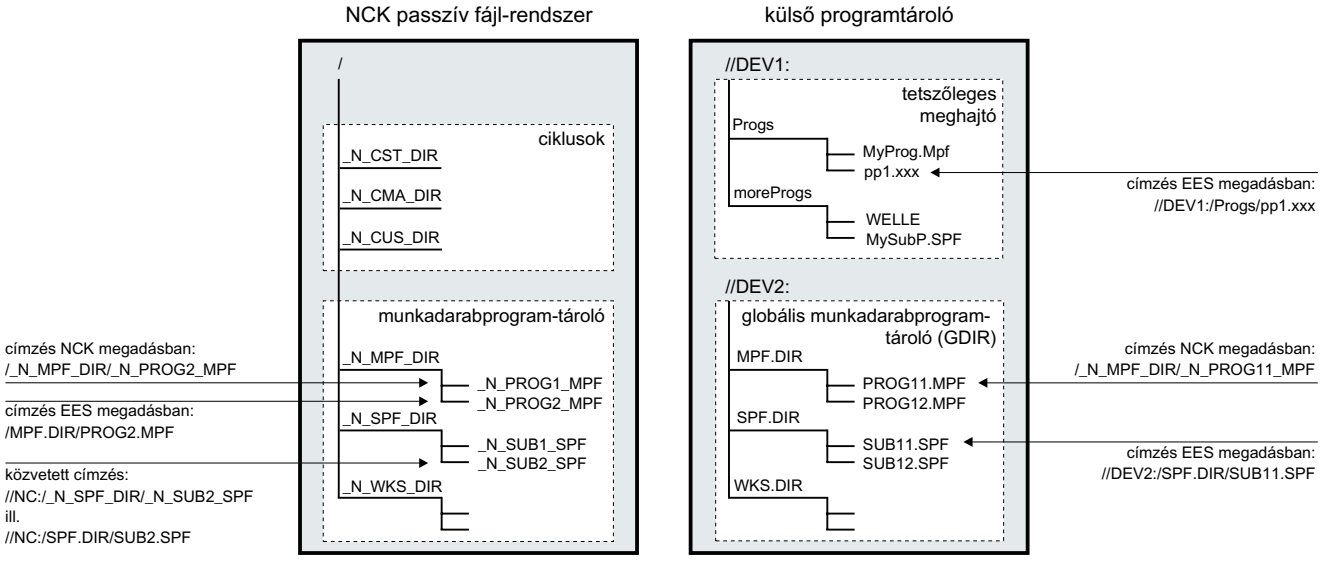
Megjegyzés

A nagy- és kisbetűs írás problémájának elkerülésére a fájl címzésnél (lásd "Programtároló fájljainak címzése (Oldal 215)"), **case-insensitive** fájl-rendszereket kell alkalmazni külső programtárolóként.

3.1.3 Programtároló fájljainak címzése

Egy fájl a programtárolóban, amelyet egy fájl-kezelő utasítás (pl. WRITE, DELETE, READ, ISFILE, FILEDATE, FILETIME, FILESIZE, FILESTAT, FILEINFO) megszólít, vagy egy abszolút ág plusz fájlnév vagy csak a fájlnév határoz meg. A második esetben a kiválasztott program ága lesz fájl-ágként használva.

Címzés NC/EES megadásban



Passzív fájlrendszer fájljainak címzése

A passzív fájlrendszer fájljainak címzése szokásosan az **NC címzésel** történik (könyvtár- és fájlnevek az "_N_" tartomány jelöléssel kezdődnek, elválasztó jel a könyvtár-/fájl-bővítményhez "_") a meghajtó nevének megadása nélkül. Azonban a címzés **EES címzés**-ben ("_N_" tartomány-jelölő nélkül, elválasztó jel a könyvtár-/fájl-bővítményhez ".") is megengedett.

Példa:

- NC címzés: "/_N_SPF_DIR/_N_SUB1_SPF"
- EES címzés: "/SPF.DIR/SUB1.SPF"

Megjegyzés

A passzív fájlrendszer fájljainak címzése az EES címzésben belül mindig át lesznek alakítva NC címzésre.

- Könyvtár- és fájlnevek az "_N_" tartomány jelölővel lesznek kibővítve.
- Ha a hátulról negyedik karakter a könyvtár- ill. fájlnéven egy pont ("."), az át lesz alakítva alsóvonalra ("_").

Az "//NC:" előre definiált meghajtó-névvel a passzív fájl rendszert célzottan is lehet címezni.

Példa:

- NC címzés: "//NC:/_N_SPF_DIR/_N_SUB1_SPF"
- EES címzés: "//NC:/SPF.DIR/SUB1.SPF"

Külső programtároló fájljainak címzése

Egy külső programtároló fájljainak címzése, amely nem GDIR, EES címzésben kell történjen. A címzési ág elején meg kell adni a meghajtó nevet (pl. "//DEV1:"). Megengedett az összes, a /user/sinumerik/hmi/cfg/logdrive.ini-ben megadott készüléknév.

Példa:

- EES címzés: "//DEV1:/MyProgDir/pp1.xxx"
- NC címzés: nem megengedett

Globális munkadarabprogram-tároló (GDIR) fájljainak címzése

A GDIR fájljainak címzésénél az EES címzés ág-megadás mellett az NC címzés ág-megadás is megengedett.

Példa:

- EES címzés: "//DEV2:/MPF.DIR/PROG11.MPF"
- NC címzés: "/_N_MPF_DIR/_N_PROG11_MPF"

Megjegyzés

A GDIR fájlrendszer fájljainak címzése az NC címzésben belül a következő szabályok szerint lesznek átalakítva EES címzésre:

- Az "_N_" tartomány-jelölő a könyvtár- és fájlnevekben el lesz távolítva.
- Ha a hátulról negyedik karakter a könyvtár- ill. fájlnéven egy pont ("_"), az át lesz alakítva alsóvonalra (".").

Szabályok az ág-megadáshoz

Egy teljes ág-megadás meghajtó-névből, könyvtár-ágból és fájlnévből áll.

Meghajtó-név

A meghajtó-név megadására a következő szabályok érvényesek:

- Megengedett az összes, a /user/sinumerik/hmi/cfg/logdrive.ini-ben megadott készüléknév.
- Az elején áll a "//" jel, követi legalább egy betű vagy számjegy.
- A következő karakterek lehetnek betűk, számjegyek, "_" és üres jelek tetszőleges kombinációi.
- A név lezárása egy betűvel vagy számjeggyel történik, követve egy ":" karakterrel:
- További különleges karakterek nem megengedettek.

Megjegyzés

A passzív fájl-rendszerre az "//NC:" meghajtó-név előre definiált.

Példák:

- Külső programtároló:
  - //Drive1:
  - //Drive_1:
  - //Drive 1:
  - //A B:
  - //1 B C 2:

Könyvtár-ág

A könyvtár-ág megadására a következő szabályok érvényesek:

- A könyvtár-ág elején és végén és az egyes ág-részek elválasztására a "/".

  Megjegyzés

  Egy dupla ferde-vonal ("//") a könyvtár-ágon belül **nem** megengedett!

- Könyvtár-nevek:
  - A könyvtár-nevek egy betűvel vagy számjeggyel kell kezdődjenek: Csak az NC címzésű címzésnél kezdődnek a könyvtár-nevek a "_N_" tartomány jelölővel.
  - A következő karakterek lehetnek betűk, számjegyek és "_" tetszőleges kombinációi.

    Megjegyzés

    Külső programtárolóknál a könyvtár-névben megengedettek az üres jelek is. Ez azonban nem érvényes, ha a külső programtároló globális munkadarabprogram-tárolóként (GDIR) van létrehozva.

  - További különleges karakterek nem megengedettek.
- Könyvtár bővítmények:
  - A könyvtár bővítmények pontosan három betűből/számjegyből kell álljanak.
  - Ezek "_"-vel (NC címzés) ill. "."-vel (EES címzés) lesznek a könyvtár-nevektől elválasztva.

  Megjegyzés

  A passzív fájl-rendszerben csak _DIR és _WPD könyvtár-bővítmények vannak.

Példák:

- Passzív fájl-rendszer ill. GDIR:
  - NC címzés: _N_WKS_DIR/_N_MYNCPROGS_WPD/...
  - EES címzés: WKS.DIR/MYPROGS.WPD/...
- Külső programtároló:
  - /abc
  - /ab_c.def
  - /ab c1.def
  - /a b c .d11
  - /abc.def/ghi.klm

Fájlnév

A fájlnevekre a következő szabályok érvényesek:

- Csak az NC címzésű címzésnél kezdődnek a fájl-nevek a "_N_" tartomány jelölővel.
- Az két következő karakter két betű vagy egy alsóvonal után egy betű kell legyen.

  Megjegyzés

  Ha ez a feltétel tejesül, egy NC programot a programnév megadásával egy másik programból alprogramként lehet hívni. Ha azonban a programnév számmal kezdődik, akkor az alprogram hívás csak CALL utasítással lehetséges.

- A következő karakterek lehetnek betűk, számjegyek és "_" tetszőleges kombinációi.
- Fájl-bővítmény:
  - A fájl-bővítmény pontosan három betűből/számjegyből kell álljon.

    Megjegyzés

    Megengedett fájl-bővítmények a passzív fájl-rendszerbenProgramtároló az NC-ben (Oldal 211)

  - Ez "_"-vel (NC címzés) ill. "."-vel (EES címzés) lesz a fájl-névtől elválasztva.

Példák:

- Passzív fájl-rendszer ill. GDIR:
  - NC címzés: _N_SUB1_SPF
  - EES címzés: SUB1.SPF
- Külső programtároló:
  - Teil1
  - _Teil1
  - Teil_1.spf
  - Teil1.mpf

DIN alprogram-nevek

A DIN alprogram-nevekre a következő szabályok érvényesek:

- Az első karakter az "L" betű.
- A következő karakterek számjegyek (legalább egy).
- Fájl-bővítmény:
  - A fájl-bővítmény pontosan három betűből kell álljon.
  - Ez "_"-vel (NC címzés) ill. "."-vel (EES címzés) lesz a fájl-névtől elválasztva.

Példák:

- L123
- L1_SPF (NC címzés) ill. L1.SPF (EES címzés)

Maximális ág-hossz

A meghajtó-név és a könyvtár-ág megadására maximum 128 bájt áll rendelkezésre, a fájlnév hossza maximum 31 bájt lehet. A teljes ágra a maximális hossz 159 bájt.

3.1.4 Kereső-ág alprogramhívásnál

Az alprogramhívásnál ág-megadás nélkül az abszolút ág egy fix kereső-ág feldolgozásával lesz megállapítva.

A programtárolók a következő sorrendben lesznek megvizsgálva:

<table>
<tr><th colspan="2"></th><th>Könyvtár</th><th>Leírás</th></tr>
<tr><td colspan="2">1</td><td>aktuális könyvtár / név</td><td rowspan="3">Az aktuális könyvtár az a könyvtár, amelyikben a program választás történt.
Ez lehet:
• egy munkadarab-könyvtár vagy az _N_MPF_DIR szabványos könyvtár az NC munkadarabprogram-tárolóban vagy a globális munkadarabprogram-tárolóban
vagy
• egy külső programtároló tetszőleges könyvtára</td></tr>
<tr><td colspan="2">2</td><td>aktuális könyvtár/név_SPF</td></tr>
<tr><td colspan="2">3</td><td>aktuális könyvtár/név_MPF</td></tr>
<tr><td rowspan="2">4</td><td>a</td><td>//NC:/_N_SPF_DIR/név_SPF</td><td>alprogram könyvtár az NC munkadarabprogram-tárolóban</td></tr>
<tr><td>b</td><td>//DEV2:/_N_SPF_DIR / név_SPF [1)]</td><td>alprogram könyvtár a globális munkadarabprog-ram-tárolóban
Utalás:
Ez a keresési lépés kimarad, ha nincs globális munkadarabprogram-tároló létrehozva vagy a program választás az NC munkadarabprogram-tárolóban történt.</td></tr>
</table>

	Könyvtár	Leírás
5	CALLPATH-szal programozott kereső-ág bővítés (lásd "Keresőágat alprogram-hívásnál bővíteni (CALLPATH) (Oldal 199)"). **Utalás:** Ez a keresési lépés kimarad, ha nincs CALLPATH programozva.	
6	/_N_CUS_DIR / *név_SPF*	felhasználói ciklusok könyvtár
7	/_N_CMA_DIR / *név_SPF*	gyártói ciklusok könyvtár
8	/_N_CST_DIR / *név_SPF*	szabvány ciklusok könyvtár

1) //DEV2:" például a meghajtó, amelyiken a globális munkadarabprogram-tároló létre van hozva.

A keresésre a következő szabályok érvényesek:

- A kereső-ág minden egyes alprogramhívásra végig lesz futva, vagyis nem számít, hol található a fölérendelt program.
- A könyvtáraktól függően különféle fájl-típusok lesznek figyelembe véve.
- Alapvetően egy könyvtárban nem lesz keresés az alárendelt, vagyis a skatulyázott könyvtárakban.

3.1.5 Az ág és a fájlnév lekérdezése

Egy NC program ág és fájlnév lekérdezéséhez a következő, a munkadarabprogramban olvasható rendszerváltozók állnak rendelkezésre:

Rendszerváltozó	Típus	Jelentés
$P_STACK	INT	Megadja a program-szintet, ahol az aktuális NC program feldolgozása fut.
$P_PATH[<n>]	STRING	Megadja az NC program ágát, amelyik az <n> mező-index által kiválasztott program-szinten feldolgozásra kerül. Példák: $P_PATH[0] megadja a főprogram ágát, pl. "/_N_WKS_DIR/_N_WELLE_WPD/". $P_PATH[$P_STACK - 1] megadja a felhívó program ágát. Ha az ág egy olyan NC programra vonatkozik, amelyik az NC passzív fájl-rendszerében vagy a globális munkadarabprogram-tárolóban (GDIR) van elhelyezve, az ág megadása NC címzésben történik. Ha az ág egy olyan NC programra vonatkozik, amelyik egy másik külső programtárolóban, mint a globális munkadarabprogram-tároló van elhelyezve, az $P_PATH az ágat EES címzésben adja meg.
$P_PROG[<n>]	STRING	Megadja az NC program nevét, amelyik az <n> mező-index által kiválasztott program-szinten feldolgozásra kerül. Ha az NC programra az NC passzív fájl-rendszerében vagy a globális munkadarabprogram-tárolóban van elhelyezve, a programnév megadása NC címzésben történik. Ha az NC program egy másik külső programtárolóról, mint a globális munkadarabprogram-tároló lesz feldolgozva, az $P_PROG az ágat EES címzésben adja meg.

Rendszerváltozó	Típus	Jelentés	
$P_PROGPATH	STRING	Megadja az NC program ágát, amelyik éppen feldolgozásra kerül. Az $P_PROGPATH felhívása azonos a $P_PATH[$P_STACK]-ével.	
$P_IS_EES_PATH[<n>]	BOOL	Lekérdezés, hogy az $P_PATH[<n>] által megadott ág ill. az $P_PROG[<n>] megadott programnév az NC címzésnek vagy az EES címzésnek felel-e meg.	
		= FALSE	$P_PATH[<n>] és $P_PROG[<n>] NC címzést adnak. Azaz a jelölő előtt egy "_N_" előtag van. A fájl jellemző elválasztó jele az "_". Példák: • ág NC címzésben: "/_N_WKS_DIR/_N_MYWPD_WPD/" • programnév NC címzésben:"_N_MYPROG_MPF" Egy ág NC címzésben vonatkozhat az NC passzív fájl-rendszerére és a globális munkadarabprogram-tárolóra is.
		= TRUE	$P_PATH[<n>] és $P_PROG[<n>] EES címzést adnak. Azaz a jelölő előtt nincs "_N_" előtag van. A fájl jellemző elválasztó jele az ".". Példák: • ág EES címzésben: "//DEV1:/WKS.DIR/MYWPD.WPD/" • Programnév EES címzésben: "MYPROG.MPF"

<n>: Az index <n> definiálja a programszintet, amelyikből az ág információt ki kell olvasni (értéktartomány: 0 ... 17)

Megjegyzés

Az EES üzemben a globális munkadarabprogram-tárolón (GDIR) kívül a $P_PROG, $P_PATH, és $P_PROGPATH ág-nevek EES címzésben lesznek megadva. A felhasználói programokat, amelyek ezeket az ág-neveket kiértékelik és tovább feldolgozzák, az EES üzemhez úgy kell kibővíteni, hogy az ág-neveket EES címzésben is fel tudják dolgozni.

3.2 Munkatároló (CHANDATA, COMPLETE, INITIAL)

Funkció

A munkatároló tartalmazza az aktuális rendszeradatokat és alkalmazói adatokat, amelyekkel a vezérlés dolgozik(aktív fájlrendszer), pl.:

- Aktív gépadatok
- Szerszámkorrekció-adatok
- Nullaponteltolások
- ...

Inicializáló programok

Ezek olyan programok amelyekkel a munkatároló adatainak elő-beállítása (inicializálása) történik. Ehhez a következő fájltípusok alkalmazhatók:

Fájltípus	Leírás
name_TEA	Gépadatok
name_SEA	Beállítási-adatok
name_TOA	Szerszámkorrekciók
name_UFR	Nullaponteltolások/frame-ek
name_INI	Inicializálási fájlok
name_GUD	Globális felhasználói adatok
name_RPA	R-paraméterek

Adat-tartományok

Az adatokat különböző tartományokba lehet besorolni, ahol érvényesek. Például egy vezérlésnek több csatornája lehet vagy szokásos a több tengely is.

A következők léteznek:

Jelölő	Adat-tartományok
NC	NC-specifikus adatok
CH<n>	NCK-specifikus adatok (<n> a csatornaszámot adja meg)
AX<n>	tengely-specifikus adatok (<n> a géptengely-számot adja meg)
TO	szerszámadatok
COMPLETE	minden adat

Inicializáló programot külső PC-n létrehozni

Az adat-tartomány jelölők és adattípus jelölők segítségével meg lehet határozni a tartományokat, amelyeket az adatmentésnél egységként kezelünk.

_N_AX5_TEA_INI	gépadatok a tengely 5-re
_N_CH2_UFR_INI	framek a csatorna 2-re
_N_COMPLETE_TEA_INI	összes gépadat

A vezérlés üzembehelyezése után a munkatárolóban van egy adatkészlet, ami a vezérlés rendeltetés-szerű üzemét biztosítja.

Eljárás többcsatornás vezérléseknél (CHANDATA)

`CHANDATA(<csatornaszám>)` több csatornára csak a N_INITIAL_INI fájlban megengedett. Ez az üzembehelyezési fájl, amellyel a vezérlés összes adata inicializálva lesz.

```
%_N_INITIAL_INI
CHANDATA(1)
                                        ; géptengely hozzárendelés csatorna 1:
$MC_AXCONF_MACHAX_USED[0]=1
$MC_AXCONF_MACHAX_USED[1]=2
$MC_AXCONF_MACHAX_USED[2]=3
CHANDATA(2)
                                        ; géptengely hozzárendelés csatorna 2:
$MC_AXCONF_MACHAX_USED[0]=4
$MC_AXCONF_MACHAX_USED[1]=5
CHANDATA(1)
                                        ; tengely gépadatok:
                                        ; pontos-állj ablak durva:
$MA_STOP_LIMIT_COARSE[AX1]=0.2          ; tengely 1
$MA_STOP_LIMIT_COARSE[AX2]=0.2          ; tengely 2
                                        ; pontos-állj ablak finom:
$MA_STOP_LIMIT_FINE[AX1]=0.01           ; tengely 1
$MA_STOP_LIMIT_FINE[AX1]=0.01           ; tengely 2
```

FIGYELEM

CHANDATA utasítás

IA munkadarabprogramban a `CHANDATA` utasítást csak arra a csatornára szabad használni, amelyben az NC program feldolgozása történik. Vagyis az utasítást lehet arra használni, hogy az NC programok védve legyenek egy nem tervezett csatorna által való végrehajtástól.

Hiba esetén a program feldolgozása meg lesz szakítva.

Megjegyzés

INI-fájlok a feladatlistában nem tartalmaznak `CHANDATA` utasításokat.

Inicializálási programok mentése (COMPLETE, INITIAL)

A munkatároló fájljait lehet egy külső PC-re menteni és onnan újra beolvasni.

- A fájlok mentése a `COMPLETE`-tel történik.
- Az `INITIAL`-lal az összes tartományról egy INI-fájl: az _N_INITIAL_INI lesz létrehozva.

Inicializálási programok beolvasása

FIGYELEM
Adatvesztés Az "INITIAL_INI" fájl beolvasásakor az összes adat, amelyeket a fájl nem ad meg, az alap-beállítás adataival lesz inicializálva. Ez alól csak a gépadatok jelentenek kivételt. Tehát a **beállítási adatok, szerszámadatok, NPE, GUD-értékek, ...** az alap-beállítási adatokkal (általában "nulla") lesznek feltöltve.

Az egyes gépadatok beolvasásához pl. a COMPLETE_TEA_INI fájl alkalmas. Ennél a fájnál a vezérlés csak gépadatokat vár. Ebben az esetben a többi adattartomány érintetlen marad.

Inicializálási programok töltése

Az INI-programokat munkadarab-programként is ki lehet választani és felhívni, ha csak egy csatorna adatait használják. Így lehetséges programvezérelt adatok inicializálása is.

Védelmi tartományok: 4

4.1 Védelmi tartományok definiálni (CPROTDEF, NPROTDEF)

A védelmi tartományok, amelyek a gépelemeket az ütközésektól kell óvják, a munkadarabprogramban blokkokban vannak definiálva. Ezek a következő elemeket tartalmazzák:

1. munkasík megadása
   A tényleges védelmi tartomány definíció előtt ki kell választani a munkasíkot, amelyre a védelmi tartomány kontúr-leírása vonatkozik.
2. definíció kezdete
   Az NC utasítástól függően vagy egy csatorna-specifikus vagy egy gép-specifikus védelmi tartomány lesz létrehozva.
3. védelmi tartomány kontúr-leírása
   Egy védelmi tartomány kontúrja elmozdulásokkal lesz leírva. Ezek nem lesznek végrehajtva és nincs kapcsolatuk a megelőző vagy következő geometria leírásokra. Kizárólag a védelmi tartományt definiáljuk.
4. definíció vége

Szintaxis

```
DEF INT <Var>
G17/G18/G19
CPROTDEF/NPROTDEF(<n>,<t>,<AppLim>,<AppPlus>,<AppMinus>)
G0/G1/... X/Y/Z...
...
EXECUTE(<Var>)
```

Jelentés

`DEF INT <Var>`:	INTEGER típusú helyi segédváltozó definiálása	
`<Var>`:	segédváltozó neve	
`G17/G18/G19`:	munkasík **Utalás:** A munkasíkot a definíció vége előtt nem szabad megváltoztatni. Az applikálta programozása a definíció kezdete és vége között nem megengedett.	
`CPROTDEF()`:	Előre definiált eljárás egy **csatorna**-specifikus védelmi tartomány definiálásához	
`NPROTDEF()`:	Előre definiált eljárás egy **gép**-specifikus védelmi tartomány definiálásához	
`<n>`:	definiált védelmi tartomány száma	
	adattípus:	INT

`<t>`:	védelmi tartomány típusa		
	adattípus:	BOOL	
	érték:	`TRUE`	**szerszámra vonatkozó** védelmi tartomány
		`FALSE`	**munkadarabra vonatkozó** védelmi tartomány
`<AppLim>`:	határolás módja a 3. dimenzióban		
	adattípus:	INT	
	érték:	`0`	nincs határolás
		`1`	határolás pozitív irányban
		`2`	határolás negatív irányban
		`3`	határolás pozitív és negatív irányban
`<AppPlus>`:	határolás értéke pozitív irányban a 3. dimenzióban		
	adattípus:	REAL	
`<AppMinus>`:	határolás értéke negatív irányban a 3. dimenzióban		
	adattípus:	REAL	
`G0/G1/... X/Y/Z... ...`:	Egy védelmi tartomány kontúrját a kiválasztott síkban maximum 11 elmozdulással adjuk meg. Az első elmozdulás a rámenetel a kontúrra. A kontúr-leírás utolsó pontja egybe kell essen a kontúr-leírás első pontjával. A védelmi tartomány a kontúrtól balra levő tartomány: • belső védelmi tartomány Egy belső védelmi tartomány kontúrját az óramutató járásával szemben kell leírni. • külső védelmi tartomány (csak a munkadarabra vonatkozó védelmi tartományokra megengedett) Egy külső védelmi tartomány kontúrját az óramutató járása szerint kell leírni. A következő kontúrelemek megengedett: • `G0`, `G1` egyenes kontúrelemek számára • `G2` óramutató járása szerinti körszeletek számára Csak a munkadarabra vonatkozó védelmi tartományokra megengedett! Nem megengedett szerszám-vonatkozású védelmi tartományoknál, amelyek csak konvexek lehetnek. • `G3` óramutató járásával ellentétes körszeletek számára **Utalás:** Egy védelmi tartományt nem lehet egy teljes körrel leírni. Egy teljes kört fel kell osztani két rész körre. **Utalás:** A `G2` → `G3` ill. `G3` → `G2` sorrend nem megengedett. A két kör mondat közé egy rövid `G1` mondatot kell beszúrni.		
`EXECUTE(<Var>)`:	Előre definiált eljárás a definíció végének jelöléséhez Az `EXECUTE`-tal a normál program-feldolgozáshoz kapcsolunk vissza.		

Példa

Lásd a példát "Védelmi tartományok aktiválása/deaktiválása (CPROT, NPROT) (Oldal 229)"-ban.

További információk

Gép-specifikus védelmi tartományok

Egy gép-specifikus védelmi tartomány ill. annak kontúrja a geometria-tengelyekkel lesz definiálva, azaz a az alap-koordinátarendszerre (AKR) vonatkoztatva. A helyes védelmi tartomány védelemhez az összes csatornában, amelyekben a gép-specifikus védelmi tartomány aktív, az összes érintett csatornában azonos kell legyen az alap-koordinátarendszer (AKR).

- koordináta kezdőpont helyzete a gép-nullapontra vonatkoztatva
- koordinátatengelyek tájolása

Kontúr-leírás vonatkoztatási pontja

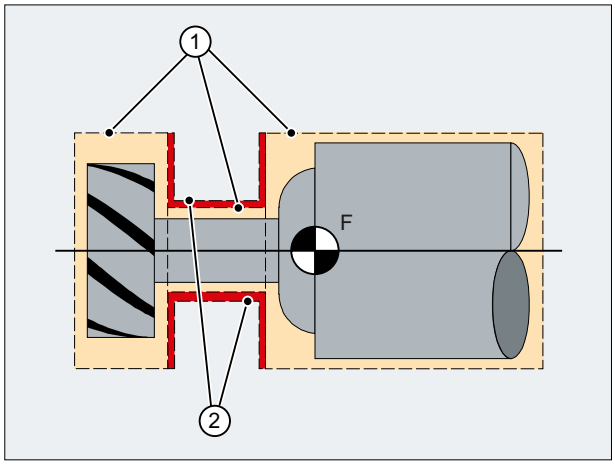
- Szerszámra vonatkozó védelmi tartományok
  A **szerszám** vonatkozású védelmi tartományokat abszolút, az **F szerszámtartó vonatkoztatási pontra** vonatkoztatva kell megadni.
- Munkadarabra vonatkozó védelmi tartományok
  A **munkadarab** vonatkozású védelmi tartományokat abszolút, az **alap-koordinátarendszer (AKR)** nullapontjára vonatkoztatva kell megadni.

Forgásszimmetrikus védelmi tartományok

Forgásszimmetrikus védelmi tartományoknál (pl.: orsótokmány) a teljes kontúrt le kell írni, nem csak kontúrt a forgásközépig.

Szerszámra vonatkozó védelmi tartományok

Szerszámra vonatkozó védelmi tartományoknak mindig konvexnek kell lenni. Ha konkáv védelmi tartományt szeretnénk, akkor ezt szét kell szedni több konvex védelmi tartományra.



① konvex védelmi tartományok

② konkáv védelmi tartományok (**nem megengedett!**)

F szerszámtartó vonatkoztatási pont

Peremfeltételek

Egy védelmi tartomány aktiválása közben a következő funkciók nem lehetnek aktívak ill. nem használhatók:

- szerszámsugár-korrekció (marósugár-korrekció, vágóélsugár-korrekció)
- transzformáció
- referenciapontra menet (`G74`)
- fix-pontra menet (`G75`)
- várakozási idő (`G4`)
- mondat előrefutás állj (`STOPRE`)
- programvég (`M17`, `M30`)
- M funkciók: `M0`, `M1`, `M2`

4.2 Védelmi tartományok aktiválása/deaktiválása (CPROT, NPROT)

Előzőleg a munkadarab-programban definiált védelmi tartományokat bármikor aktiválni ill. egy későbbi aktiváláshoz a PLC felhasználói programmal elő-aktiválni lehet. Aktív védőtartományokat bármikor újra deaktiválni lehet.

Az aktiválásnál ill. elő-aktiválásnál ezen kívül fennáll a lehetőség a védelmi tartomány vonatkoztatási pontjának eltolására.

Megjegyzés

Egy védelmi tartomány csak a csatornának, amelyben aktiválva lett, összes referencia-tengelyének referálása után lesz figyelembe véve.

Megjegyzés

Védelmi tartományok felügyelete

Ha egy szerszámra vonatkozóan egyetlen védelmi tartomány sem aktív, akkor a szerszámpálya a munkadarabra vonatkozó védelmi tartományokra meg lesz vizsgálva.

Ha egy munkadarabra vonatkozóan egyetlen védelmi tartomány sem aktív, akkor nincs védelmi tartomány felügyelet.

Szintaxis

```
CPROT(<n>,<Status>,<XMov>,<YMov>,<ZMov>)
NPROT(<n>,<Status>,<XMov>,<YMov>,<ZMov>)
```

Jelentés

CPROT:	Előre definiált eljárás egy **csatorna**-specifikus védelmi tartomány aktiválásához		
NPROT:	Előre definiált eljárás egy **gép**-specifikus védelmi tartomány aktiválásához		
<n>:	védelmi tartomány száma		
	adattípus:	INT	
<állapot>:	Ezzel a paraméterrel a csatorna-specifikus aktiválási állapot lesz beállítva		
	adattípus:	INT	
	érték:	0	védelmi tartomány deaktiválása
		1	védelmi tartomány elő-aktiválása
		2	védelmi tartomány aktiválása
		3	védelmi tartomány elő-aktiválása feltételes állj-jal

<XMov>,<YMov>,<ZMov>:	hozzáadódó eltolás értékek X/Y/Z irányban Az eltolás 1, 2 vagy 3 dimenzióban mehet végbe. Az eltolási értékek a következőkre vonatkoznak: • **munkadarabra** vonatkozó védelmi tartomány nullapontjára • **szerszámra** vonatkozó védelmi tartománynál az F szerszámtartó vonatkoztatási pontra	
	adattípus:	REAL

Példa

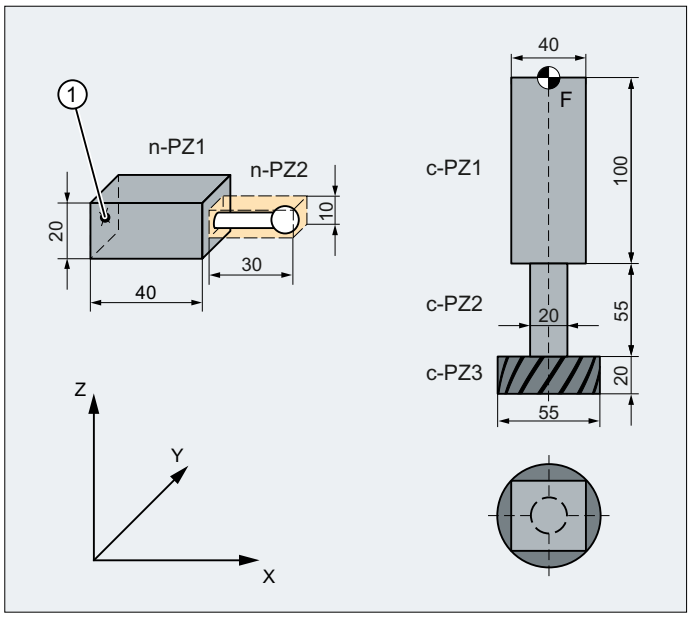
Egy marógépnél felügyelni kell a maró lehetséges ütközését a mérőtapintóval. A mérőtapintó helyzetét az aktiválásnál egy eltolással kell megadni.

Ehhez a következő védelmi tartományokat definiáljuk:

- Egy-egy gép-specifikus és munkadarab-specifikus védelmi tartomány a mérőtapintó-tartónak (n-PZ1) és magának a mérőtapintónak (n-PZ2).
- Egy-egy csatorna-specifikus és szerszámra vonatkozó védelmi tartomány a marótartónak (c-PZ1), a marótengelynek (c-PZ2) és magának a marónak (c-PZ3).

Minden védelmi tartomány tájolása Z-irányba mutat.

A mérőtapintó vonatkoztatási pontjának helye az aktiválásnál X = –120, Y = 60 és Z = 80-nál legyen.



① mérőtapintó védelmi tartományának vonatkoztatási pontja

F szerszámtartó vonatkoztatási pont

Programkód	Kommentár
DEF INT PROTZONE	; egy segédváltozó definíciója
G17	; munkasík XY

```
Programkód                          Kommentár
; védelmi tartományok definiálása:
NPROTDEF(1,FALSE,3,10,-10)          ; n-PZ1 védőtartomány
G01 X0 Y-10
X40
Y10
X0
Y-10
EXECUTE(PROTZONE)
NPROTDEF(2,FALSE,3,5,-5)            ; n-PZ2 védőtartomány
G01 X40 Y-5
X70
Y5
X40
Y-5
EXECUTE(PROTZONE)
CPROTDEF(1,TRUE,3,0,-100)           ; c-PZ1 védőtartomány
G01 X-20 Y-20
X20
Y20
X-20
Y-20
EXECUTE(PROTZONE)
CPROTDEF(2,TRUE,3,-100,-150)        ; c-PZ2 védőtartomány
G01 X0 Y-10
G03 X0 Y10 J10
X0 Y-10 J-10
EXECUTE(PROTZONE)
CPROTDEF(3,TRUE,3,-150,-170)        ;c-PZ3 védőtartomány
G01 X0 Y-27.5
G03 X0 Y27.5 J27.5
X0 Y27.5 J-27.5
EXECUTE(PROTZONE)

; Védelmi tartományok aktiválása:
NPROT(1,2,-120,60,80)               ; n-PZ1 védőtartomány aktiválása eltolással
NPROT(2,2,-120,60,80)               ; n-PZ2 védőtartomány aktiválása eltolással
CPROT(1,2,0,0,0)                    ; c-PZ1 védőtartomány aktiválása
CPROT(2,2,0,0,0)                    ; c-PZ2 védőtartomány aktiválása
CPROT(3,2,0,0,0)                    ; c-PZ3 védőtartomány aktiválása
```

További információk

Aktiválási állapot a vezérlés felfutása után

Egy védelmi tartomány már a vezérlés felfutása és a tengelyek referálása után aktív lehet. Ez akkor van, ha a védelmi tartományra a következő rendszerváltozók TRUE-ra vannak állítva:

- $SN_PA_ACTIV_IMMED[<n>] (gép-specifikus védelmi tartományokra)
- $SC_PA_ACTIV_IMMED[<n>] (csatorna-specifikus védelmi tartományokra)
  Az "<n>" index megfelel a védelmi tartomány számának: 0 = 1. védelmi tartomány

A védelmi tartomány az állapot = 2-vel és eltolás nélkül lesz aktiválva.

Egy védelmi tartomány többszörös aktiválása

Egy gép-specifikus védelmi tartomány egyszerre több csatornában is hatásos lehet (pl.: csúcstámasz védelmi tartománya két szemközti szánnál). A védelmi tartományok felügyelete csak akkor megy végbe, ha minden geometriai tengely referálva van.

A védelmi tartomány egy csatornában nem aktiválható egyszerre különböző eltolásokkal.

Védelmi tartomány felügyelet aktív szerszámsugár-korrekciónál

Egy működőképes védelmi tartomány felügyelet aktív szerszámsugár-korrekciónál csak akkor lehetséges, ha a szerszámsugár-korrekció síkja azonos a védelmi tartomány definíciók síkjával.

4.3 Védelmi-tartomány sértés, munkatér-határolás és szoftver-végállások vizsgálata (CALCPOSI)

Funkció

A CALCPOSI funkció megvizsgálja a munkadarab koordinátarendszerben (MKR), hogy a **geometria-tengelyek** kezdő pozícióiból kiindulva egy megadott utat meg lehet-e tenni az aktív határolások megsértése nélkül. Ha a mozgási utat a határolások miatt nem lehet teljesen megtenni, visszaadásra kerül egy pozitív, decimális kódolású állapot-érték és a maximálisan lehetséges mozgásút.

Definíció

```
INT CALCPOSI(VAR REAL[3] <Start>, VAR REAL[3] <Dist>, VAR REAL[5]
<Limit>, VAR REAL[3] <MaxDist>, BOOL <MeasSys>, INT <TestLim>)
```

Szintaxis

```
<Status> = CALCPOSI(VAR <Start>, VAR <Dist>, VAR <Limit>, VAR
<MaxDist>, <MeasSys>, <TestLim>)
```

Jelentés

`CALCPOSI(...):`	Előre definiált funkció a határolás sértések vizsgálatára a geometria-tengelyek vonatkozásában.	
	Előrefutás-állj:	nem
	egyedül a mondat-ban:	igen

`<állapot>`: (rész 1)	Funkció visszaadási érték. Negatív értékek hibaállapotot mutatnak.		
	adattípus:		INT
	értéktartomány:		-8 ≤ x ≤ 100000
	érték:	0	A mozgásutat teljesen meg lehet tenni.
		-1	A <határ>-ban legalább egy komponens negatív.
		-2	Hiba egy transzformáció kiszámításánál. Példa: A mozgási út egy szingularitáson keresztül vezet, így a tengely pozíciók nem definiáltak.
		-3	A megadott <táv> mozgásút és a maximálisan lehetséges <maxtáv> mozgásút lineárisan függőek. **Utalás** Nem léphet fel <teszthatár>, bit 4 == 1-gyel összefüggésben.
		-4	A <táv>-ban levő mozgásirány vetülete a határoló felületre nulla-vektor ill. a mozgásirány merőleges a megsértett határoló felületre. **Utalás** Nem léphet fel <teszthatár>, bit 5 == 1-gyel összefüggésben.
		-5	<teszthatár>-ban bit 4 == 1 ÉS bit 5 == 1
		-6	Legalább egy géptengely, amelyet a mozgási határok megvizsgálásához figyelembe kell venni, nem referált.
		-7	Ütközés elkerülés funkció: Kinematikai láncok vagy védelmi tartományok érvénytelen definíciója.
		-8	Ütközés elkerülés funkció: A funkciót a szűkös tároló miatt nem lehet végrehajtani.
`<állapot>`: (rész 2)	egyes helyiérték **Utalás** Ha egyszerre több határ lesz megsértve, az lesz jelentve, amelyik a megadott mozgásút legerősebb korlátozását okozza.		
	érték:	1	Szoftver-végállások határolják a mozgásutat
		2	Munkatér-határolás határolja a mozgásutat
		3	Védelmi tartományok határolják a mozgásutat
		4	Ütközés elkerülés funkció: Védelmi tartományok határolják a mozgásutat
	tízes helyiérték		
	érték:	1x	A kezdőérték megsérti a határt.
		2x	A megadott egyenes megsérti a határt. Ez az érték lesz visszaadva akkor is, ha a végpont maga nem sérti meg a határt, azonban a kezdő- és végpont közötti úton egy határérték-sértés lépne fel (pl. keresztülhaladás agy védelmi tartományon, kanyargós szoftver végállások MKR-ben nem-lineáris transzformációknál, pl. Transmit).

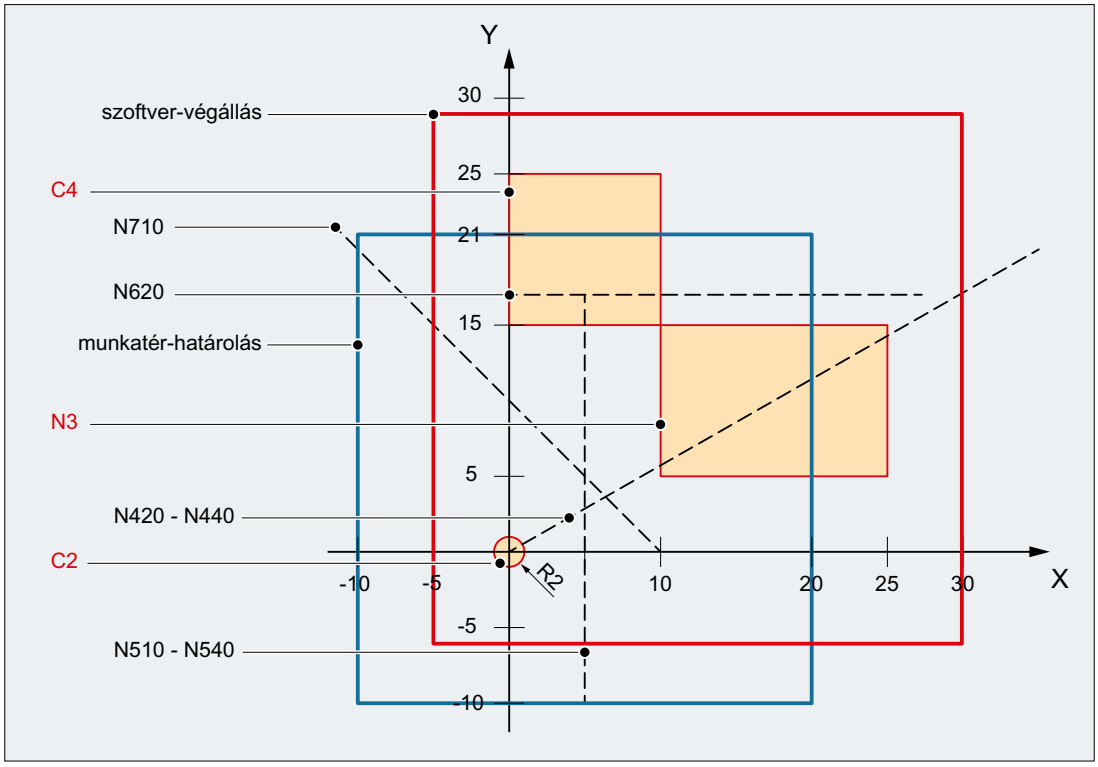
<állapot>: (rész 3)	százas helyiérték		
	érték:	1xx	ÉS egyes-hely == 1 vagy 2: A pozitív határérték van megsértve.
			ÉS egyes-hely == 3 [1]: Egy NC-specifikus védőtartomány van megsértve.
		2xx	ÉS egyes-hely == 1 vagy 2: A negatív határérték van megsértve.
			ÉS egyes-hely == 3 [1]: Egy csatorna-specifikus védelmi-tartomány van megsértve.
<állapot>: (rész 4)	ezres helyiérték		
	érték:	1xxx	ÉS egyes-hely == 1 vagy 2: Tényező, amivel a határt megsértő tengelyszám szorozva lesz. A tengelyek számozása 1-gyel kezdődik. Vonatkozás: • szoftver-végállás: Géptengelyek • Munkatér-határolás: geometria-tengelyek ÉS egyes-hely == 3 [1]: Tényező, amivel a megsértett védőtartomány száma szorozva lesz.
<állapot>: (rész 5)	egyszázezres-hely		
	érték:	0xxxxx	egyszázezres-hely == 0:<Dist> változatlan marad
		1xxxxx	A <Dist>-ban visszaadásra kerül egy irányvektor, amely definiálja a további mozgásirányt a határoló felületen. Csak a következő peremfeltételeknél léphet fel: • Szoftver-végállás vagy munkatér-határolás megsértve (nem a kezdőpontban) • Egy transzformáció **nem** aktív • <TestID>, bit 4 vagy bit 5 == 1
<Start>:	referencia egy vektorra a kezdő pozíciókkal: • <Start> [0]: 1. geometria-tengely • <Start> [1]: 2. geometria-tengely • <Start> [2]: 3. geometria-tengely		
	paraméter-típus:		bemenet
	adattípus:		VAR REAL[3]
	értéktartomány:		-max. REAL érték ≤ x[<n>] ≤ +max. REAL érték

`<Dist>`:	Referencia egy vektorra.	
	bemenet: növekményes mozgásutak • <Dist> [0]: 1. geometria-tengely • <Dist> [1]: 2. geometria-tengely • <Dist> [2]: 3. geometria-tengely	
	Kimenet (csak beállított egyszázezres-hellyel az <állapot>-ban):	
	<távolság> kimeneti értékként egy egység-vektort tartalmaz **v**, ami definiálja a további mozgásirányt az MKR-ben.	
	eset 1: **v** vektor képzése <TestID>, bit 4 == 1 esetén A <távolság> és <maxtávolság> bemeneti vektorok határozzák meg a mozgás síkját. Ez a sík lesz metszve a megsértett határoló felülettel. A két sík metszésének egyenese definiálja a **v** vektor irányát. Ennél a tájolás (előjel) úgy lesz megválasztva, hagy a <MaxDist> és **v** bemeneti vektorok közötti szög ne legyen nagyobb 90 foknál.	
	eset 2: **v** vektor képzése <TestID>, bit 5 == 1 esetén A **v** vektor az egység-vektor a <távolság>-ban levő mozgásvektor vetületének irányában a határoló felületen Ha a mozgásvektor vetülete a határoló felületre egy nulla-vektor, egy hiba lesz visszaadva.	
	paraméter-típus:	Be/kimenet
	adattípus:	VAR REAL[3]
	értéktartomány:	-max. REAL érték ≤ x[<n>] ≤ +max. REAL érték
`<határ>`:	referencia a hossz 5 egy mezőjére • <határ> [0 - 2]: geometria tengelyek hozzárendelése a határolásokhoz: – <határ> [0]: 1. geometria-tengely – <határ> [1]: 2. geometria-tengely – <határ> [2]: 3. geometria-tengely A minimális távolságok be lesznek tartva: – Munkatér-határolás: Nincsenek korlátozások – szoftver-végállás: Ha nincs aktív transzformációm vagy egy olyan transzformáció aktív, amelynél lehetséges a géptengelyek egyértelmű hozzárendelése a lineáris géptengelyekhez, pl. 5 tengelyes transzformációk. • <határ> [3]: Tartalmazza a minimális távolságot a lineáris tengelyekre, amelyekhez pl. egy nem lineáris transzformáció alapján nem lehet hozzárendelni geometria-tengelyeket. Ezen kívül ez az érték határértékként lesz alkalmazva hagyományos védelmi tartományok és az ütközés elkerülés védelmi tartományok felügyeleténél. • <határ> [4]: Tartalmazza a minimális távolságot a forgó géptengelyekre, amelyeket pl.egy nem-lineáris transzformáció miatt nem lehet hozzárendelni geometria-tengelyekhez. **Utalás** Ez az érték csak speciális transzformációk szoftver-végállásainak felügyeleténél hatásos.	
	paraméter-típus:	bemenet
	adattípus:	VAR REAL[5]
	értéktartomány:	-max. REAL érték ≤ x[n] ≤ +max. REAL érték

<MaxDist>:	Referencia egy vektorra növekményes mozgásúttal, amelynél egy egy tengely-határ megadott minimális távolságát az összes érintett géptengely nem lépi túl. • <Dist> [0]: 1. geometria-tengely • <Dist> [1]: 2. geometria-tengely • <Dist> [2]: 3. geometria-tengely Ha a mozgás nincs korlátozva, ennek a visszaadási paraméternek az értéke egyenlő a <Dist> tartalmával.		
	<TestID>, bit 4 == 1 esetén: <Dist> és <MaxDist> <MaxDist> és <Dist> bemeneti értékként olyan vektorokat kell tartalmazzanak, amelyek egy mozgássíkot feszítenek ki. A két vektor egymástól lineárisan független kell legyen. A <MaxDist> értéke tetszőleges. A mozgásirány kiszámításához lásd a <Dist>leírását.		
	paraméter-típus:		kimenet
	adattípus:		VAR REAL[3]
	értéktartomány:		-max. REAL érték ≤ x[<n>] ≤ +max. REAL érték
<MeasSys>:	Pozíció és hossz adatok mértékrendszere (hüvelyk / metrikus) (**opcionális**)		
	adattípus:		BOOL
	érték:	FALSE (alapérték)	A mértékrendszer megfelel a G-csoport 13 (G70, G71, G700, G710) aktuálisan aktív G utasításának. **Utalás** Aktív G70-nél és metrikus alaprendszernél vagy aktív G71-nél és hüvelyk alaprendszernél az $AA_IW és $AA_MW rendszerváltozók az alaprendszerben lesznek kiadva és a CALCPOSI alkalmazáshoz esetleg át kell számítani azokat.
		TRUE	mértékrendszer a beállított alaprendszer szerint: MD52806 $MN_ISO_SCALING_SYSTEM
<teszthatár>:	A felügyelendő határolások bit kódolású kiválasztása (**opcionális**)		
	adattípus:		INT
	alapérték:		Bit 0, 1, 2, 3, 6, 7 == 1 (207)
	bit	**decimális**	**Jelentés**
	0	1	Szoftver végálláskapcsoló
	1	2	Munkatér-határolás
	2	4	Aktivált hagyományos védelmi tartományok
	3	8	Előre aktivált hagyományos védelmi tartományok
	4	16	A szoftver-végálláskapcsolók ill. a munkatér-határolás megsértésénél a <táv>-ban a mozgásirányt az **eset 1** (lásd fent) szerint visszaadni.
	5	32	A szoftver-végálláskapcsolók ill. a munkatér-határolás megsértésénél a <táv>-ban a mozgásirányt az **eset 2** (lásd fent) szerint visszaadni.
	6	64	Ütközés-elkerülés aktivált védelmi tartományai
	7	128	Ütközés-elkerülés előre aktivált védelmi tartományai
	8	256	Ütközés-elkerülés aktivált és előre aktivált védelmi tartomány párok
[1] Ha több védelmi-tartomány lesz megsértve, az a védelmi tartomány lesz jelezve, amelyik a megadott mozgás legerősebb korlátozását okozza.			

Példa

Határolások



A példában a hatásos szoftver végállások és munkatér határolások az X-Y síkban és a következő három védelmi tartomány van ábrázolva:

- C2: szerszámra vonatkoztatott, csatorna-specifikus védelmi tartomány, aktív, köralakú, sugár = 2 mm
- C4: munkadarabra vonatkoztatott, csatorna-specifikus védelmi tartomány, négyzetes, oldalhossz = 10 mm
- N3: gép-specifikus védelmi tartomány, aktív, négyszögletes, oldalhossz = 10 mm x 15 mm

NC program

Az NC programban ezután a védelmi tartományok és munkatér határolások lesznek definiálva. Azután fel lesz hívva a `CALCPOSI()` funkció különböző paraméterezésekkel.

Programkód

```
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
```

Programkód

```
; szerszámra vonatkozó védelmi tartomány C2
N70 CPROTDEF(2, TRUE, 0)
N80 G17 G1 X-2 Y0
N90 G3 I2 X2
N100 I-2 X-2
N110 EXECUTE(_PA)
; munkadarabra vonatkozó védelmi tartomány C4
N120 CPROTDEF(4, FALSE, 0)
N130 G17 G1 X0 Y15
N140 X10
N150 Y25
N160 X0
N170 Y15
N180 EXECUTE(_PA)
; gép-specifikus védelmi tartomány N3
N190 NPROTDEF(3, FALSE, 0)
N200 G17 G1 X10 Y5
N210 X25
N220 Y15
N230 X10
N240 Y5
N250 EXECUTE(_PA)
; védelmi-tartományokat aktiválni ill. előre aktiválni
N260 CPROT(2, 2, 0, 0, 0)
N270 CPROT(4, 1, 0, 0, 0)
N280 NPROT(3, 2, 0, 0, 0)
; munkatér-határolásokat definiálni
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.
```

Programkód

```
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)
; farme-ket az N690-ból és N700-ból ismét törölni
N720 TRANS
N730 _START[0] =  0.
N740 _START[1] = 10.
N750 _START[2] =  0.
; _DIST és _MAXDIST vektorok definiálják a mozgási síkot
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
```

CALCPOSI() eredményei

N...	<Status>	<MaxDist>[0] ≙ X	<MaxDist>[1] ≙ Y	Megjegyzés
420	3123	8.040	4.594	N3 meg lesz sértve.
430	1122	20.000	11.429	Nincs felügyelet a védelmi tartományokra, munkatér-határolás megsértve.
440	1121	30.000	17.143	Már csak a szoftver végállások felügyelete aktív.
510	4213	0.000	0.000	Kezdőpont megsérti C4-et
520	0000	0.000	-27.000	Előre aktivált C4 nem lesz felügyelve. A megadott utat teljesen meg lehet tenni.
540	2222	0.000	-25.000	A _LIMIT[1]=2 miatt az elmozdulást a munkatér-határolás korlátozza.

N...	<Status>	<MaxDist>[0] ≙ X	<MaxDist>[1] ≙ Y	Megjegyzés
620	4223	-13 000	0.000	A távolság C4-hez C2 és _LIMIT[3] miatt összesen 4 mm. A C2 → N3 0,1 mm-es távolság nem okozza az elmozdulás korlátozását.
710	1221	0.000	21.213	Frame eltolással és elforgatással aktív. A megengedett elmozdulás a _DIST-ben az eltolt és elforgatott MKR-ben érvényes.
820	102121	18.000	18.000	Az Y tengely szoftver végálláskapcsolója meg lesz sértve. A <_TESTLIM> = 17-tel igényelve lesz egy tovább mozgási irány kiszámítása. Ez az irány a <_DIST>-ben van (0.707, 0.0, 0.707). Ez érvényes, mert a <_STATUS>-ban az egyszázezres helyiérték be van állítva.

További információk

Tengely állapot "Referálva"

Az összes `CALCPOSI()` által figyelt géptengely referálva kell legyen.

Kör vonatkozású útadatok

Az összes kör-vonatkozású útadat **mindig** sugár adatként lesz értelmezve. Ezt különösen a sík-tengelyeknél aktív átmérő-programozással (`DIAMON` / `DIAM90`) kell figyelembe venni.

Mozgásút csökkentés

Ha egy tengely megadott mozgásútja határolt, a `<MaxDist>` visszaadási értékben a többi tengely mozgásútjai is arányosan csökkentve lesznek. Az eredmény végpont ezáltal továbbra is a megadott pályán van.

Körtengelyek

A körtengelyek csak akkor lesznek felügyelve, ha nem modulo-körtengelyek.

Megengedett, hogy egy vagy több érintett tengelyre nincsenek szoftver-végállások ill. munkatér-határolások ill. védelmi-tartományok definiálva.

Szoftver-végállás és munkatér-határolás állapot

Szoftver-végállások és munkatér-határolások csak akkor lesznek figyelembe véve, ha azok a `CALCPOSI()` végrehajtásánál aktívak. Az állapotot pl. befolyásolni lehet a következővel:

- gépadat: MD21020 $MC_WORKAREA_WITH_TOOL_RADIUS
- beállítási adat: $AC_WORKAREA_CS_...
- NC/PLC interfész jelek: DB31, ... DBX12.2 / 3
- utasítások: `WALIMON` / `WALIMOF`

Szoftver-végállások és transzformációk

A `CALCPOSI()`-nál a különféle kinematikai transzformációknál (pl. `TRANSMIT`) a mozgási tartomány bizonyos helyein a több-értelműség miatt a géptengelyek pozícióit (GKR) nem lehet mindig egyértelműen meghatározni a geometria-tengelyek pozícióiból (MKR). A normális

mozgási üzemben az egyértelműség általában az előtörténetből adódik, és abból, hogy egy folytonos mozgás az MKR-ben meg kell feleljen a géptengelyek folytonos mozgásának a GKR-ben. A szoftver-határok felügyeleténél az ilyen esetekben a `CALCPOSI()` végrehajtásának időpontjában ismert gép-pozíció lesz felhasználva a több-értelműség feloldásához.

Megjegyzés

előrefutás-állj

A `CALCPOSI()` alkalmazásánál transzformációkkal összefüggésben kizárólag a felhasználó felelőssége, hogy a géptengely pozíciók szinkronizálásához előrefutásban a `CALCPOSI()` előtt egy előrefutás-álljt (`STOPRE`) programozzon.

Védelmi tartomány távolság és hagyományos védelmi tartományok

A hagyományos védelmi tartományoknál **nem** garantált, hogy egy mozgásnál a megadott mozgási úton a `<határ>[3]` paraméterben megadott biztonsági távolság az összes védelmi tartománnyal szemben be lesz tartva. Csak az garantált, hogy a `<táv>`-ban visszaadott végpont biztonsági távolsággal történő meghosszabbításánál nem lesz megsértve védelmi tartomány. Az egyenes azonban tetszőlegesen közel vezethet el egy védelmi tartomány mellet.

Védelmi tartomány távolság és ütközés-elkerülés védelmi tartományai

Az ütközés elkerülés védelmi tartományoknál garantált, hogy egy mozgásnál a megadott mozgási úton a `<határ>[3]` paraméterben megadott biztonsági távolság az összes védelmi tartománnyal szemben be lesz tartva.

A `<határ>[3]` paraméterben megadott biztonsági távolság csak akkor lesz hatásos, ha érvényes:

`<határ>[3]` **>** (MD10619 $MN_COLLISION_TOLERANCE)

Ha a `<teszthatár>` paraméterben a bit 4 be van állítva (további mozgásirány kiszámítása), akkor a `<táv>`-ban levő irányvektor csak akkor érvényes, ha az egyszázezres helyiérték a funkció visszaadási értékében (`<állapot>`) be van állítva. Ha egy ilyen irányt nem lehet megállapítani, mert vagy meg lettek sértve védelmi tartományok vagy egy transzformáció aktív, a beadási érték a `<táv>`-ban változatlan marad. További hibajelzés nincsen.

Speciális út-utasítások 5

5.1 Kódolt pozíciókra menni (CAC, CIC, CDC, CACP, CACN)

A következő utasításokkal lehetséges egyenes- és körtengelyeket pozíciószámokkal a gépadat táblázatokban megadott fix tengelypozíciókra menni. Ezt programozási módot "Kódolt pozícióra menés"-nek nevezik.

Szintaxis

```
CAC(<n>)
CIC(<n>)
CACP(<n>)
CACN(<n>)
```

Jelentés

`CAC(<n>)`:	n pozíciószám kódolt pozíciójára menni
`CIC(<n>)`:	kódolt pozícióra, kiindulva az aktuális pozíciószámból, n pozícióhellyel előre (+n) vagy vissza (-n) menni
`CDC(<n>)`:	n pozíciószám kódolt pozíciójára a legrövidebb úton menni (csak körtengelyekre)
`CACP(<n>)`:	n pozíciószám kódolt pozíciójára pozitív irányban menni (csak körtengelyekre)
`CACN(<n>)`:	n pozíciószám kódolt pozíciójára negatív irányban menni (csak körtengelyekre)
`<n>`:	pozíciószám a gépadat táblázaton belül értéktartomány: 0, 1, … (táblázat helyek max. száma - 1)

Példa: Egy pozícionáló tengely kódolt pozícióira menet

Programozási kód	**Kommentár**
`N10 FA[B]=300`	`; előtolás B pozicionáló tengelyekre`
`N20 POS[B]=CAC(10)`	`; pozíciószám 10 kódolt pozíciójára menni`
`N30 POS[B]=CIC(-4)`	`; "aktuális pozíciószám" - 4 kódolt pozíciójára menni`

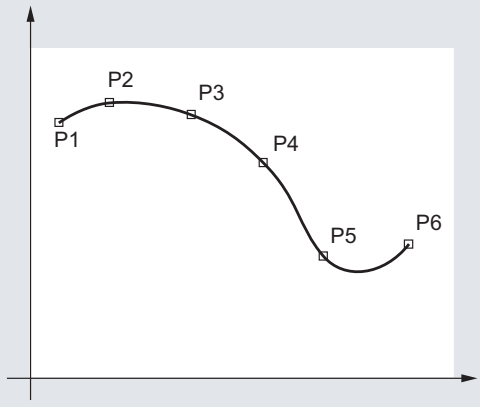
Irodalom

- Bővítő funkciók működési kézikönyv, osztó-tengelyek (T1)
- Szikonakciók működési kézikönyv

5.2 Spline interpoláció (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL)

A munkadarabok tetszőlegesen tördelt kontúrjait analitikailag nem lehet pontosan leírni. Az ilyen kontúrok ezért egy korlátozott számú támponttal, pl. a felület digitalizálásánál, közelítve lesznek. Egy digitalizált felület létrehozásához egy munkadarabon a támpontokat össze kell kötni egy kontúr-leíráshoz. Ezt teszi lehetővé a Spline interpoláció.

Egy Spline egy görbét definiál, amely 2. és 3. fokú polinomokból tevődik össze. Egy Spline támpontjainak a tulajdonságai **az alkalmazott Spline típustól függően** definiálhatók.



P1 ... P6: előre megadott koordináták

A SINUMERIK solution line-nál a következő Spline típusok állnak rendelkezésre:

- A-Spline
- B-Spline
- C-Spline

Szintaxis

Általános:

```
ASPLINE X... Y... Z... A... B... C...
BSPLINE X... Y... Z... A... B... C...
CSPLINE X... Y... Z... A... B... C...
```

B-Spline-nál továbbá programozható:

```
PW=<n>
SD=2
PL=<érték>
```

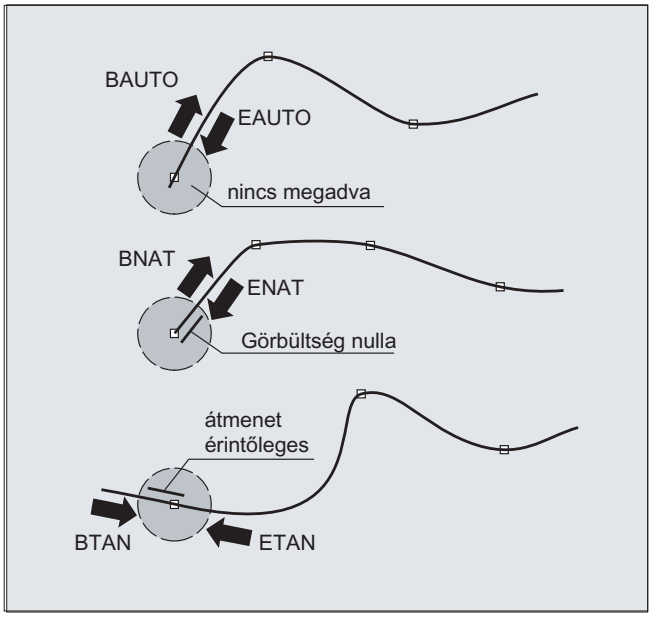
A- és C-Spline-oknál továbbá programozható:

`BAUTO` / `BNAT` / `BTAN`

`EAUTO` / `ENAT` / `ETAN`

Jelentés

Spline interpoláció típusok:			
ASPLINE:	utasítás az A-Spline interpoláció bekapcsolásához		
BSPLINE:	utasítás az B-Spline interpoláció bekapcsolásához		
CSPLINE:	utasítás a C-Spline interpoláció bekapcsolásához		
	Az ASPLINE, BSPLINE és CSPLINE utasítások modálisan hatásosak és az út-utasítások csoportjába tartoznak		
Támpontok ill. vezérlőpontok:			
X... Y... Z... A... B... C...	pozíciók derékszögű koordinátákban		
Pont-súlyozás (csak B-Spline):			
PW:	A PW utasítással minden támponthoz lehetséges egy "pont-súly" programozása.		
<n>:	"pont-súly"		
	értéktartomány:	0 ≤ n ≤ 3	
	lépéshossz:	0.0001	
	hatás:	n > 1	a görbét a vezérlő pont erősebben vonzza
		n < 1	a görbét a vezérlő pont kevésbé erősen vonzza
Spline-fok (csak B-Spline):			
SD:	Szabványosan 3. fokú poligon van alkalmazva. Az SD=2 programozásával lehetegy 2. fokú poligont is alkalmazni.		
Csomó-távolság (csak B-Spline):			
PL:	A csomó távolságok belül lesznek megfelelően kiszámítva. A vezérlés képes előre megadott csomó távolságokat is feldolgozni, amelyek az úgynevezett PL paraméter-intervallum hosszban vannak megadva.		
<érték>:	paraméter-intervallum hossz		
	értéktartomány:	mint az útméret	
Átmenet-viselkedés a Spline görbe elején (csak A- vagy C-Spline):			
BAUTO:	Nincs előbeállítás az átmeneti viselkedéshez. A kezdet az első pont helyzetéből adódik.		
BNAT:	görbültség nulla		
BTAN:	érintőleges átmenet az előző mondathoz (törlési állapot)		
Átmenet-viselkedés a Spline görbe végén (csak A- vagy C-Spline):			
EAUTO:	Nincs előbeállítás az átmeneti viselkedéshez. A vég az utolsó pont helyzetéből adódik.		
ENAT:	görbültség nulla		

ETAN:	érintőleges átmenet az előző mondathoz (törlési állapot)
	

Megjegyzés

A programozható átmeneti viselkedésnek nincsenek hatása a B- Spline-ra. A B-Spline a kezdő- és végpontjában mindig érintőleges a vezérlő-poligonhoz.

Peremfeltételek

- A szerszámsugár-korrekció alkalmazható.
- Az ütközés felügyelet a síkba vetítve történik.

Példák

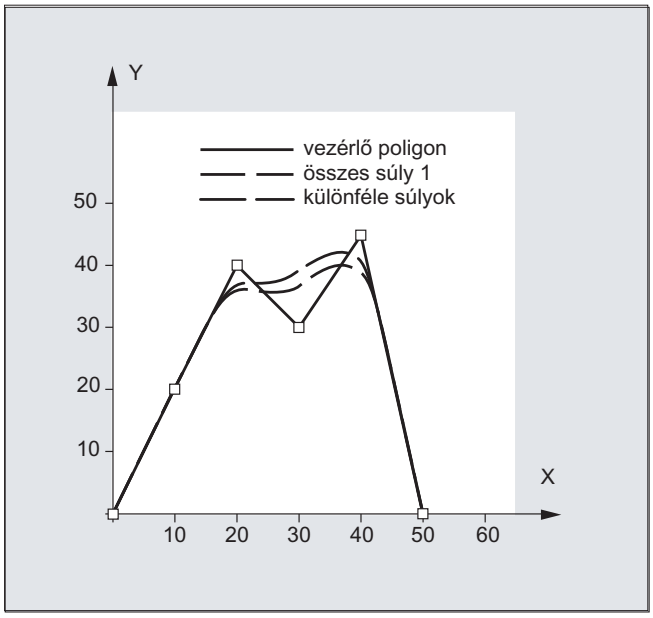
Példa 1: B-Spline

```
Program-kód 1 (minden súly 1)
N10 G1 X0 Y0 F300 G64
N20 BSPLINE
N30 X10 Y20
N40 X20 Y40
N50 X30 Y30
N60 X40 Y45
N70 X50 Y0
```

Program-kód 2 (különböző súlyok)

```
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
```

Programkód 3 (vezérlő poligon)	Kommentár
N10 G1 X0 Y0 F300 G64	
N20	; nincs
N30 X10 Y20	
N40 X20 Y40	
N50 X30 Y30	
N60 X40 Y45	
N70 X50 Y0	



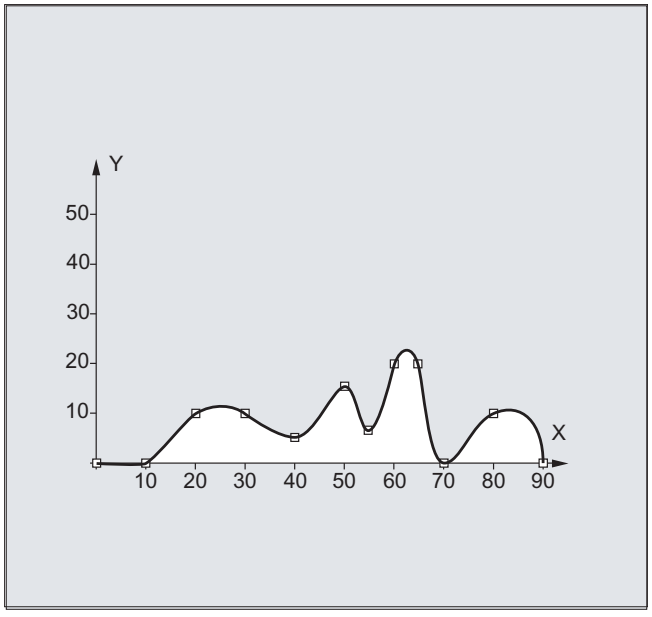
Példa 2: C-Spline, görbület a kezdetén és végén nulla

Programkód

```
N10 G1 X0 Y0 F300
N15 X10
N20 BNAT ENAT
N30 CSPLINE X20 Y10
```

Programkód

```
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
```



Példa 3: Spline interpoláció (A-Spline) és koordináta-transzformáció (ROT)

Főprogram:

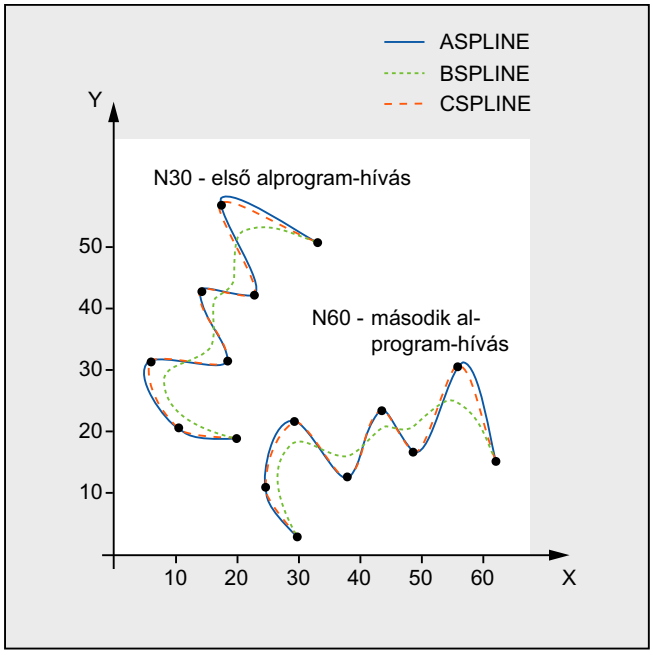
Programkód	Kommentár
N10 G00 X20 Y18 F300 G64	; Kezdőpontra rámenetel.
N20 ASPLINE	; A-Spline interpolációs típust aktiválni.
N30 KONTUR	; Az alprogram első felhívása.
N40 ROT Z-45	; Koordináta-transzformáció: MKR forgatása -45°-kal a Z tengely körül
N50 G00 X20 Y18	; Kontúr kezdőpontot felvenni.
N60 KONTUR	; Az alprogram második felhívása.
N70 M30	; programvég

"Kontur" alprogram (tartalmazza az összes támpont koordinátát):

Programkód

```
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
```

A következő ábra a programpéldából adódó Spline görbe (ASPLINE) mellett tartalmazza azokat a Spline görbéket is, amelyek egy B- vagy C-Spline interpoláció aktiválásánál (BSPLINE, CSPLINE) adódtak volna:



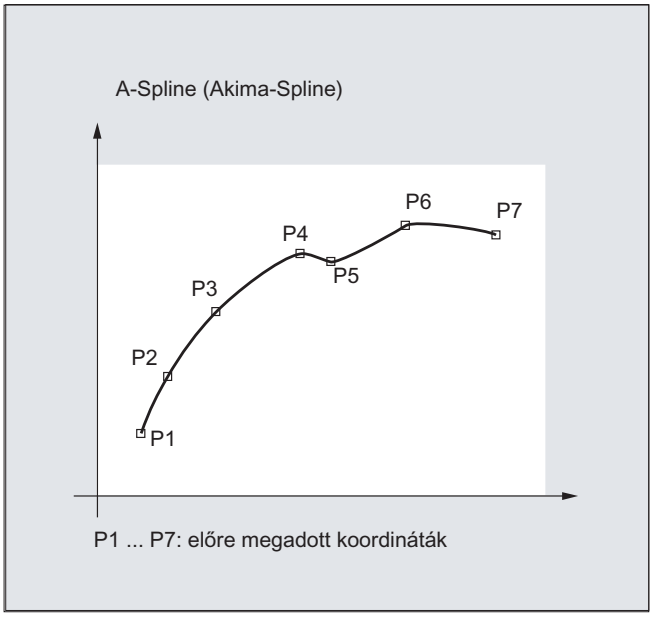
További információk

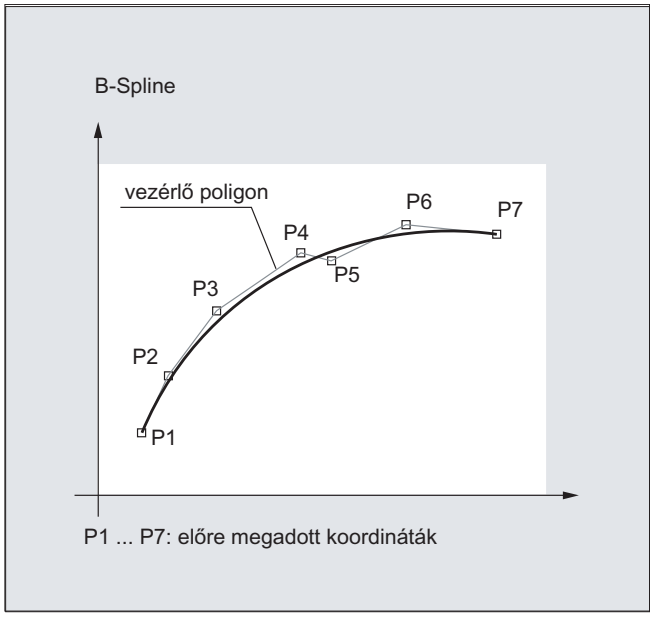
A Spline interpoláció előnyei

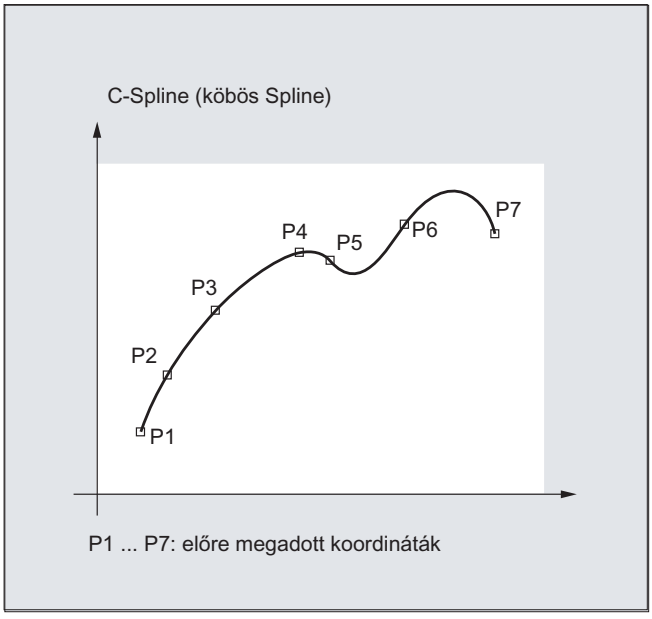
A Spline interpoláció alkalmazásával a következő előnyök érhetők el a `G01` egyenes mondatok alkalmazásával szemben:

- A kontúr leírásához szükséges munkadarabprogram mondatok száma csökken.
- Puhább, a mechanikát jobban kímélő kontúr lefutás a munkadarabprogram mondatok között.

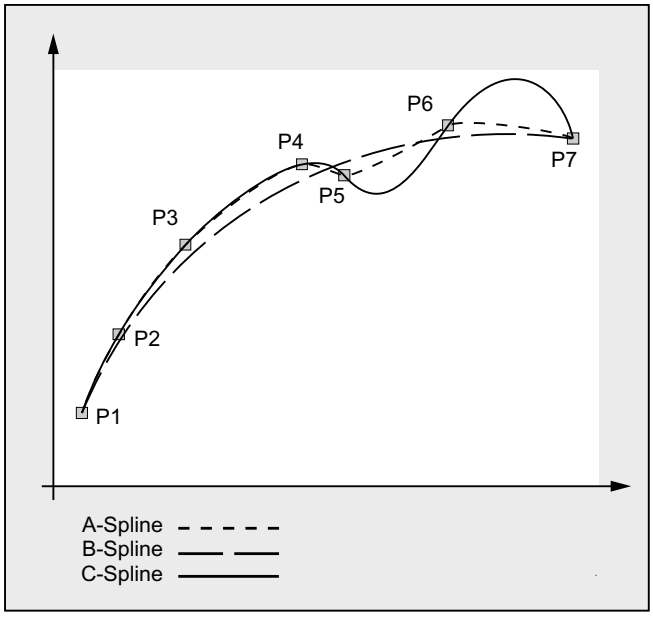
A különböző Spline típusok tulajdonságai és alkalmazásuk

Spline-típus	tulajdonságok és alkalmazás
A-Spline	 **Tulajdonságok:** • Pontosan a megadott támpontokon keresztül fut. • A görbe lefutása érintő- de nem görbület-állandó. • Alig eredményez akaratlan lengéseket. • A támpontok változtatásának befolyási tartománya helyi, vagyis egy támpont megváltoztatása csak max. 6 szomszédos támpontra hat ki. **Alkalmazás:** Az A-Spline mindenek előtt a nagy meredekség-változású (pl. lépcső-formájú) görbék interpolációjára alkalmas.

Spline-típus	tulajdonságok és alkalmazás
B-Spline	 **Tulajdonságok:** • Lefutás nem közvetlenül a megadott támpontokon, hanem csak a közelükben. A görbét a támpontok vonzzák. A támpontok súlyozásával egy tényezővel a görbe lefutását kiegészítőleg befolyásolni lehet. • A görbe lefutása érintő- és görbület-állandó. • Nem eredményez akaratlan lengéseket. • A támpontok változtatásának befolyási tartománya helyi, vagyis egy támpont megváltoztatása csak max. 6 szomszédos támpontra hat ki. **Alkalmazás:** A B-Spline elsősorban CAD-rendszerek csatolására szolgál.

Spline-típus	tulajdonságok és alkalmazás
C-Spline	 **Tulajdonságok:** • Pontosan a megadott támpontokon keresztül fut. • A görbe lefutása érintő- és görbület-állandó. • Gyakran eredményez akaratlan lengéseket, különösen a nagy meredekség-változású helyeken. • A támpontok változtatásának befolyási tartománya globális, vagyis egy támpont megváltoztatása a teljes görbe lefutására kihat. **Alkalmazás:** A C-Spline-t akkor lehet jól használni, ha a támpontok egy analitikailag ismert görbén (kör, parabola, hiperbola) találhatók.

A három Spline-típus összevetése azonos támpontok esetén



Spline-mondatok minimális száma

A `ASPLINE`, `BSPLINE` és `CSPLINE` G utasítások mondat végpontokat kötnek össze Spline-okkal. Ehhez előrefutással egy sor mondatot (végpontokat) kell egyidejűleg kiszámítani. A puffer nagysága a számításhoz alap-beállításban 10 mondat. Nem minden mondat információ egy Spline-végpont. A vezérlésnek azonban a 10 mondatból egy bizonyos számú Spline-végpont mondatra van szüksége:

Spline-típus	Spline-mondatok minimális száma
A-Spline:	Minden 10 mondatból legalább **4** Spline-mondat kell legyen. Kommentár mondatok és paraméter számítások itt nem számítanak.
B-Spline:	Minden 10 mondatból legalább **6** Spline-mondat kell legyen. Kommentár mondatok és paraméter számítások itt nem számítanak.
C-Spline:	A Spline-mondatok szükséges minimális száma a következő összegből adódik: MD20160 $MC_CUBIC_SPLINE_BLOCKS + 1 értéke Az MD20160-ban a pontok száma van megadva, amelyekből a Spline-szakasz lesz kiszámítva. Alap beállítás 8. Minden 10 mondatból alap-esetben legalább **9** Spline-mondat kell legyen.

Megjegyzés

Az elfogadható értékek átírásánál, ill. ha egy a Spline-ban érintett tengely pozícionáló tengelyként van programozva, egy vészjelzés kerül kiadásra.

Rövid Spline-mondatok összefogása

A Spline interpolációnál keletkezhetnek rövid Spline-mondatok, amelyek a pályasebesség szükségtelen csökkenéséhez vezetnek. A "Rövid Spline-mondatok összefogása" funkcióval ezeket a mondatokat össze lehet úgy fogni, hogy az eredő mondathossz kielégítően nagy és nem okoz pályasebesség csökkenést.

A funkció a következő csatorna-specifikus gépadattal aktiválható:

MD20488 $MC_SPLINE_MODE ((beállítás Spline interpolációhoz)

Irodalom:
Funktionshandbuch Grundfunktionen; Bahnsteuerbetrieb, Genauhalt, LookAhead (B1), Kapitel: Rövid Spline-mondatok összefogása

5.3 Spline-kötelék (SPLINEPATH)

A Spline-kötelékben interpoláló tengelyeket a `SPLINEPATH` utasítással kell kiválasztani. Maximum nyolc pályatengely lehetséges a Spline-interpolációnál.

Megjegyzés

Ha a `SPLINEPATH` nincs programozva, a csatorna első három tengelye mozog Spline-kötelékként.

Szintaxis

A Spline-kötelék megadása egy külön mondatban történik.

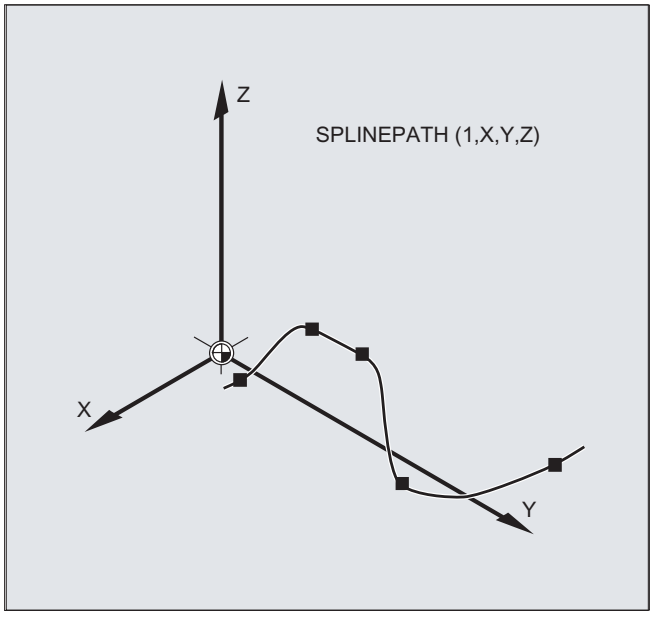
```
SPLINEPATH(n,X,Y,Z,…)
```

Jelentés

`SPLINEPATH`:	utasítás a Spline-kötelék megadására
`n`:	=1 (fix érték)
`X,Y,Z,…` :	a Spline-kötelékben interpoláló pályatengelyek megadása

Példa: Spline-kötelék három pályatengellyel

Programkód	**Kommentár**
`N10 G1 X10 Y20 Z30 A40 B50 F350`	
`N11 SPLINEPATH(1,X,Y,Z)`	`; Spline-kötelék`
`N13 CSPLINE BAUTO EAUTO X20 Y30 Z40 A50 B60`	`; C-Spline`
`N14 X30 Y40 Z50 A60 B70`	`; támpontok`
`…`	
`N100 G1 X… Y…`	`; Spline-interpoláció kikap-csolása`


Z
SPLINEPATH (1,X,Y,Z)
X
Y

5.4 NC-mondat kompresszort be-/kikapcsolni (COMPON / COMPCURV, COMPCAD, COMPSURF, COMPOF)

A lineáris mondatok kompressziójának funkciói (és a paraméterezéstől függően a kör és/vagy gyorsmeneti mondatok is) a 30-as G-csoport G utasításaival lesznek be-/kikapcsolva. Az utasítások modálisan hatnak.

Szintaxis

```
COMPON / COMPCURV / COMPCAD / COMPSURF
...
COMPOF
```

Jelentés

COMPON:	COMPON kompresszor funkció bekapcsolása
COMPCURV:	COMPCURV kompresszor funkció bekapcsolása
COMPCAD:	COMPCAD kompresszor funkció bekapcsolása
COMPSURF:	COMPSURF kompresszor funkció bekapcsolása
COMPOF:	aktuálisan aktív kompresszor funkció kikapcsolása

Megjegyzés

A felületi minőség további javitására lehet használni a G642 átsimítás funkciót és a SOFT rándítás-határolást. Ezeket az utasításokat a program elejére kell írni.

Példa: COMPCAD

```
Programkód                    Kommentár
N10 G00 X30 Y6 Z40
N20 G1 F10000 G642            ; bekapcsolás: G642 átsimítás funkció
N30 SOFT                      ; bekapcsolás: SOFT rántás-határolás
N40 COMPCAD                   ; bekapcsolás: COMPCAD kompresszor funkció
N50 STOPFIFO
N24050 Z32.499                ; 1. elmozdulási mondat
N24051 X41.365 Z32.500        ; 2. elmozdulási mondat
...
N99999 X... Z...              ; utolsó elmozdulási mondat
COMPOF                        ; kompresszor funkció ki
...
```

5.5 Polinom-interpoláció (POLY, POLYPATH, PO, PL)

A polinom-interpoláció (`POLY`) tulajdonképpen nem egy Spline-interpolációs mód. Elsősorban a külsőleg létrehozott Spline-görbék programozásához szolgál interfészként. Itt lehetséges Spline-szakaszok közvetlen programozása.

Ez az interpolációs mód tehermentesíti az NC-t a polinom-együtthatók kiszámításától. Akkor használható optimálisa, ha az együtthatók közvetlenül egy CAD-rendszertől vagy egy posztprocesszortól jönnek.

Szintaxis

3. fokú polinom

```
POLY PO[X]=(xe,a2,a3) PO[Y]=(ye,b2,b3) PO[Z]=(ze,c2,c3) PL=n
```

5.fokú polinom és új polinom-szintaxis:

```
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")
```

Megjegyzés

Az egy mondatban programozott polinom-együtthatók és tengelyek összege nem haladhatja meg a mondatonként megengedett tengelyszámot.

Jelentés

`POLY:`	polinom-interpoláció bekapcsolása egy POLY-t tartalmazó mondattal
`POLYPATH:`	polinom-interpoláció választható az AXIS vagy VECT tengelycsoportok mindegyikére
`PO[tengely-jelölő/változó]:`	végpontok és polinom-együtthatók
`X, Y, Z:`	tengely-jelölők
`xe, ye, ze :`	tengely végpozíciók megadása; értéktartomány mint az útméreteknél
`a2, a3, a4, a5:`	Az a_2, a_3, a_4, és a_5 együtthatók az értékeikkel lesznek beírva; értéktartomány mint az útméretnél Az utolsó együttható elmaradhat, ha az értéke nulla.
`PL:`	A paraméter-tartomány hossza, amelyen a polinom definiálva van (az f(p) függvény definíciós tartománya). A tartomány mindig 0-nál kezdődik, a p a 0 ... PL értéket veheti fel. Elvi értéktartomány a PL-re: 0,0001 … 99 999,9999 **Utalás:** A PL érték arra a mondatra érvényes, amelyben áll. Ha nincs PL programozva, PL=1 hatásos.

A polinom-interpoláció be-/kikapcsolása

A polinom-interpoláció a munkadarabprogramban a POLY G-utasítással lesz bekapcsolva.

A POLY G-utasítás a G0, G1, G2, G3, ASPLINE, BSPLINE és CSPLINE G-utasításokkal együtt a 1. G-csoporthoz tartozik.

A csak névvel és végponttal programozott tengelyek (pl. X10), egyenesként lesznek megtéve. Ha egy NC-mondat összes tengelye így van programozva, a vezérlés úgy viselkedik, mint a G1-nél.

A polinom-interpoláció az 1. G-csoport egy másik utasításának (pl. G0, G1) programozásával közvetetten ki lesz kapcsolva.

Polinom-együttható

A PO-érték (PO[]=) ill. ...=PO(...) megadja egy tengelyre az összes polinom együtthatót. A polinom fokának megfelelően több érték van vesszőkkel elválasztva. Egy mondaton belül lehetségesek különböző fokú polinomok a különböző tengelyekre.

POLYPATH alprogram

A POLYPATH(...)-szal a polinom-interpolációt szelektíven lehet megadni a következő tengelycsoportokra:

Csak pályatengelyek és kiegészítő tengelyek:	POLYPATH("AXES")
Csak tájolási tengelyek: (tájolási transzformációval mozgásnál)	POLYPATH("VECT")

A nem engedélyezett tengelyek lineárisan mozognak.

Alap-beállításként a polinom-interpoláció mindkét tengelycsoportra engedélyezve van.

A POLYPATH( ) paraméter nélküli programozása deaktiválja a polinom-interpolációt az összes tengelyre.

Példa

Programkód	Kommentár
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	; polinom-interpoláció be
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	; vegyes adatok a tengelyekre
N27 PO[X]=(10,2.5) PO[Y]=(2.3)	; nincs PL programozva, PL=1 hatásos
N30 G1 X… Y… Z.	; polinom-interpoláció ki
…	

Példa: Új polinom-szintakszis

továbbra is érvényes polinom-szintaxis	Új polinom-szintakszis
`PO[tengely-jelölő]=(.. , ..)`	`tengely-jelölő=PO(.. , ..)`
`PO[PHI]=(.. , ..)`	`PHI=PO(.. , ..)`
`PO[PSI]=(.. , ..)`	`PSI=PO(.. , ..)`
`PO[THT]=(.. , ..)`	`THT=PO(.. , ..)`
`PO[]=(.. , ..)`	`PO(.. , ..)`
`PO[változó]=IC(.. , ..)`	`változó=PO IC(.. , ..)`

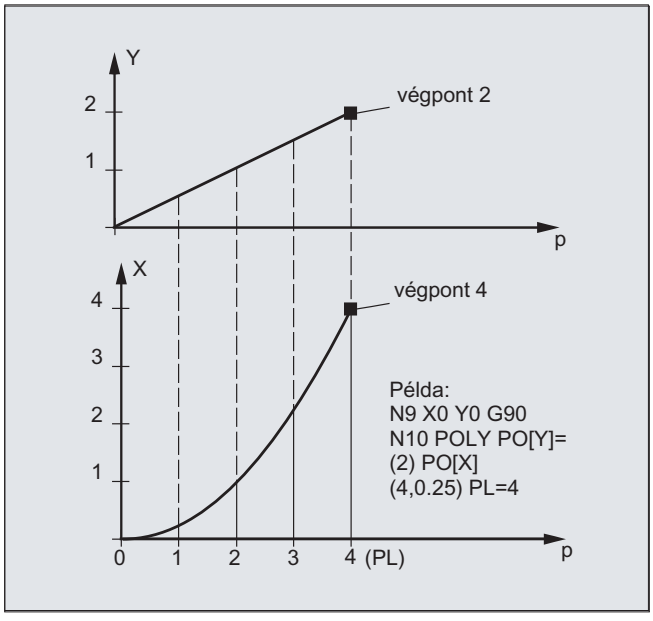
Példa: Görbe az X/Y síkban

Programozás

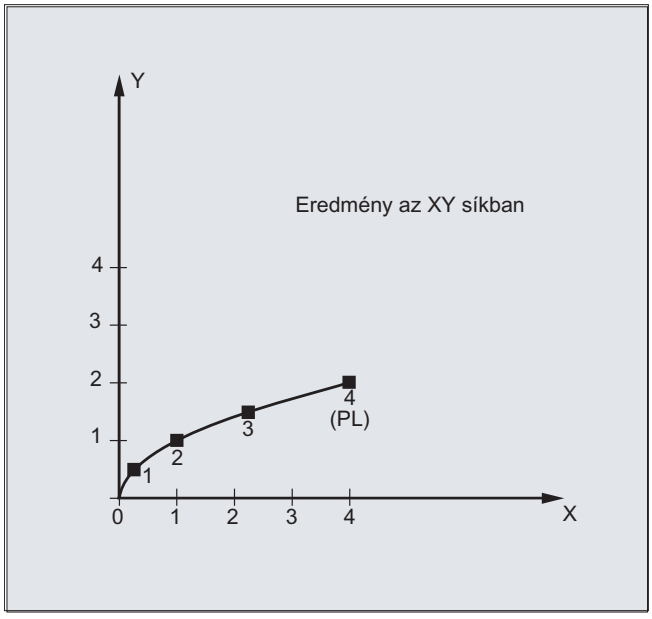
Programkód

```
N9 X0 Y0 G90 F100
N10 POLY PO[Y]=(2) PO[X]=(4,0.25) PL=4
```

X(p) és Y(p) görbék lefutása



Görbe lefutása az XY síkban



Leírás

A polinom-függvény általános formája a következő:

$f(p)= a_0 + a_1p + a_2p^2 + . . . + a_np^n$

ahol: a_i: állandó együtthatók (i = 0, 1, ..., n)
p: Paraméter

A vezérlésben maximum 5. fokú polinomokat lehet programozni:

$f(p)= a_0 + a_1p + a_2p^2 + a_3p^3 + a_4p^4 + a_5p^5$

Az együtthatók konkrét értékekkel feltöltésével különböző görbék, mint egyenesek, parabolák és hatvány-függvények hozhatóak létre.

Egyenes létrehozása: $a_2 = a_3 = a_4 = a_5 = 0$:

$f(p)= a_0 + a_1p$

Ahol a következő érvényes:

$a_{0:}$ tengely-pozíció az előző mondat végén
p = PL
$a_1 = (x_E - a_0 - a_2*p^2 - a_3*p^3) / p$

Lehet polinomokat programozni, **anélkül**, hogy a polinom-interpoláció a `POLY` G-utasítással aktiválva lenne. Ebben az esetben azonban nem a programozott polinom lesz interpolálva, hanem a minden tengely programozott végpontja lineárisan lesz felvéve (`G1`). Csak polinom-interpoláció aktiválása után a munkadarabprogramban (`POLY`) lesznek a programozott polinomok, mint olyanok, mozgatva.

Különlegesség: nevező-polinom

A geometria-tengelyekre lehet a `PO[]=(...)`-val a tengelynév megadása nélkül egy közös nevező-polinomot programozni, azaz a geometria-tengelyek mozgásai két polinom hányadosaként lesznek interpolálva.

Ezzel lehetséges pl. kúpmetszetek (kör, ellipszis, parabola, hiperbola) pontos ábrázolása.

Példa:

Programkód	Kommentár
`POLY G90 X10 Y0 F100`	`; A geometria-tengelyek egyenesen mozognak az X10 Y0 pozícióra.`
`PO[X]=(0,-10) PO[Y]=(10) PO[]=(2,1)`	`; A geometria-tengelyek negyedkörön mozognak az X0 Y10 pozícióra.`

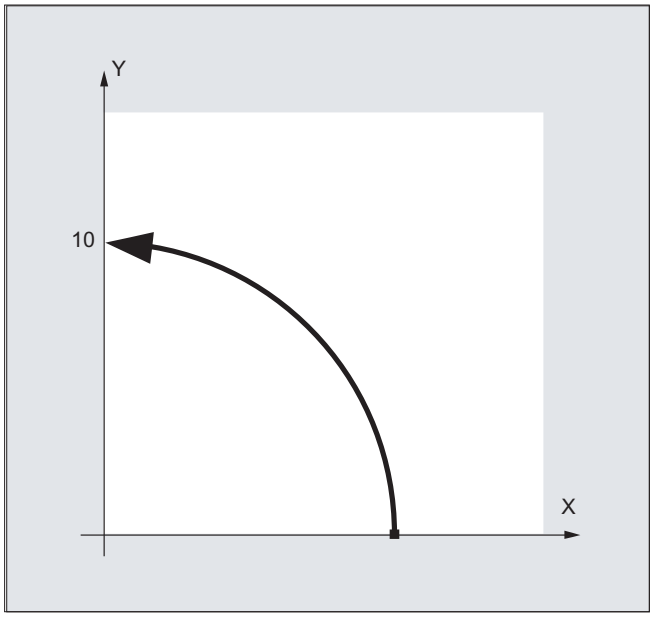
A nevező-polinom állandó együtthatója (a_0) mindig 1-nek van feltételezve. A programozott végpont független a `G90` / `G91`-től.

A programozott értékekből az X(p) és Y(p) kiszámítása:

$X(p) = (10 - 10 * p^2) / (1 + p^2)$
$Y(p) = 20 * p / (1 + p^2)$
$0 \leq p \leq 1$ esetén

A programozott kezdőpontok, végpontok, az a_2 együttható és PL=1 alapján a következő közbenső értékek adódnak:

számláló (X) = $10 + 0 * p - 10 * p^2$
számláló (Y) = $0 + 20 * p + 0 * p^2$
nevező = $1 + p^2$



Bekapcsolt polinom-interpolációnál egy nevező-polinom programozása nullahelyekkel a tartományon belül `[0,PL]` vészjelzéssel elutasításra kerül. A kiegészítő-tengelyekre a nevező- polinomnak nincs hatása.

Megjegyzés

A szerszámkorrekció a polinom-interpolációnál a `G41`, `G42`-vel bekapcsolható és akár az egyenes- vagy körinterpolációnál használható.

5.6 Beállítható pálya-vonatkoztatás: (SPATH, UPATH)

Polinom-interpolációnál (POLY, ASPLINE, BSPLINE, CSPLINE, COMPON, COMPCURV) az i pályatengely pozíciói a $p_i(U)$ polinommal lesznek megadva. Az U görbeparaméter ennél egy NC mondaton belül 0-tól 1-ig változik.

FGROUP-pal lehet a tengelyeket (FGROUP tengelyek) kiválasztani, amelyekre a pálya-előtolás F vonatkozik. Azonban az FGROUP tengelyek interpolációja állandó sebességgel az S pályaúton a polinom-interpoláció közben általában az U görbeparaméternek egy nem állandó változását jelenti. Az FGROUP-ban nem szereplő tengelyekre két lehetőségből lehet választani, hogy azok hogyan kövessék az FGROUP tengelyeket

- szinkron az S pályaúttal (SPATH)
- szinkron az U görbe-paraméterekkel (UPATH)

Szintaxis

```
SPATH
UPATH
```

Jelentés

SPATH:	Az FGROUP-ban nem szereplő tengelyek az S pályaútra vonatkoztatva fognak mozogni
UPATH:	Az FGROUP-ban nem szereplő tengelyek az U görbe-paraméterekre vonatkoztatva fognak mozogni

Megjegyzés

Az UPATH és az SPATH meghatározzák az F-szó polinom (FPOLY, FCUB, FLIN) összefüggését is a pályamozgással.

Peremfeltételek

Az SPATH ill. UPATH-nak nincs jelentősége:

- egyenes interpolációnál (G1)
- kör interpolációnál (G2, G3)
- meneteknél (G33, G34, G35, G33x, G63)
- Ha az összes pályatengelyt az FGROUP tartalmazza.

Példa

A következő példa mutatja a különbséget a kétféle mozgás-vezetés között.

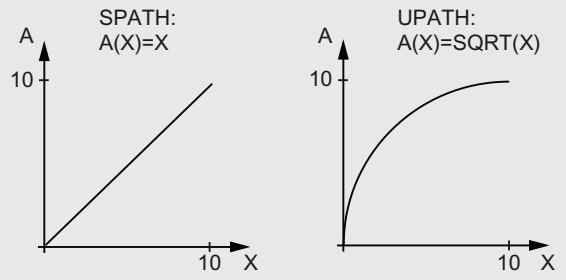
Programkód

```
N10 FGROUP(X,Y,Z)
N15 G1 X0 A0 F1000 SPATH                ; SPATH
N20 POLY PO[X]=(10,10) A10
```

Programkód

```
N10 FGROUP(X,Y,Z)
N15 G1 X0 A0 F1000 UPATH                ; UPATH
N20 POLY PO[X]=(10,10) A10
```

Mindkét programszakaszban az `N20` mondatban az FGROUP tengelyek S útja az U görbeparaméterek négyzetétől függ. Ekkor az X útja mentén különböző pozíciók adódnak az A szinkron-tengelyre attól függően, hogy `SPATH` vagy `UPATH` aktív.



különböző geometriai kapcsolatok tengelyek között SPATH és UPATH esetén

További információk

A vezérlés viselkedése Reset-nél és gép-/opciós-adatok

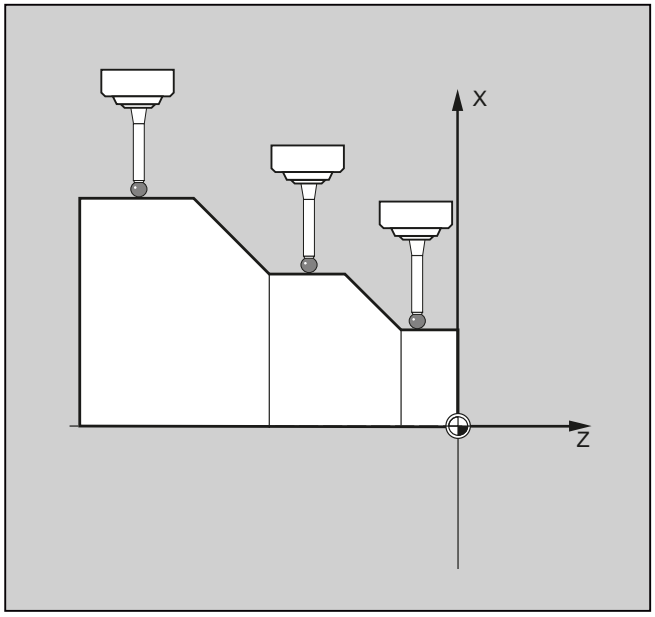
A Reset után az MD20150 $MC_GCODE_RESET_VALUES [44] által meghatározott G funkció hatásos (45. G-csoport).

Az átsimítás módjának alapbeállítás értékét az MD 20150 $MC_GCODE_RESET_VALUES [9] adja meg (10. G-csoport).

Az MD33100 $MA_COMPRESS_POS_TOL[<n>] tengely-specifikus gépadatnak kibővített jelentése van: tartalmazza a tűréseket a kompresszor funkciókhoz és az átsimításhoz G642-vel.

5.7 Mérés kapcsoló tapintóval (MEAS, MEAW)

A "Mérés kapcsoló tapintóval" funkcióval a munkadarabon valós pozíciók lesznek felvéve és a mérőtapintó kapcsoló élnél a mérés-mondatban programozott összes tengelyre a pozíciók mérésre majd a megfelelő tároló cellákban tárolásra kerülnek.



A funkció programozásához a következő két fix cím áll rendelkezésre:

- `MEAS`
  A `MEAS`-sal a maradékút a valós-és a célpozíció között törölve lesz
- `MEAW`
  A `MEAW` olyan mérési feladatokhoz lesz használva, amelyeknél mindenképpen a programozott pozíciót kell felvenni,

A `MEAS` és a `MEAW` mondatonként hatásosak és a mozgási utasításokkal együtt lesznek programozva. Az előtolások és az interpoláció módja (G0, G1, …), akárcsak a tengelyek száma, a mindenkori mérési feladathoz illesztendő.

Szintaxis

```
MEAS=<TE> G... X... Y... Z...
MEAW=<TE> G... X... Y... Z...
```

Jelentés

`MEAS`:	utasítás: mérés maradékút **törléssel**	
	hatásosság:	mondatonként
`MEAW`:	utasítás: mérés maradékút **törlés nélkül**	
	hatásosság:	mondatonként

<TE>:	trigger-esemény a mérés kiváltásához	
	típus:	INT
	értéktartomány:	-2, -1, 1, 2
	jelentés:	
	(+)1	1-es tapintó felfutó éle (1-es mérőbemeneten)
	-1	1-es tapintó lefutó éle (1-es mérőbemeneten)
	(+)2	2-es tapintó felfutó éle (2-es mérőbemeneten)
	-2	2-es tapintó lefutó éle (2-es mérőbemeneten)
	Utalás: Maximum 2 mérőtapintó létezik (kiépítéstől függően).	
G...:	Interpolációs mód, pl. G0, G1, G2 vagy G3	
X... Y... Z...:	végpontok derékszögű koordinátákban	

Példa

Programkód	Kommentár
N10 MEAS=1 G1 F1000 X100 Y730 Z40	; Mérési mondat az első mérőbemenet mérőtapintójával és egyenes-interpoláció. Előre-futás állj automatikusan létrehozva.
...	

További információk

Mérési feladat állapota

Ha a programban szükséges egy kiértékelés, hogy a mérő-tapintó kapcsolt-e, le lehet kérdezni az $AC_MEA[n] (<n> = mérő-tapintó száma) állapot-változót:

Érték	Jelentés
0	mérési feladat nincs végrehajtva
1	mérési feladat sikeresen végrehajtva (mérőtapintó kapcsolt)

Megjegyzés

Ha a mérőtapintó a programban kitér, a változó 1-be lesz állítva. Egy mérési mondat indításánál a változó automatikusan a tapintó kezdőállapotába lesz beállítva.

Mérési érték felvétele

A mondat összes elmozduló pálya- és pozícionáló-tengelyének a pozíciói fel lesznek véve (a tengelyek maximális száma a vezérlés konfigurációja szerint). A `MEAS`-nál a mozgás a mérő-tapintó kapcsolása után le lesz fékezve.

Megjegyzés

Ha egy mérési mondatban egy geometria-tengely van programozva, a mérési értékek az összes aktuális geometria-tengelyre tárolva lesznek.

Ha egy mérési mondatban egy transzformációban érintett tengely van programozva, az összes, a transzformációban érintett tengely mért értékei felvételre kerülnek

---

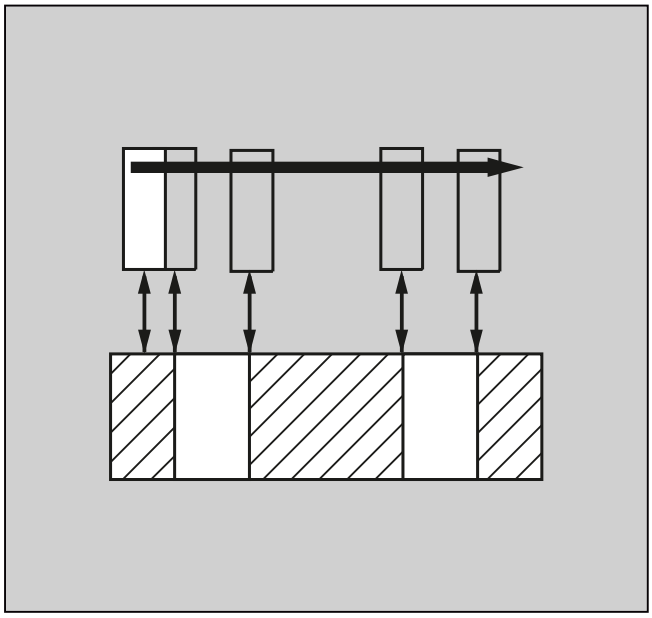
Mérés eredmények olvasása

A mérés eredmények a mérő-tapintóval megmért tengelyekre a következő rendszerváltozókban olvashatók:

- $AA_MM[<tengely>]
  mérés eredmények a gép-koordinátarendszerben
- $AA_MW[<tengely>]
  mérés eredmények a munkadarab-koordinátarendszerben

5.8 Tengely-specifikus mérés (MEASA, MEAWA, MEAC) (opció)

A tengely-specifikus mérésnél több mérő-tapintót és több mérőrendszert lehet használni.



A funkció programozásához a `MEASA`, `MEAWE` és `MEAC` kulcsszavak állnak rendelkezésre.

A `MEASA` ill. `MEAWA` utasításokkal a mindenkori programozott tengelyre mérésenként max. négy mérési érték lesz feldolgozva és a trigger-eseménynek megfelelően a rendszerváltozókban letéve.

Folyamatos mérési feladatokat a `MEAC` utasítással lehet végrehajtani. Ebben az esetben a mérési eredmények FIFO-változókban lesznek letéve.

Szintaxis

```
MEASA[<tengely>]=(<modus>,<TE1>,...,<TE4>)
MEAWA[<tengely>]=(<modus>,<TE1>,...,<TE4>)
MEAC[<tengely>]=(<modus>,<mérés tároló>,<TE1>,...,<TE4>)
```

Megjegyzés

A `MEASA` és a `MEAWA` mondatonként hatásosak és együtt programozhatók egy mondatban. Ha azonban a `MEASA`/`MEAWA` és a `MEAS`/`MEAW` egy mondatban van programozva, akkor hibajelzés keletkezik.

Jelentés

`MEASA`:	kulcsszó: tengely-specifikus mérés maradékút-törléssel	
	hatásosság:	mondatonként
`MEAWA`:	kulcsszó: tengely-specifikus mérés maradékút törlés nélkül	
	hatásosság:	mondatonként

MEAC:	kulcsszó: tengely-specifikus folyamatos mérés maradékút-törlés nélkül	
	hatásosság:	mondatonként
<tengely>:	méréshez használt csatorna-tengely neve	
<modus>:	kétjegyű szám az üzemmód (mérés modus és mérőrendszer) megadására	
	egyes jegy (mérés módus):	
	0	mérési feladatot megszakítani
	1	max. 4 különböző egyidőben aktiválható trigger-esemény
	2	max. 4 egymás után aktiválható trigger-esemény
	3	max. 4 egymásután aktiválható trigger-esemény, de nincs felügyelet a trigger-esemény 1 Start-nál (21700/21703 vészjelzések elnyomva). **Utalás:** Ez a modus MEAC-nál nem lehetséges.
	tízes kegy (mérőrendszer):	
	0 (vagy semmi)	aktív mérőrendszer
	1	mérőrendszer 1
	2	mérőrendszer 2
	3	mindkét mérőrendszer
<TE>:	trigger-esemény a mérés kiváltásához	
	típus:	INT
	értéktartomány:	-2, -1, 1, 2
	Jelentés:	
	(+)1	mérő-tapintó 1 felfutó éle
	-1	mérő-tapintó 1 lefutó éle
	(+)2	mérő-tapintó 2 felfutó éle
	-2	mérő-tapintó 2 lefutó éle
<mérés tároló>:	FIFO (kör-tároló) száma	

Példák

Példa 1: Tengely-specifikus mérés maradék út törléssel módus 1-ben (kiértékelés időrendi sorrendben)

a) 1 mérőrendszerrel

Programkód	Kommentár
...	
N100 MEASA[X]=(1,1,-1) G01 X100 F100	; Mérés modus 1-ben aktív mérőrendszerrel. Várás a mérőjelre felfutó/lefutó éllel a mérő-tapintó 1-től az X = 100 elmozdulási úton
N110 IF $AC_MEA[1]==FALSE GOTOF ENDE	; Mérés sikerességét ellenőrizni.
N120 R10=$AA_MM1[X]	; Az első programozott trigger-eseményhez (felfutó él) tartozó mérési értéket tárolni.

Programkód	Kommentár
N130 R11=$AA_MM2[X]	; A második programozott trigger-eseményhez (lelfutó él) tartozó mérési értéket tárolni.
N140 ENDE:	

b) 2 mérőrendszerrel

Programkód	Kommentár
...	
N200 MEASA[X]=(31,1,-1) G01 X100 F100	; Mérés modus 1-ben mindkét mérőrendszerrel. Várás a mérőjelre felfutó/lefutó éllel a mérő-tapintó 1-től az X = 100 elmozdulási úton
N210 IF $AC_MEA[1]==FALSE GOTOF ENDE	; Mérés sikerességét ellenőrizni.
N220 R10=$AA_MM1[X]	; Mérőrendszer 1 mérési értékét a felfutó élnél tárolni.
N230 R11=$AA_MM2[X]	; Mérőrendszer 2 mérési értékét a felfutó élnél tárolni.
N240 R12=$AA_MM3[X]	; Mérőrendszer 1 mérési értékét a lefutó élnél tárolni.
N250 R13=$AA_MM4[X]	; Mérőrendszer 2 mérési értékét a lefutó élnél tárolni.
N260 ENDE:	

Példa 2: Tengely-specifikus mérés maradék út törléssel módus 2-ben (kiértékelés időrendi sorrendben)

Programkód	Kommentár
...	
N100 MEASA[X]=(2,1,-1,2,-2) G01 X100 F100	; Mérés modus 2-ben aktív mérőrendszerrel. Várakozás a mérőjelre a mérő-tapintó 1 felfutó él, mérő-tapintó 1 lefutó él, mérő-tapintó 2 felfutó él és a mérő-tapintó 2 lefutó él sorrendben, az X = 100 elmozdulás alatt.
N110 IF $AC_MEA[1]==FALSE GOTOF MESSTASTER2	; Mérő-tapintó 1 mérés sikerességét ellenőrizni.
N120 R10=$AA_MM1[X]	; Az első programozott trigger-eseményhez (mérőtapintó 1 felfutó éle) tartozó mérési értéket tárolni.
N130 R11=$AA_MM2[X]	; A második programozott trigger-eseményhez (mérőtapintó 1 felfutó éle) tartozó mérési értéket tárolni.
N140 MESSTASTER2:	
N150 IF $AC_MEA[2]==FALSE GOTOF ENDE	; Mérő-tapintó 2 mérés sikerességét ellenőrizni.

Programkód	Kommentár
N160 R12=$AA_MM3[X]	; A harmadik programozott trigger-esemény-hez (mérőtapintó 2 felfutó éle) tartozó mérési értéket tárolni.
N170 R13=$AA_MM4[X]	; A negyedik programozott trigger-esemény-hez (mérőtapintó 2 felfutó éle) tartozó mérési értéket tárolni.
N180 ENDE:	

Példa 3: Tengely-specifikus folyamatos mérés modus 1-ben (kiértékelés időrendi sorrendben)

a) Max. 100 mérési érték mérése

Programkód	Kommentár
...	
N110 DEF REAL MESSWERT[100]	
N120 DEF INT Schleife=0	
N130 MEAC[X]=(1,1,-1) G01 X1000 F100	; Mérés módus 1-ben az aktív mérőrendszer-rel, a mérési értékek tárolása az $AC_FI-FO1-ben, várakozás a mérőjelre lefutó él-lel a mérőtapintó 1-től az X = 1000 elmoz-dulási úton.
N135 STOPRE	
N140 MEAC[X]=(0)	; Mérést a tengelypozíció után megszakí-tani.
N150 R1=$AC_FIFO1[4]	; A befutott mérési értékek számát az R1 paraméterben tárolni.
N160 FOR Schleife=0 TO R1-1	
N170 MESSWERT[Schleife]=$AC_FIFO1[0]	; A mérési értékeket az $AC_FIFO1-ből ki-olvasni és tárolni.
N180 ENDFOR	

b) Mérés maradék-út törléssel 10 mérési érték után

Programkód	Kommentár
...	
N10 WHEN $AC_FIFO1[4]>=10 DO MEAC[x]=(0) DELDTG(x)	; Maradék-út törlés.
N20 MEAC[x]=(1,1,1,-1) G01 X100 F500	
N30 MEAC[X]=(0)	
N40 R1=$AC_FIFO1[4]	; Mérési értékek száma.
...	

c) Egy felfutó/lefutó fogél mérése 2 mérőtapintóval

Programkód	Kommentár
...	
N110 DEF REAL MESSWERT[16]	

Programkód	Kommentár
N120 DEF INT Schleife=0	
N130 MEAC[X]=(1,1,-1,2) G01 X100 F100	; Mérés módus 1-ben az aktív mérőrendszerrel, a mérési értékek tárolása az $AC_FIFO1-ben, várakozás a mérőjelre a mérőtapintó 1 lefutó él, mérőtapintó 2 felfutó él sorrendben, X = 100 elmozdulási úton.
N140 STOPRE	; Előrefutás-állj
N150 MEAC[X]=(0)	; Mérést a tengelypozíció után megszakítani.
N160 R1=$AC_FIFO1[4]	; A befutott mérési értékek számát az R1 paraméterben tárolni.
N170 FOR Schleife=0 TO R1-1	
N180 MESSWERT[Schleife]=$AC_FIFO1[0]	; A mérési értékeket az $AC_FIFO1-ből kiolvasni és tárolni.
N190 ENDFOR	

További információk

Mérési feladat

Egy mérési feladat programozása történhet a munkadarabprogramban vagy egy szinkronakcióból (lásd "Szinkron-akciók (Oldal 601)") fejezet. Tengelyenként ugyanabban az időpontban csak egy mérési feladat lehet aktív.

Megjegyzés

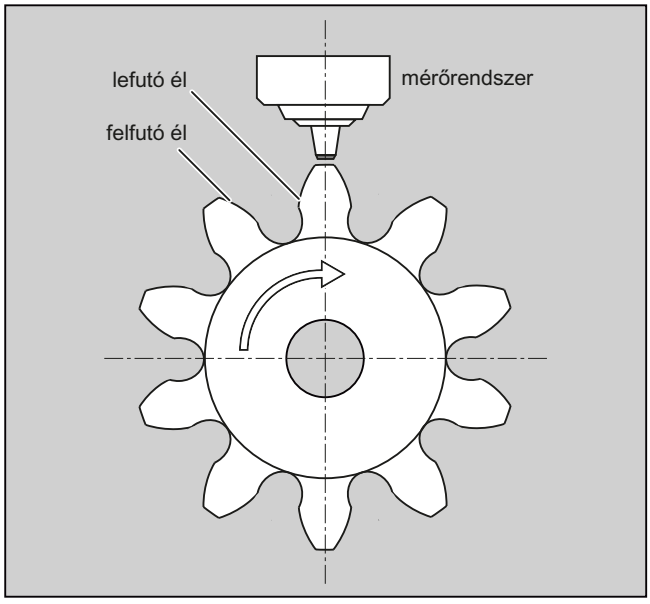
Az előtolást a mindenkori mérési feladathoz kell illeszteni.

A `MEASA` és a `MEAWA` esetében korrekt eredményeket csak olyan előtolásoknál lehet garantálni, amelyeknél helyzetszabályzó-ütemenként egynél nem több azonos és 4-nél nem több különböző trigger-esemény történik.

A folyamatos mérésnél `MEAC`-cal az interpolációs-ütem és a helyzetszabályzó-ütem aránya nem lehet nagyobb 1:8-nál.

Trigger-esemény

Egy trigger-esemény a mérő-tapintó számából és a mérőjel kapcsolási kritériumából (felfutó vagy lefutó él) tevődik össze.



Minden mérésre az alkalmazott mérő-tapintók max. 4 trigger-eseményét lehet feldolgozni, vagyis max. két mérő-tapintót egyenként két mérési éllel. A feldolgozás sorrendje és trigger-események maximális száma függ a kiválasztott módustól.

Megjegyzés

A modus 1-re érvényes: Ugyanazt a trigger-eseményt egy mérési feladatban csak egyszer szabad programozni.

`MEAC`-nál a mérési értékek száma trigger eseményenként a PROFIBUS távirat 395 alkalmazásával összesen 8 mérés-értékre felfutó és 8-ra lefutó élnél növelhető trigger eseményenként és helyzetszabályozó ütemenként.

- egy mérőtapintó: 8 mérésérték felfutó és 8 lefutó élnél
- két mérőtapintó: mérőtapintónként 4 mérésérték felfutó és 4 lefutó élnél

Ezáltal a PROFIBUS távirat 395 alkalmazásánál magasabb előtolásokat és fordulatszámokat lehet megvalósítani.

Irodalom:
Bővítő funkciók működési kézikönyv; Mérés (M5), fejezet: Tengely mérés

Üzemmód

A módus első számjegyével (tízes hely) lehet kiválasztani a kívánt mérőrendszert. Ha csak egy mérőrendszer van, de a másodikat programozták, automatikusan a meglevő lesz alkalmazva.

A második számjeggyel (egyes jegy) lesz kiválasztva a mérés-módus. Ezzel illesztjük a mérési eljárást a mindenkori vezérlés lehetőségeihez

- Modus 1
  A trigger-események kiértékelése a fellépésük időbeli sorrendjében történik. Ebben a módusban a hattengelyes modulok alkalmazásánál csak egy trigger-esemény programozható ill. több trigger esemény megadásánál automatikusan a módus 2-be lesz átállítva (jelzés nélkül).
- Modus 2
  A trigger-események kiértékelése a programozott sorrendben történik.
- Modus 3
  A trigger-események kiértékelése a programozott sorrendben történik, de a trigger-esemény 1 felügyelete nélkül START-nál.

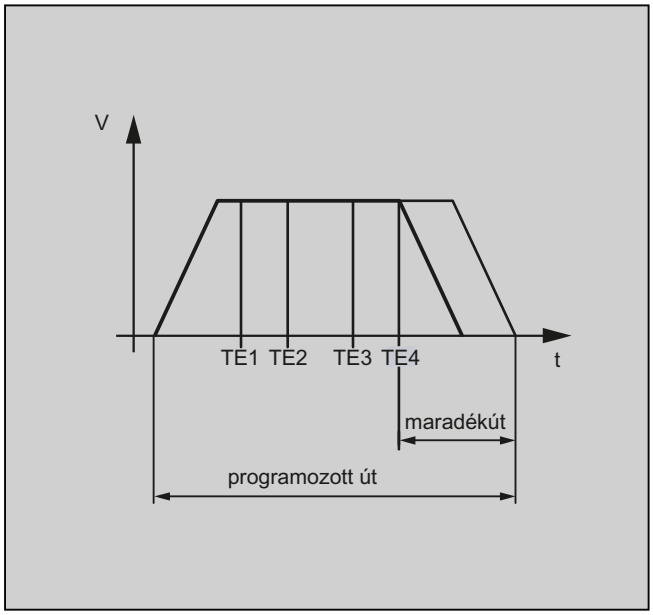
Megjegyzés

2 mérőrendszer alkalmazásánál csak két trigger-esemény programozható.

Mérés maradék-út törléssel és anélkül

A `MEASA` programozásánál a maradékút törlés csak az összes szükséges mérési érték felvétele után történik meg.

A `MEAWA` speciális mérési feladatokhoz lesz használva, amelyeknél mindenképpen a programozott pozíciót kell felvenni,



Megjegyzés

A `MEASA` nem programozható szinkronakciókban. Helyettesítésként lehet a `MEAWA`-t plusz maradékút törlést szinkronakcióként programozni.

Ha a mérési feladat `MEAWA`-val szinkronakcióból van indítva, a mérési értékek csak gép-koordinátarendszerben állnak rendelkezésre.

Mérési eredmények MEASA, MEAWA esetében

A mérés eredményei a következő rendszerváltozókban állnak rendelkezésre:

- gép-koordinátarendszerben:

`$AA_MM1[<tengely>]`	programozott mérőrendszer méréseredménye trigger-esemény 1-nél
...	...
`$AA_MM4[<tengely>]`	programozott mérőrendszer méréseredménye trigger-esemény 4-nél

- munkadarab-koordinátarendszerben:

`$AA_MW1[<tengely>]`	programozott mérőrendszer méréseredménye trigger-esemény 1-nél
...	...
`$AA_MW4[<tengely>]`	programozott mérőrendszer méréseredménye trigger-esemény 4-nél

Geometria-tengelyek / transzformációk

Ha a tengelyirányú mérést egy geometria-tengelyre akarjuk elindítani, ugyanazt a mérési feladatot a többi geometria-tengelyre is programozni kell. Ugyanez érvényes azokra a tengelyekre is, amelyek egy transzformációban vesznek részt.

Példák:
```
N10 MEASA[Z]=(1,1) MEASA[Y]=(1,1) MEASA[X]=(1,1) G0 Z100
```

vagy
```
N10 MEASA[Z]=(1,1) POS[Z]=100
```

Mérési feladat 2 mérőrendszerrel

Ha egy mérési feladat két mérőrendszerrel lesz végrehajtva, a mindenkori tengely mindkét mérőrendszerének mindkét lehetséges trigger-eseménye fel lesz dolgozva. A foglalt rendszerváltozók kitöltése a következő:

`$AA_MM1[<tengely>]`	ill.:	`$AA_MW1[<tengely>]`	mérőrendszer 1 mérési értéke trigger-esemény 1-nél
`$AA_MM2[<tengely>]`	ill.:	`$AA_MW2[<tengely>]`	mérőrendszer 2 mérési értéke trigger-esemény 1-nél
`$AA_MM3[<tengely>]`	ill.:	`$AA_MW3[<tengely>]`	mérőrendszer 1 mérési értéke trigger-esemény 2-nél
`$AA_MM4[<tengely>]`	ill.:	`$AA_MW4[<tengely>]`	mérőrendszer 2 mérési értéke trigger-esemény 2-nél

Rendszerváltozók

A mérő-tapintó állapota a következő rendszerváltozókban áll rendelkezésre:

$A_PROBE[<n>]

Érték	Jelentés
1	mérő-tapintó kitért
0	mérő-tapintó nem tért ki

A mérőtapintó határolás a következő rendszerváltozókban áll rendelkezésre:

$A_PROBE_LIMITED[<n>]

Érték	Jelentés
1	mérőtapintó határolás aktív
0	mérőtapintó határolás nem aktív

<n> = mérőtapintó

Irodalom:
Rendszerváltozók jegyzékes kézikönyv

Mérési feladat állapota MEASA, MEAWA esetén

Ha a programban kiértékelés szükséges, a mérési feladat állapotát a $AC_MEA[<n>]-nel, ahol <n> = mérő-tapintó száma, lehet lekérdezni. Ha az <n> mérő-tapintónak az összes egy mondatban programozott trigger-eseménye bekövetkezett, ez a változó az 1 értéket adja. Egyébként az érték 0.

Megjegyzés

Ha a mérés nem szinkronakcióból lett indítva, az $AC_MEA nem lesz tovább aktualizálva. Ebben az esetben a DB31, ... DBX62.3 új PLC interfészjelet ill. az azonos értékű $AA_MEAACT[<tengely>] változót kell lekérdezni.

Jelentés:

$AA_MEAACT==1: mérés aktív

$AA_MEAACT==0: mérés nem aktív

Folyamatos mérés (MEAC)

A mérési értékek `MEAC`-nál a gép-koordinátarendszerben keletkeznek és a megadott FIFO[n]-tárolóban (kör-tároló) lesznek letéve. Ha mérésre két mérő-tapintó van megtervezve, a második mérő-tapintó mérési értékei külön a kiegészítőleg erre megtervezett (MD-vel beállítható) FIFO[n+1]-tárolóba lesznek letéve.

A FIFO-tároló egy kör-tároló, amibe a mérési értékek a körbefutás elv szerint a $AC_FIFO változókba lesznek bevive, (lásd"Szinkron-akciók (Oldal 601)" fejezetben).

Megjegyzés

A FIFO-tartalmat csak egyszer lehet a kör-tárolóból kiolvasni. A mérési adatok többszöri alkalmazásához az alkalmazói adatokat átmenetileg tárolni kell.

Ha a mérési értékek száma túllépi a FIFO-tárolóra gépadatban megadott maximumot, a mérés automatikusan befejeződik.

Végtelen mérést a mérési értékek ciklikus kiolvasásával lehet megvalósítani. Ehhez a kiolvasásnak legalább olyan gyakorisággal kell történni, mint az új mérési értékek bevitelének.

Irodalom:

- Szinkronakciók működési kézikönyv; Részletes leírás, fejezet: Paraméter ($AC_FIFO)
- Bővítő funkciók működési kézikönyv; Mérés(M5), fejezet: Tengely mérés

Védelem a hibás programozástól

A következő programozási hibák felismerésre és hibával kijelzésre kerülnek:

- `MEASA`/`MEAWA` és a `MEAS`/`MEAW` egy mondatban van programozva
  Példa:
  `N01 MEAS=1 MEASA[X]=(1,1) G01 F100 POS[X]=100`
- `MEASA`/`MEAWA` <2 vagy >5 paraméterszámmal
  Példa:
  `N01 MEAWA[X]=(1) G01 F100 POS[X]=100`
- `MEASA`/`MEAWA` 1/ -1/ 2/ -2-vel nem egyenlő trigger-eseménnyel
  Példa:
  `N01 MEASA[B]=(1,1,3) B100`
- `MEASA`/`MEAWA` helytelen modussal
  Példa:
  `N01 MEAWA[B]=(4,1) B100`
- `MEASA`/`MEAWA` duplán programozott trigger-eseménnyel
  Példa:
  `N01 MEASA[B]=(1,1,-1,2,-1) B100`
- `MEASA`/`MEAWA` és hiányzó geometria-tengely
  Példa:
  `N01 MEASA[X]=(1,1) MEASA[Y]=(1,1) G01 X50 Y50 Z50 F100 ;geo-tengely X/Y/Z`
- Nem egységes mérési feladat geometria-tengelyeknél
  Példa:
  `N01 MEASA[X]=(1,1) MEASA[Y]=(1,1) MEASA[Z]=(1,1,2) G01 X50 Y50 Z50 F100`

5.9 Speciális funkciók OEM-felhasználóknak (OMA1 ... OMA5, OEMIPO1, OEMIPO2, G810 ... G829)

OEM-címek

Az OEM-címek jelentését az OEM-felhasználó határozza meg. A funkcionalitást a Compile-ciklusok adják. 5 OEM-cím foglalt (`OMA1` ... `OMA5`). A cím jelölők beállíthatók. Az OEM-címek minden mondatban megengedhetők.

Foglalt G utasítás hívások

Az OEM-alkalmazók számára foglalt G utasítás hívások:

- `OEMIPO1`, `OEMIPO2` (G-csoport 1-ből)
- `G810` ... `G819` (G-csoport 31)
- `G820` ... `G829` (G-csoport 32)

A funkcionalitást a Compile-ciklusok adják.

Funkciók és alprogramok

Ezen kívül az OEM-felhasználók létrehozhatnak előre definiált funkciókat és alprogramokat is paraméterátadással.

Megjegyzés

Munkadarab szimuláció

A SW 4.4-ig semmi, a SW 4.4-től csak kiválasztott Compile ciklusok (CC) támogatottak a munkadarab szimulációnál

Utasítások a munkadarabprogramban a nem támogatott Compile ciklusokkal (`OMA1` ... `OMA5`, `OEMIPO1/2`, `G810` ... `G829`, saját eljárások és funkciók) egyedi kezelés nélkül vészjelzéshez és a szimuláció megszakításához vezetnek.

Megoldás: A hiányzó CC-specifikus nyelvi elemeket a munkadarabprogramban egyedileg kezelni ($P_SIM-lekérdezés).
Példa:

```
N1 G01 X200 F500
IF (1==$P_SIM)
N5 X300 ;szimulációnál CC nem aktív
ELSE
N5 X300 OMA1=10
ENDIF
```

5.10 Előtolás-csökkentés sarokkésleltetéssel (FENDNORM, G62, G621)

Az automatikus sarokkésleltetésnél az előtolás harangformában kevéssel az érintett sarok előtt lesz lecsökkentve. Ezenkívül lehetséges a megmunkálásban érintett szerszámtartó külső méreteit beállítási adatokkal paraméterezni. Ezek a következők:

- Előtolás-csökkentés kezdete és vége
- Override, amivel az előtolás csökkentve lesz
- A releváns sarok felismerése

Releváns sarkok azok a sarkok, amelyek belső szöge kisebb a beállítási adatként paraméterezett saroknál.

A `FENDNORM` alapértékkel az automatikus sarok-override funkció ki lesz kapcsolva.

Irodalom:
/FBFA/ ISO-dialektusok működési leírás

Szintaxis

```
FENDNORM
G62 G41
G621
```

Jelentés

`FENDNORM`:	automatikus sarokkésleltetés ki
`G62`:	sarokkésleltetés a belső sarkokon aktív szerszámsugár-korrekciónál
`G621`:	sarokkésleltetés minden sarkokon aktív szerszámsugár-korrekciónál

G62 csak a belső sarkokon hat a következőknél

- aktív szerszámsugár-korrekció `G41`, `G42` és
- aktív pályavezérlő-üzem `G64`, `G641` és

A megfelelő sarokra menet csökkentett sebességgel történik a következő szerint:

F * (override előtolás-csökkentéshez) * előtolás-override

A maximálisan lehetséges előtolás-csökkentés pontosan akkor lesz elérve, amikor a szerszám a középpont-pályára vonatkoztatva az érintett saroknál az irányváltást végrehajtja.

A G621 a G62-vel azonosan minden sarkon hat, az `FGROUP`-pal megadott tengelyeknél.

5.11 Programozható mozgás vége kritériumok (FINEA, COARSEA, IPOENDA, IPOBRKA, ADISPOSA)

A pálya-interpoláció (G601, G602 und G603) mondatváltási kritériumaihoz hasonlóan a mozgás vége kritériumot az egyes tengely interpolációnál egy munkadarab-programban ill. szinkronakciókban parancs-/ PLC-tengelynél programozható.

Aszerint, milyen mozgás vége kritérium van beállítva, a munkadarab-program mondatok ill. technológia ciklus mondatok egyes tengely mozgásoknál különböző gyorsan lesznek befejezve. Ugyanez érvényes a PLC-re FC15/16/18 esetében.

Szintaxis

```
FINEA[<tengely>]
COARSEA[<tengely>]
IPOENDA[<tengely>]
IPOBRKA(<tengely>[,<időpont>])
ADISPOSA(<tengely>[,<modus>,<ablak nagyság>])
```

Jelentés

FINEA:	mozgás vége kritérium: "Pontos-állj finom"		
	hatásosság:	modális	
COARSEA:	mozgás vége kritérium: "Pontos-állj durva"		
	hatásosság:	modális	
IPOENDA:	mozgás vége kritérium: "Interpoláció-állj"		
	hatásosság:	modális	
IPOBRKA:	Mondatváltás kritérium: fékrámpa		
	hatásosság:	modális	
ADISPOSA:	tűrés-ablak mozgás vége kritériumhoz		
	hatásosság:	modális	
<tengely>:	csatorna-tengely név (X, Y, ...)		
<időpont>:	mondatváltás időpontja a fékrámpára vonatkoztatva %-ban: • 100% = fékrámpa kezdete • 0% = fékrámpa vége, azonos jelentésű IPOENDA-val		
	típus:	REAL	
<modus>:	tűrés-ablak vonatkoztatása		
	értéktartomány:	0	tűrésablak nem aktív
		1	tűrésablak parancspozícióra vonatkoztatva
		2	tűrésablak valóspozícióra vonatkoztatva
	típus:	INT	
<ablak nagyság>:	tűrésablak nagysága		
	típus:	REAL	

Példák

Példa 1: mozgás vége kritérium: "Interpoláció-állj"

Programkód

```
; X pozícionáló tengely menjen 100-ra, sebesség 200 m/
perc, gyorsítás 90 %,
; mozgás vége kritérium: Interpoláció-állj
N110 G01 POS[X]=100 FA[X]=200 ACC[X]=90 IPOENDA[X]

; szinkron-akció:
; MINDIG HA: bemenet 1 bekapcsolva
; AKKOR az X pozícionáló tengely menjen 50-ra, sebesség
200 m/perc, gyorsítás 140%,
; mozgás vége kritérium: Interpoláció-állj
N120 EVERY $A_IN[1] DO POS[X]=50 FA[X]=200 ACC[X]=140
IPOENDA[X]
```

Példa 2: Mondatváltás kritérium: "fékrámpa"

Programkód	Kommentár
	; Alap-beállítás hatásos
N40 POS[X]=100	; X pozicionáló mozgása a pozíció 100-ra
	Mondatváltás kritérium: pontos-állj finom
N20 IPOBRKA(X,100)	; Mondatváltás kritérium: "fékrámpa",
	100% = fékrámpa kezdete
N30 POS[X]=200	; Mondatváltás történik, amikor az X tengely fékezni kezd
N40 POS[X]=250	; X tengely nem fékezik tovább a pozíció 200-ra, hanem tovább megy a pozíció 250-re.
	A mondatváltás a tengely fékezésének megkezdésekor történik.
N50 POS[X]=0	; X tengely fékez és visszamegy a pozíció 0-ra.
	Mondatváltás a 0 pozíciónál és "Pontos-állj finom"-nál történik
N60 X10 F100	; X tengely pályatengelyként mozog a pozíció 10-re

További információk

Rendszerváltozó mozgás-vége kritériumhoz

A hatásos mozgás-vége kritériumot $AA_MOTEND rendszerváltozóval lehet olvasni.

Irodalom: /LIS2sl/ Listák kézikönyv, könyv 2

Mondatváltás kritérium: "fékrámpa" (IPOBRKA)

Ha a "fékrámpa" mozgás-vége kritérium aktiválásánál az opcionális mondatváltás időpontra egy érték van programozva, ez a következő pozícionáló mozgásra lesz hatásos és főfutás-szinkron a beállítási adatba lesz beírva. Ha a mondatváltás időpontra nincs megadva érték, a beállítási adat aktuális értéke lesz hatásos.

SD43600 $SA_IPOBRAKE_BLOCK_EXCHANGE

Egy tengely mozgás-vége kritérium következő programozásánál (`FINEA`, `COARSEA` , `IPOENDA`) az `IPOBRKA` a megfelelő tengelyre deaktiválva lesz.

Kiegészítő mondatváltás kritérium: "Tűrés-ablak" (ADISPOSA)

A `ADISPOSA`-val kiegészítő mondatváltás kritériumként lehet definiálni egy tűrés-ablakot a mondatvégpont körül (választhatóan valós- vagy parancspozíció). A mondatváltáshoz ekkor két feltétel kell teljesüljön:

- Mondatváltás kritérium: "fékrámpa"
- Mondatváltás kritérium: "tűrés-ablak"

Irodalom

További információk a pozícionáló-tengelyek mondatváltás-kritériumához:

- Működési kézikönyv Bővítő funkciók; Pozícionáló tengelyek (P2)
- Programozási kézikönyv, Alapok; "Előtolás-szabályozás" fejezet

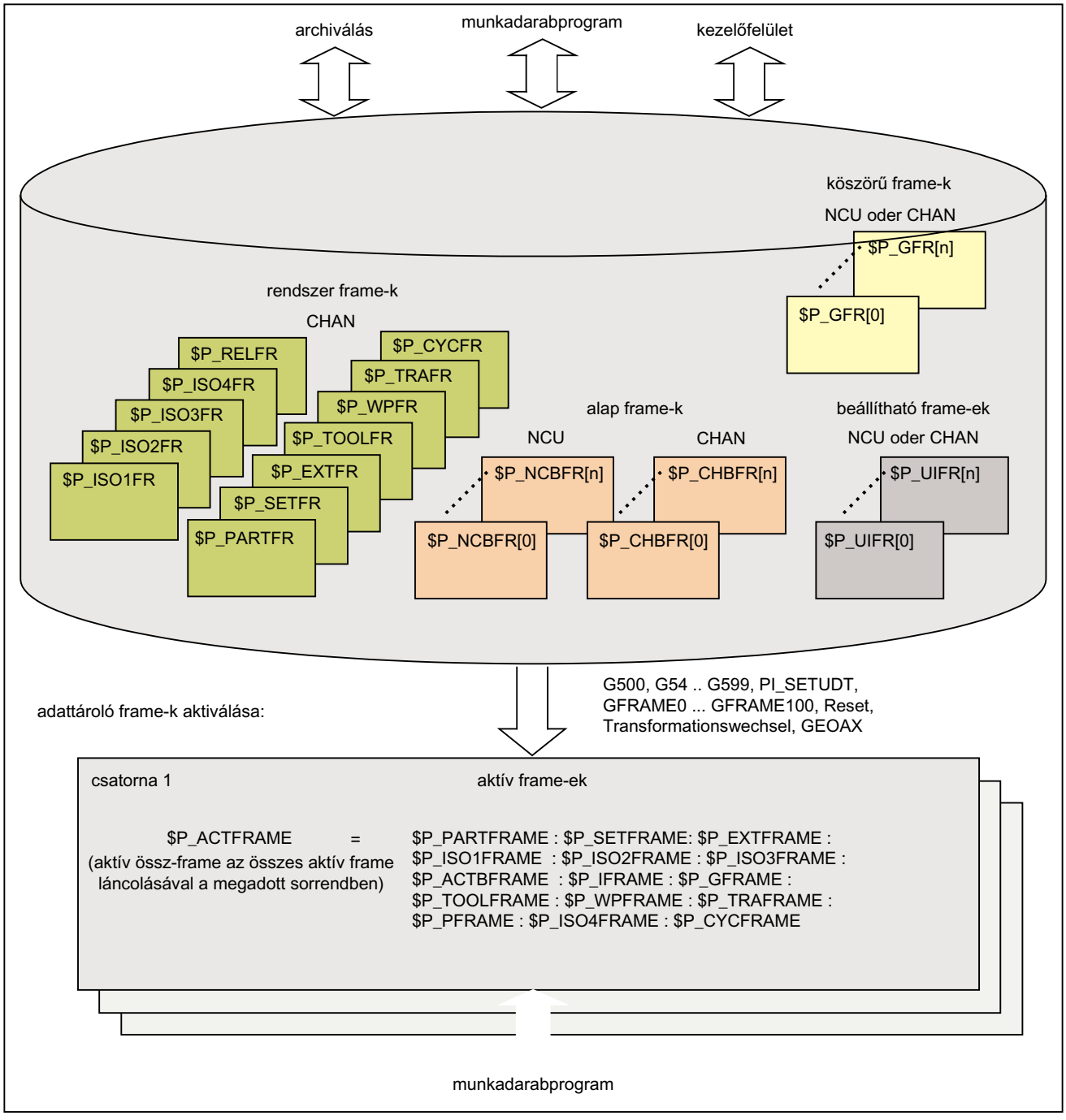
Koordináta-transzformációk (frame-k) 6

6.1 Koordináta-transzformáció frame változókkal

Az Alapok programozási kézikönyv, "Koordináta-transzformációk (Frames)" leírt utasítások mellett, mint pl. `ROT`, `AROT`, `SCALE`, stb. munkadarab-koordinátarendszert (MKR) az $P_...FR (adattároló frame-k) és a az $P_...FRAME (aktív frame-k) frame változókkal is lehet transzformálni.

A következő kép áttekintést ad a frame változók strukturálásáról:

- adattároló frame-k
- aktív frame-ek
- aktív össz-frame-ek: összes aktív frame láncolása
- NCU-globális frame-k
- csatorna-specifikus frame-k



Kép 6-1 Frame változók áttekintése

6.1.1 Előre definiált frame változók ($P_CHBFRAME, $P_IFRAME, $P_PFRAME, $P_ACTFRAME)

Aktív: csatorna-specifikus alap frame $P_CHBFRAME[<n>] ($P_BFRAME)

Megjegyzés

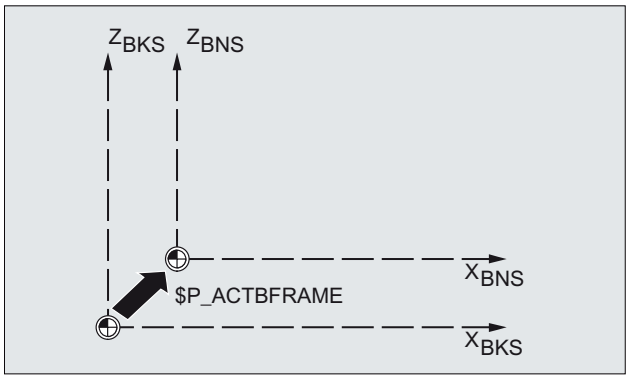
Az aktuális `$P_BFRAME` alap-frame és az `$P_UBFR` adattároló alap-frame kompatibilitás okokból meg lesznek tartva.

- `$P_BFRAME` □ `$P_CHBFRAME[0]`
- `$P_UBFR` □ `$P_CHBFR[0]`.

A $P_CHBFRAME[<n>] frame változók definiálják a kapcsolatot alap-koordinátarendszer (AKR) és alap-nullapontrendszer (ANR) között.

Ha az $P_CHBFRAME[`<n>`] csatorna-specifikus alap-frame az NC programban azonnal hatásos kell legyen, a következő lehetőségek állnak rendelkezésre

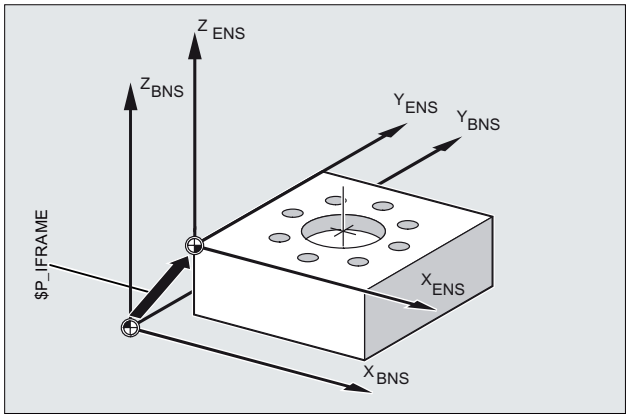
- Utasítások
  - `G500` (Az összes beállítható frame kikapcsolása, az alap-frame-k aktívak maradnak)
  - `G54` ... `G599` (beállítható nullaponteltolások)
- Az adattárolás egy csatorna-specifikus alap-frame-jének hozzárendelése egy aktuális csatorna-specifikus alap-frame-hez:
  `$P_CHBFRAME[<n>] = $P_CHBFR[<m>]`



Aktív: $P_IFRAME csatorna-specifikus beállítható frame

A $P_IFRAME frame változó definiálja a kapcsolatot az alap-nullapontrendszer (ANR) és beállítható nullapontrendszer (BNR) között.

- `$P_IFRAME` megfelel `$P_UIFR[$P_IFRNUM]`-nak
- `$P_IFRAME` tartalma pl. `G54` programozása után a `G54` által definiált átalakítás, forgatás, skálázás és tükrözés.

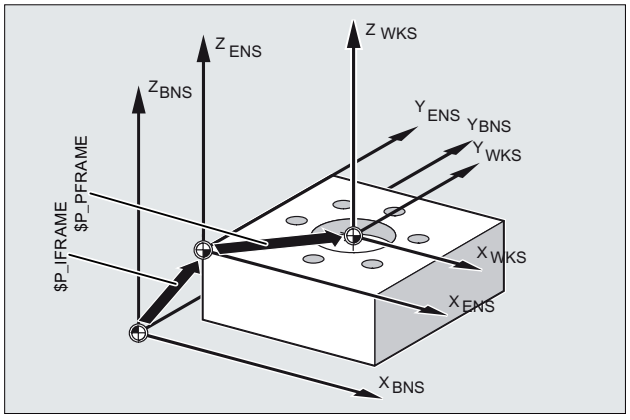


Aktív: $P_PFRAME csatorna-specifikus programozható frame

A $P_PFRAME frame változó definiálja a kapcsolatot a beállítható nullapontrendszer (BNR) és a munkadarab-koordinátarendszer (MKR) között.

`$P_PFRAME` tartalmazza az eredő frame-t, ami

- **programozásból**: `TRANS/ATRANS`, `ROT/AROT`, `SCALE/ASCALE`, `MIRROR/AMIRROR` ill.
- **hozzárendelésből**: `CTRANS`, `CROT`, `CMIRROR`, `CSCALE` a programozható FRAME-hez adódik.

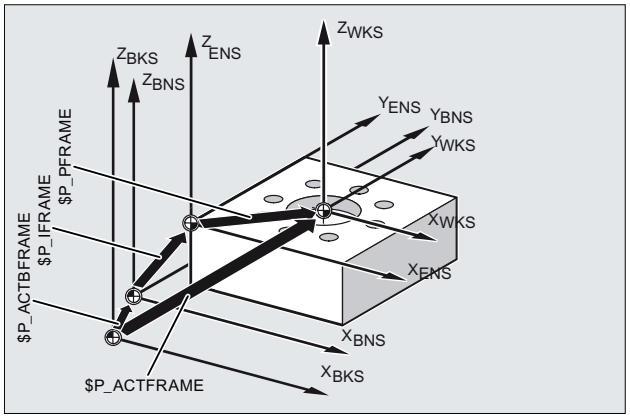


Aktív: $P_ACTFRAME össz-frame

A csatornában hatásos össz-frame az összes, a csatornában hatásos frame láncolásából adódik.

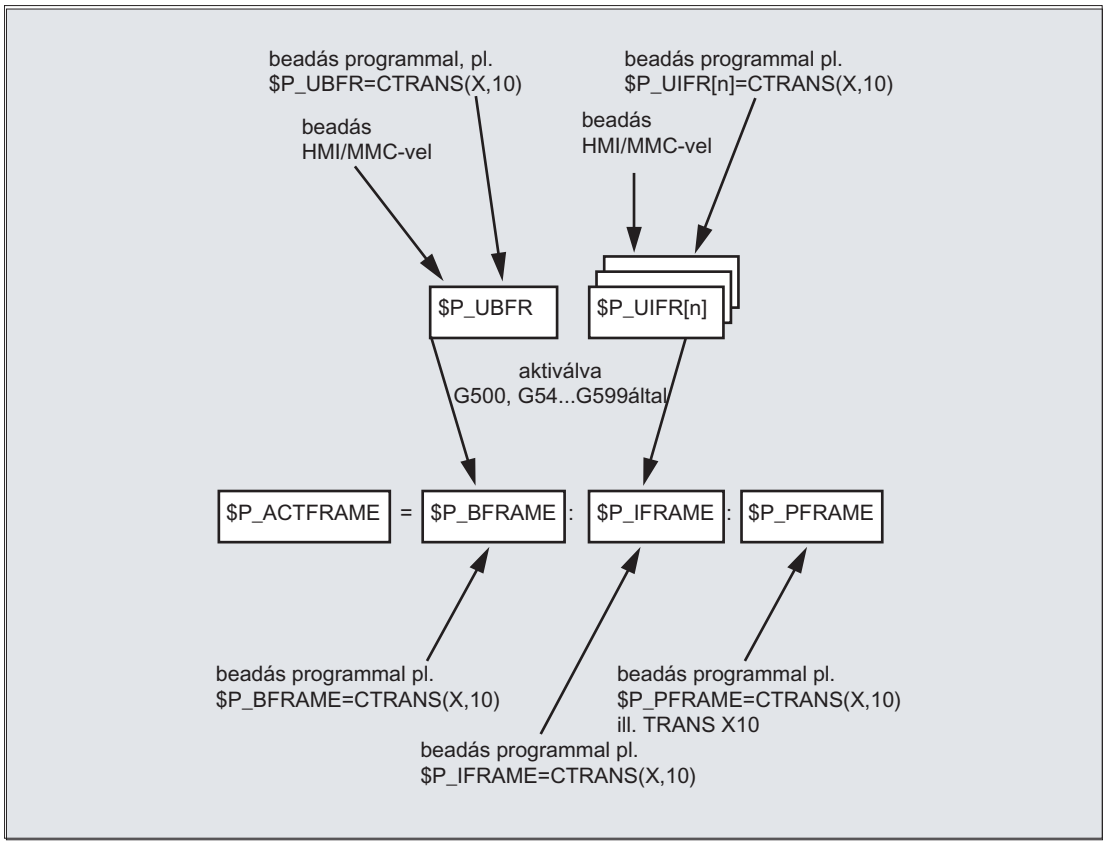
```
$P_ACTFRAME =    $P_PARTFRAME : $P_SETFRAME  : $P_EXTFRAME :
                 $P_ISO1FRAME : $P_ISO2FRAME : $P_ISO3FRAME :
                 $P_ACTBFRAME : $P_IFRAME    : $P_GFRAME :
                 $P_TOOLFRAME : $P_WPFRAME   : $P_TRAFRAME :
                 $P_PFRAME    : $P_ISO4FRAME : $P_CYCFRAME
```

`$P_ACTFRAME` leírja az aktuális munkadarab koordinátarendszer nullapontját.



Kép 6-2 $P_ACTFRAME frame-változó

Ha a következő frame-k `$P_BFRAME` / `$P_CHBFRAME[<n>]`, `$P_IFRAME` vagy `$P_PFRAME` egyike megváltozik, a `$P_ACTFRAME` aktuális össz-frame újra lesz számítva.



Alap-frame és beállítható frame Reset után hatásosak, ha az MD 20110 RESET_MODE_MASK a következőképpen van beállítva:

bit0=1, bit14=1 --> `$P_UBFR` (alap-frame) hat

bit0=1, bit5=1 --> `$P_UIFR[$P_UIFRNUM]` (beállítható frame) hat

Adattárolás: Csatorna-specifikus alap frame $P_CHBFR[<n>]

A $P_CHBFR[<n>] frame változóval lesznek olvasva / írva az alap-frame-k az adattárolásban. Az adattárolás frame az írással nem lesz azonnal aktív a csatornában. Az írt frame aktiválása a következővel történik:

- csatorna Reset és MD20110 $MC_RESET_MODE_MASK,bit0 == 1 és bit14 == 1
- `G500`, `G54` ... `G57`, `G505` ... `G599` utasítás (alap-frame be/kikapcsolása utána az aktuális össz-frame újra számításával)

Adattárolás: $P_UIFR[<n>] csatorna-specifikus beállítható frame

A$P_UIFR[<n>] frame változóval lesznek olvasva / írva a beállítható frame-k az adattárolásban. A frame az írással nem lesz azonnal aktív a csatornában. Az írt frame beszámítása a csatornában a következővel történik:

- `G500` utasítás (az összes beállítható frame ill, nullaponteltolás kikapcsolása)
- `G54` ... `G57`, `G505` ... `G599` utasítás (egy beállítható frame ill, nullaponteltolás bekapcsolása)

Aktív beállítható frame	Adattároló frame	(megfelel utasításnak)
`$P_IFRAME` =	`$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 Érték-hozzárendelés frame-ekhez

6.2.1 Közvetlen érték-hozzárendelés (tengelyérték, szög, mérték)

Az NC programban lehet értékeket közvetlenül hozzárendelni, frame-khez vagy frame-változókhoz.

Szintaxis

Szintaxis

```
$P_PFRAME = CTRANS(X, <eltolás-érték>, Y, <eltolás-érték>, Z,
<eltolás-érték>, ...)
```

```
$P_PFRAME = ROT(X, <szög>, Y, <szög>, Z, <szög>, ...)
```

```
$P_UIFR[..] = CROT(X, <szög>, Y, <szög>, Z, <szög>,  ...
```

```
$P_PFRAME = CSCALE(X, <érték>, Y, <érték>, Z, <érték>, ...)
```

```
$P_PFRAME = CMIRROR(X, Y, Z)
```

`$P_CHBFRAME[<n>]` szintakszisa azonos a `$P_PFRAME`-ével.

Jelentés

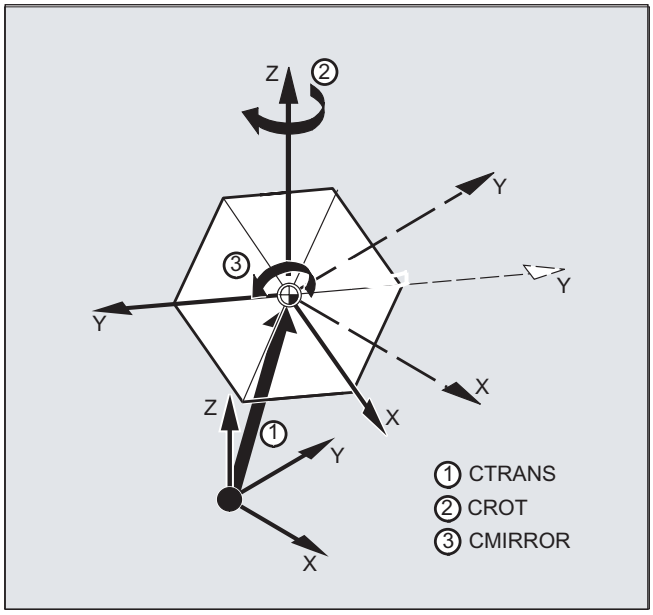
`CTRANS`:	eltolás a megadott tengelyeken
`CROT`:	forgatás a megadott tengelyek körül
`CSCALE`:	mérték változtatás a megadott tengelyen
`CMIRROR`:	megadott tengely irányának megfordítása
`X, Y, Z`:	eltolás-érték a megadott geometriai tengely irányában
`<eltolás-érték>`:	eltolás-érték
`<szög>`:	szög, amivel forgatva lesz
`<érték>`:	érték megadás

Példák

Érték hozzárendelések az aktuális programozható frame frame komponenseihez

Érték hozzárendelés az aktuális programozható frame eltolás, forgatás és tükrözés frame komponenseihez:

```
$P_PFRAME = CTRANS(X,10,Y,20,Z,5) : CROT(Z,45) : CMIRROR(Y)
```



Egy frame forgatás komponensének írása

Értékek hozzárendelése `CROT`-tal a `$P_UIFR` beállítható adattároló frame forgatás komponenseinek mindhárom tengelyéhez:

```
$P_UIFR[5] = CROT(X, 0, Y, 0, Z, 0)
```

Ennek alternatívája az egyes értékek közvetlen hozzárendelése az adattároló frame forgatás komponenseinek mindenkori tengelyeihez:

```
$P_UIFR[5, Y, RT]=0
$P_UIFR[5, X, RT]=0
$P_UIFR[5, Z, RT]=0
```

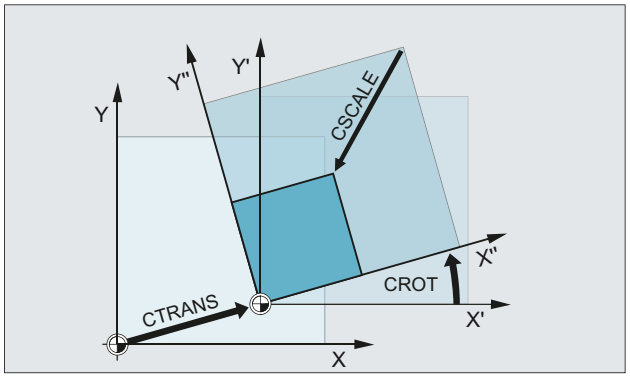
Leírás

Több művelet egy frame-n a `:` láncolás operátorral köthető össze. A műveletek egymás után balról jobbra lesznek végrehajtva.

Példa

Láncolt műveletek a $P_PFRAME frame-n eltolással, forgatással és skálázással:

```
$P_PFRAME = CTRANS(...) : CROT(...) : CSCALE...
```



6.2.2 Frame komponensek olvasása és megváltoztatása (TR, FI, RT, SC, MI)

Önnek lehetősége van arra, hogy hozzáférjen egy frame **egyes** adataihoz, pl. egy meghatározott eltolási értékhez vagy forgásszöghöz. Ezeket az értékeket meg tudja változtatni vagy hozzá tudja rendelni egy másik változóhoz.

Szintaxis

`R10=$P_UIFR[$P_UIFNUM,X,RT]`	Az X tengely körüli RT forgásszög a $P_UIFRNUM éppen érvényes beállítható nullapont eltolásból legyen hozzárendelve az R10 változóhoz.
`R12=$P_UIFR[25,Z,TR]`	A TR eltolási érték Z-ben a beállított 25-ös frame adatrekordjából legyen hozzárendelve az R12 változóhoz.
`R15=$P_PFRAME[Y,TR]`	Az aktuális programozható frame TR eltolási értéke Y-ban legyen hozzárendelve az R15 változóhoz.
`$P_PFRAME[X,TR]=25`	Az aktuális programozható frame TR eltolási értéke X-ben legyen megváltoztatva. Mostantól X25 érvényes.

Jelentés

$P_UIFRNUM:	Ezzel a változóval automatikusan létrejön a vonatkoztatás az éppen érvényes beállítható nullapont eltoláshoz.
P_UIFR[n,…,…] :	Az n frame-szám megadásával az n számú beállítható frame-hez férhetünk hozzá.
	Komponensek megadása, amiket olvasni vagy megváltoztatni kell:
TR:	TR eltolás
FI:	FI finom eltolás
RT:	RT forgatás
SC:	SC mérték-változtatás

MI:	MI tükrözés
X, Y, Z:	Kiegészítésképpen (lásd példák) adjuk meg a megfelelő X, Y, Z tengelyt.

Értéktartomány RT forgatáshoz

forgatás az 1. geometria tengely kör- -180° ... +180°
ül:

forgatás az 2. geometria tengely kör- -90° ... +90°
ül:

forgatás az 3. geometria tengely kör- -180° ... +180°
ül:

Leírás

Frame felhívása

A $P_UIFRNUM rendszerváltozó megadásával közvetlenül hozzáférhet a $P_UIFR ill. G54, G55, …-tel éppen beállított nullapont eltoláshoz
($P_UIFRNUM tartalmazza az éppen beállított frame számát).

Az összes többi $P_UIFR eltárolt beállítható frame-t a $P_UIFR[n] megfelelő szám megadásával hívhatja fel.

Előre definiált frame változóknak és saját definiálású frame-knek a nevét adja meg, pl. $P_IFRAME.

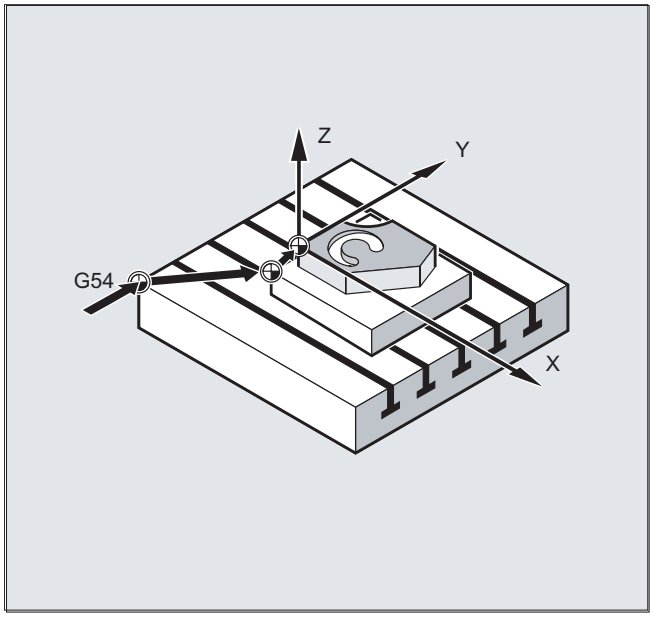
Adatok felhívása

A szögletes zárójelekben annak az értéknek a tengelyneve és frame komponensei állnak, amihez hozzá akar férni vagy amit meg akar változtatni, pl. [X, RT] vagy [Z, MI].

6.2.3 Számolás frame-ekkel

Az NC programban egy frame-t hozzá lehet rendelni egy másik frame-hez vagy frame-ket lehet láncolni.

A frame-láncolatok alkalmasak pl. több munkadarab leírására, amelyek egy palettán vannak elrendezve és egy gyártási folyamatban kell azokat megmunkálni.



Paletta feladatok leírásához a frame komponensek tartalmazhatnak pl. csak meghatározott részértékeket, amik összeláncolása által különböző munkadarab-nullapontok lesznek generálva.

Példák

Hozzárendelések

Programkód	Kommentár
`DEF FRAME EINSTELLUNG_1`	`; egy lokális frame változó definíciója`
`EINSTELLUNG_1 = CTRANS(X,10)`	`; egy funkció eredményeinek hozzárendelése a frame változókhoz`
`$P_PFRAME = EINSTELLUNG_1`	`; frame változók hozzárendelése az aktuális frame-hez`
`DEF FRAME EINSTELLUNG_4`	`; egy lokális frame változó definíciója`
`EINSTELLUNG_4 = $P_PFRAME`	`; aktuális frame közbenső tárolása a frame változókban`
`...`	
`$P_PFRAME = EINSTELLUNG_4`	`; aktuális frame visszaolvasása a frame változókból`

Láncolások

A frame utasítások a programozott sorrendben a : operátorral egymással láncolva lesznek. A frame-komponensek mint pl. eltolások, forgatások stb. egymás után összeadódóan lesznek végrehajtva.

Programkód	Kommentár
`$P_IFRAME = $P_UIFR[15] : $P_UIFR[16]`	`; A két beállítható adattároló frame láncolásából adódó`
	`; eredmény frame hozzárendelése az aktív beállítható`
	`; össz-frame-hez.`
	`; Alkalmazási példa`
	`; $P_UIFR[15]: eltolás`
	`; $P_UIFR[16]: forgatás`
`$P_UIFR[3] = $P_UIFR[4] : $P_UIFR[5]`	`; A két beállítható adattároló frame láncolásából adódó`
	`; eredmény frame hozzárendelése egy másik beállítható`
	`; adattároló frame-hez`

6.2.4 Frame változók definíciója (DEF FRAME)

Az előre definiált frame változók mellett lehet saját frame változókat is definiálni. A saját definiálású frame változók FRAME típusú felhasználói változók. A frame-ek nevét a felhasználói változókra vonatkozó szabályok keretein belül szabadon lehet megadni.

A CTRANS, CROT, CSCALE és CMIRROR funkciókkal a saját definiálású frame-ekhez értékeket lehet rendelni.

Szintaxis

```
DEF FRAME <név>
```

Jelentés

`DEF FRAME`:	FRAME típusú felhasználói változók definiálása.
<név>:	Frame változó neve

Példa

Egy "PALETTE" frame változó definiálása és eltolás és forgatás értékek hozzárendelése.

Programkód	Kommentár
`DEF FRAME PALETTE`	`; PALETTE frame változót definiálni`
`PALETTE = CTRANS(...) : CROT(...)`	`; Az eltolásból és forgatásból adódó eredmény frame`
	`; hozzárendelése a PALETTE frame változóhoz`

6.3 Durva- és finom-eltolás (CTRANS, CFINE)

Finom-eltolás

A `CFINE(...)` finom-eltolást a következő frame-eken lehet alkalmazni:

- beállítható frame-k: $P_UIFR ill. $P_IFRAME
- alap frame-k: $P_NCBFR[<n>], $P_CHBFR[<n>] ill. $P_CHBFRAMES[<n>] vagy $P_ACTBFRAME
- programozható frame $P_PFRAME

Egy frame finom-eltolása a `CFINE(...)` utasítással programozható.

Durva-eltolás

A `CTRANS(...)` durva-eltolást minden frame-en lehet alkalmazni.

Össz-eltolás

Az össz-eltolás a durva- és finom-eltolás összeadásából adódik .

Gépadatok

Finom-eltolás engedélyezése

A finom-eltolás engedélyezése a következő gépadattal történik:

MD18600 $MN_MM_FRAME_FINE_TRANS = 1

Szintaxis

Finom-eltolás

- össz-frame
  - `<Frame> = CFINE(<K_1>,<érték>)`
  - `<Frame> = CFINE(<K_1>,<érték>, <K_2>, <érték>)`
  - `<Frame> = CFINE(<K_1>,<érték>, <K_2>, <érték>, <K_3>, <érték>)`
- FRAME komponensek
  - `<Frame>[<n>, <K_1>, FI] = <érték>`

Durva-eltolás

- össz-frame
  - `<Frame> = CTRANS(<K_1>,<érték>)`
  - `<Frame> = CTRANS(<K_1>,<érték>, <K_2,<érték>)`
  - `<Frame> = CTRANS(<K_1>,<érték>, <K_2,<érték>, <K_3,<érték>)`
- FRAME komponensek
  - `<Frame>[<n>,<K_1>,TR] = <érték>`

Speciálisan a `$P_PFRAME` programozható frame-hez:

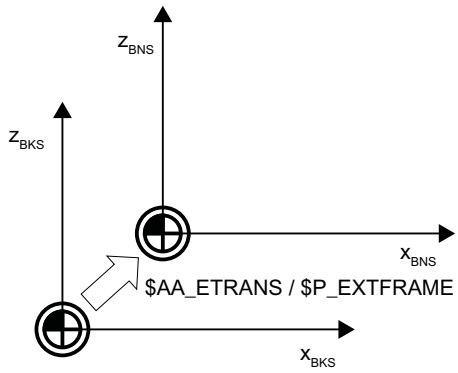
- `TRANS <K_1> <érték>`
- `TRANS <K_1> <érték><K_2> <érték>`
- `TRANS <K_1> <érték><K_2> <érték><K_3> <érték>`

Jelentés

`<Frame>`:	frame pl. beállítható adattárolás frame`$P_UIFR[<n>]`
`CFINE`:	finom-eltolás, additív eltolás
`CTRANS`:	durva-eltolás, abszolút eltolás
`TRANS`:	csak programozható frame-ek: durva-eltolás, abszolút eltolás
`<K_n>`:	X, Y, Z koordináta-tengelyek
`<érték>`:	eltolás érték

6.4 Külső nullaponteltolás ($AA_ETRANS)

A külső nullaponteltolás egy lineáris eltolás az alap-koordinátarendszer (AKR) és az alap-nullapontrendszer (ANR) között.



Az $AA_ETRANS külső nullaponteltolás a gépadat paraméterezéstől függően két módon hat:

1. Az $AA_ETRANS rendszerváltozó az NC/PLC interfész-jellel való aktiválás után közvetlenül hat eltolás-értékként
2. Az $AA_ETRANS rendszerváltozó értéke az NC/PLC interfész-jellel való aktiválás után átvételre kerül a $P:EXTFRAME rendszer frame-be és a $P_EXTFR adattároló frame-be. Ezután az aktív $P_ACTFRAME össz-frame újra ki lesz számítva.

Gépadatok

Az $AA_ETRANS rendszerváltozóval összefüggésben két eljárást kell megkülönböztetni, amelyek a következő gépadattal lesznek kiválasztva:

MD28082 $MC_MM_SYSTEM_FRAME_MASK,bit1 = <érték>

<érték>	Jelentés
0	Működés: $AA_ETRANS[<tengely>] közvetlen írása PLC, HMI vagy NC program által. Az $AA_ETRANS[<tengely>] nullaponteltolás elmozdulás engedélyezése a következő lehetséges elmozdulási mondatban: DB31, ... DBX3.0
1	Működés: Az $P:EXTFRAME aktív rendszer frame és a $P_EXTFR adattároló frame aktiválása Az $AA_ETRANS[<tengely>] nullaponteltolás elmozdulás engedélyezése: DB31, ... DBX3.0 által. Erre a csatornában a következő történik: • minden elmozdulás leállítása a csatornában (a parancs- és a PLC-tengelyeken kívül) • előre-futás állj az azt követő reorganizációval (STOPRE) • durva-eltolás aktív frame $P_EXTFRAME[<tengely>] = $AA_ETRANS[<tengely>] • durva-eltolás adattároló frame $P_EXTFR[<tengely>] = $AA_ETRANS[<tengely>] • $P_ACTFRAME aktív össz-frame újra számítása • eltolás megtétele a programozott tengelyeken • a megszakított mozgások ill. az NC program folytatása

Programozás

- Szintaxis
  `$AA_ETRANS[<tengely>] = <érték>`
- Jelentés

`$AA_ETRANS`:	rendszerváltozó a külső nullaponteltolás közbenső tárolására
`<tengely>`:	csatorna-tengely
`<érték>`:	eltolás-érték

NC/PLC interfész jel

DB31, ... DBX3.0 = 0 → 1 ⇒ $P_EXTFRAME[<tengely>] = $P_EXTFR[<tengely>] = $AA_ETRANS[<tengely>]

6.5 Valósérték beállítása a referencia állapot elvesztésével (PRESETON)

A `PRESETON()` eljárás egy vagy több tengelyre egy új valósértéket állít be gép-koordinátarendszerben (GKR). Ez megfelel egy nullaponteltolásnak a tengely MKR-ben. A tengely ettől nem fog mozogni.

A `PRESETON` egy előrefutás-álljt vált ki szinkronizációval. A valós helyzet a tengelyhez csak nyugalmi állapotban lesz hozzárendelve.

Ha a tengely `PRESETON`-nál nincs hozzárendelve a csatornához, a további eljárás a tengely-csere viselkedés tengely-specifikus beállításától függ.

MD30552 $MA_AUTO_GET_TYPE

Referencia állapot

Egy új valósérték beállításával a gép-koordinátarendszerben a referencia állapotot a géptengelyre törli:

DB31, ... DBX60.4 / .5 = 0 (referált / szinkronizált mérőrendszer 1 / 2)

Javasolt a `PRESETON`-t csak referencia kötelezettség nélküli tengelyeknél használni.

Az eredeti gép-koordinátarendszer helyreállításához a géptengelyek mérőrendszerét, pl. referenciapontra menettel a munkadarabprogramból (`G74`) újra referálni kell.

⚠ VIGYÁZAT

A referencia állapot elvesztése

Egy új valósérték beállításával a gép-koordinátarendszerben `PRESETON`-nal a referencia állapotot a géptengelyre "nem referált / nem szinkronizált"-ra lesz visszaállítva.

Programozás

Szintaxis

```
PRESETON(<tengely_1>, <érték_1> [, <tengely_2>, <érték_2>, ...
<tengely_8>, <érték_8>])
```

Jelentés

`PRESETON`:	valósérték beállítása a referencia állapot elvesztésével	
	előrefutás-állj:	igen
	egyedül a mondat-ban:	igen
`<tengely_x>`:	géptengely név	
	típus:	AXIS
	értéktartomány:	a csatornában definiált géptengely nevek
`<érték_x>`:	géptengely új valósértéke a gép-koordinátarendszerben (MKR) A beadás az aktuálisan érvényes mértékrendszerben történik (hüvelyk / metrikus) Az aktív átmérő-programozás (`DIAMON`) figyelembe lesz véve	
	típus:	REAL

Irodalom

PRESETONS az NC programokban

A `PRESETON` az NC programokban részletes leírás található:

Alap funkciók működési kézikönyv, fejezet "K2: Tengelyek, koordináta-rendszerek, frame-k" > "Koordináta-rendszerek" > "Gép-koordinátarendszer (GKR)" > " Valósérték beállítása a referencia állapot elvesztésével (PRESETON)"

PRESETONS szinkronakciókban

A `PRESETON` a szinkronakciókban részletes leírás található:

Szinkronakciók működési kézikönyv; fejezet: "Részletes leírás" > "Akciók a szinkronakciókban" > "Valósérték beállítása a referencia állapot elvesztésével (PRESETON)"

6.6 Valósérték beállítása a referencia állapot elvesztése nélkül (PRESETONS)

A `PRESETONS()` eljárás egy vagy több tengelyre egy új valósértéket állít be gép-koordinátarendszerben (GKR). Ez megfelel egy nullaponteltolásnak a tengely MKR-ben. A tengely ettől nem fog mozogni.

A `PRESETONS` egy előrefutás-álljt vált ki szinkronizációval. A valós helyzet a tengelyhez csak nyugalmi állapotban lesz hozzárendelve.

Ha a tengely `PRESETONS`-nál nincs hozzárendelve a csatornához, a további eljárás a tengely-csere viselkedés tengely-specifikus beállításától függ.

MD30552 $MA_AUTO_GET_TYPE

Referencia állapot

Egy új valósérték beállításával `PRESETONS`-szal a gép-koordinátarendszerben (MKR) a referencia állapotot a géptengelyre**nem** változik:

Előfeltételek

- **jeladó típusa**
  `PRESETONS` csak az aktív mérőrendszer következő jeladó típusainál lehetséges:
  - MD30240 $MA_ENC_TYPE[<mérőrendszer>] = 0 (szimulált jeladó)
  - MD30240 $MA_ENC_TYPE[<mérőrendszer>] = 1 (nyers-jeladó)
- **referálás mód**
  `PRESETONS` csak az aktív mérőrendszer következő referálás módjainál lehetséges:
  - MD34200 $MA_ENC_REFP_MODE[<mérőrendszer>] = 0 (nem lehetséges referenciapontra menet)
  - MD34200 $MA_ENC_REFP_MODE[<mérőrendszer>] = 1 (növekményes, forgó vagy lineáris mérőrendszerek referálása: null-impulzus a jeladó sávban)

Programozás

Szintaxis

```
PRESETONS(<tengely_1>, <érték_1> [, <tengely_2>, <érték_2>, ...
<tengely_8>, <érték_8>])
```

Jelentés

`PRESETONS`:	valósérték beállítása a referencia állapot elvesztése nélkül	
	előrefutás-állj:	igen
	egyedül a mondat-ban:	igen
`<tengely_x>`:	géptengely név	
	típus:	AXIS
	értéktartomány:	a csatornában definiált géptengely nevek

<érték_x>:	géptengely új aktuális valósértéke a gép-koordinátarendszerben (MKR) A beadás az aktív mértékrendszerben történik (hüvelyk / metrikus) Az aktív átmérő-programozás (DIAMON) figyelembe lesz véve	
	típus:	REAL

Irodalom

PRESETONS az NC programokban

A PRESETONS az NC programokban részletes leírás található:

Alap funkciók működési kézikönyv, fejezet "K2: Tengelyek, koordináta-rendszerek, frame-k" > "Koordináta-rendszerek" > "Gép-koordinátarendszer (GKR)" > " Valósérték beállítása a referencia állapot elvesztése nélkül (PRESETONS)"

PRESETONS szinkronakciókban

A PRESETONS a szinkronakciókban részletes leírás található:

Szinkronakciók működési kézikönyv; fejezet: "Részletes leírás" > "Akciók a szinkronakciókban" > "Valósérték beállítása a referencia állapot elvesztése nélkül (PRESETONS)"

6.7 Frame-számítás 3 térbeli mérési pontból (MEAFRAME)

A MEAFRAME funkció a mérőciklusok támogatásához használható Ez kiszámítja a frame-t három ideális és az azoknak megfelelő három mért pontból.

Ha egy munkadarab a megmunkáláshoz pozicionálásra kerül, akkor a pozíciója a derékszögű gép-koordinátarendszerhez viszonyítva az ideális pozíciójára vonatkozóan általában el van tolva és el is van forgatva. Pontos megmunkáláshoz vagy méréshez vagy egy költséges fizikai beállítás szükséges, vagy a mozgások megváltoztatása a munkadarabprogramban.

Egy frame-t három térbeli pont letapogatásával lehet rögzíteni, amiknek ismertek az ideális pozícióik. A letapogatás egy érintős vagy egy optikai érzékelővel történik, ami speciális, a hordozó lemezen precízen rögzített lyukakat vagy mérő-golyókat érint meg.

Szintaxis

```
MEAFRAME(<ideális pontok>,<mérőpontok>,<minőség>)
```

Jelentés

MEAFRAME:	funkció felhívás		
<ideális pontok>:	2-dim. REAL mező, ami tartalmazza az ideális pontok három koordinátáját		
<mérőpontok>:	2-dim. REALmező, ami tartalmazza a mért pontok három koordinátáját		
<minőség>:	változó, ami információkat ad FRAME számítás minőségéről		
	típus:	VAR REAL	
	érték:	-1	Az ideális pontok megközelítőleg egy egyenesen vannak: a frame-t nem lehetett kiszámítani. A visszaadott FRAME változó egy semleges frame-t tartalmaz.
		-2	A mérési pontok megközelítőleg egy egyenesen vannak: a frame-t nem lehetett kiszámítani. A visszaadott FRAME változó egy semleges frame-t tartalmaz.
		-4	A forgatási mátrix kiszámítása más okból nem sikerül.
		≥ 0.0	A torzítások összege (pontok közötti távolságok), ami szükséges a mért háromszög egy, az ideális háromszöggel kongruens háromszögbe való átvezetéséhez.

Megjegyzés

A mérés minősége

Ahhoz, hogy a mért koordinátákat egy kombinált forgatással/átalakítással hozzá lehessen rendelni az ideális koordinátákhoz, a mérési pontok által kijelölt háromszögnek kongruensnek kell lennie az ideális háromszöghöz. Ezt egy kompenzációs algoritmus hajtja végre, ami az eltérések négyzetösszegét minimalizálja, amik a mért háromszöget átvezetik az ideálisba.

A mérési pontok ténylegesen szükséges torzítása jelzésként szolgálhat a mérés minőségére vonatkozóan és ezért a MEAFRAME kiegészítő változójaként kerül kiadásra.

Megjegyzés

A MEAFRAME által létrehozott frame-t az ADDFRAME funkcióval át lehet transzformálni egy másik frame-be a frame-láncban (lásd a "Láncolás ADDFRAME-mel" példát).

Példák

Példa 1:

Munkadarabprogram 1:

Programkód
...
DEF FRAME CORR_FRAME

Mérési pontok beállítása:

Programkód	Kommentár
DEF REAL IDEAL_POINT[3,3]= SET(10.0,0.0,0.0,0.0,10.0,0.0,0.0,0.0,10.0)	
DEF REAL MEAS_POINT[3,3]= SET(10.1,0.2,-0.2,-0.2,10.2,0.1,-0.2,0.2,9.8)	; Teszthez.
DEF REAL FIT_QUALITY=0	
DEF REAL ROT_FRAME_LIMIT=5	; Az osztás-pozíció max. 5 fokos elfordulását engedi meg.
DEF REAL FIT_QUALITY_LIMIT=3	; Max. 3 mm eltolást enged meg az ideális és a mért háromszög között.
DEF REAL SHOW_MCS_POS1[3]	
DEF REAL SHOW_MCS_POS2[3]	
DEF REAL SHOW_MCS_POS3[3]	

Programkód	Kommentár
N100 G01 G90 F5000	
N110 X0 Y0 Z0	
N200 CORR_FRAME=MEAFRAME(IDEAL_POINT,MEAS_POINT,FIT_QUALITY)	
N230 IF FIT_QUALITY < 0	
SETAL(65000)	
GOTOF NO_FRAME	
ENDIF	
N240 IF FIT_QUALITY > FIT_QUALITY_LIMIT	
SETAL(65010)	
GOTOF NO_FRAME	
ENDIF	

```
Programkód                                          Kommentár
N250 IF CORR_FRAME[X,RT] > ROT_FRAME_LIMIT          ; Az 1. RPY szög határolása.
SETAL(65020)
GOTOF NO_FRAME
ENDIF
N260 IF CORR_FRAME[Y,RT] > ROT_FRAME_LIMIT          ; A 2. RPY-szög határolása.
SETAL(65021)
GOTOF NO_FRAME
ENDIF
N270 IF CORR_FRAME[Z,RT] > ROT_FRAME_LIMIT          ; A 3. RPY szög határolása.
SETAL(65022)
GOTOF NO_FRAME
ENDIF
N300 $P_IFRAME=CORR_FRAME                           ; Letapogató frame aktiválása egy
                                                    beállítható frame-mel.
                                                    ; Frame ellenőrzése a geometriai
                                                    tengelyeknek az ideális pontokra
                                                    való pozicionálásával.
N400 X=IDEAL_POINT[0,0] Y=IDEAL_POINT[0,1]
Z=IDEAL_POINT[0,2]
N410 SHOW_MCS_POS1[0]=$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                                           ; Beállítható frame deaktiválása,
                                                    mert nulla-frame-mel (nincs érték
                                                    beadva) már előre foglalt.
No_FRAME                                            ; Beállítható frame deaktiválása,
                                                    mert nulla-frame-mel (nincs érték
                                                    beadva) már előre foglalt.
M0
M30
```

Példa 2: Frame-k láncolása

MEAFRAME láncolása korrekciókhoz

A MEAFRAME funkció egy korrekció-frame-t ad. Ha ez a korrekció-frame a beállítható $P_UIFR[1] frame-mel lesz láncolva, amelyik a funkció felhívásánál aktív volt (pl. G54), akkor egy beállítható frame-t kapunk további átszámításokra mozgásokhoz vagy megmunkáláshoz.

Láncolás ADDFRAME-mel

Ha ez a korrekció-frame a frame-láncban egy másik helyen kell hasson vagy a beállítható frame előtt még más frame-k aktívak, akkor az ADDFRAME funkciót lehet használni a láncoláshoz egy másik csatorna alap-frame-be vagy egy rendszer-frame-be.

Ennél a frame-kben nem lehet aktív:

- tükrözés MIRROR -ral
- skálázás SCALE-lel

A bemenő paraméterek a parancs- és valósértékekhez a munkadarab-koordináták. A vezérlés alaprendszerében ezeket a koordinátákat mindig metrikusan vagy hüvelykben (G71/G70) és sugár-vonatkoztatású (DIAMOF) méretben kell megadni.

Irodalom:
További információk ADDFRAME-hez lásd:
Alapfunkciók működési kézikönyv; K2: Tengelyek, koordináta-rendszerek, frame-k

6.8 NCU-globális frame-k

Az NCU-globális frame-k minden NCU-hoz az összes csatorna számára csak egyszer vannak. Az NCU-globális frame-k minden csatornából írhatóak és olvashatóak. Az NCU- globális frame-k aktiválása a mindenkori csatornában történik.

A globális frame-k által **csatornatengelyeket és géptengelyeket** eltolásokkal, skálázásokkal és tükrözésekkel lehet befolyásolni.

Geometriai összefüggések és frame-láncok

Globális frame-knél a tengelyek között nem létezik geometriai összefüggés. Ezért nem hajthatók végre geometriai tengely jelölők forgatásai és programozása.

- Globális frame-knél nem használhatók fel forgatások. Egy forgatás beprogramozása az: "18310 csatorna %1 mondat %2 frame: forgatás nem megengedett" vészjelzéssel vissza lesz utasítva.
- Globális frame-k és csatorna-specifikus frame-k láncolata lehetséges. A kapott frame tartalmazza az összes frame-részt, beleértve a forgatásokat is az összes tengely számára. Egy forgatási részekkel rendelkező frame hozzárendelése egy globális frame-hez a "Frame: forgatás nem megengedett" vészjelzéssel vissza lesz utasítva.

NCU-globális frame-k

NCU-globális frame-k $P_NCBFR[n]

Összesen 8 NCU-globális alap-frame-t lehet tervezni:

Ezzel egy időben csatorna-specifikus alap-frame-k is lehetnek.

Globális frame-k egy NCU minden csatornájából írhatóak és olvashatóak. Globális frame-k írásakor a felhasználónak gondoskodnia kell egy csatorna-koordinálásról. Ez pl. várakozás-jelölőkkel (WAITMC) valósítható meg.

Gépgyártó

A globális alap-frame-k számát gépadatban adjuk meg.

Irodalom:
Alapfunkciók működési kézikönyv; Tengelyek, koordináta-rendszerek, frame-k (K2)

NCU-globális beállítható frame-k $P_UIFR[n]

Az összes G500, G54...G599 beállítható frame-t lehet NCU-globálisra vagy csatorna-specifikusra tervezni.

Gépgyártó

Minden beállítható frame áttervezhető globális frame-mé az MD18601 $MM_NUM_GLOBAL_USER_FRAMES gépadat segítségével.

A frame programutasításoknál csatornatengely jelölő és géptengely jelölő használható tengely jelölőként. Geometriai tengely jelölők programozása egy vészjelzéssel vissza lesz utasítva.

6.8.1 Csatorna-specifikus frame-k ($P_CHBFR, $P_UBFR)

A beállítható frame-ket vagy az alap-frame-ket a munkadarabprogramból vagy a BTSS-ről a kezelésben és a PLC-ből lehet írni és olvasni.

A finom eltolás globális frame-knél is lehetséges. Globális frame-k elnyomása ugyanúgy történik, mint a G53, G153, SUPA és G500 csatorna-specifikus frame-knél.

Gépgyártó

Az MD28081 $MC_MM_NUM_BASE_FRAMES gépadattal tervezhető a alap-frame-k száma a csatornában. Az alap konfiguráció így van kialakítva, hogy csatornánként legalább egy alap-frame van. Csatornánként legfeljebb 8 alap-frame lehetséges. A 8 alap-frame-n túl még 8 NCU-globális alap-frame lehet a csatornában.

Csatorna-specifikus frame-k

$P_CHBFR[n]

A $P_CHBFR[n] rendszerváltozó által az alap-frame-k olvashatóak és írhatóak. Egy alap-frame írásánál a láncolt össz alap-frame nem lesz aktiválva, az aktiválás csak egy G500, G54...G599 utasítás végrehajtásával történik meg. A változó elsősorban tárolóként szolgál az alap-frame-re irányuló, HMI vagy PLC általi írási eljárások számára. Ezek a frame-változók az adatmentéssel el lesznek tárolva.

Első alap-frame a csatornában

Az $P_UBFR előre definiált változóba való írás nem aktiválja egyidejűleg a 0 mezőindex-szel rendelkező alap-frame-t, az aktiválás csak egy G500, G54...G599 utasítás végrehajtásával történik meg. A változó a programban is írható vagy olvasható.

$P_UBFR

$P_UBFR megegyezik $P_CHBFR[0]-val. Alap-beállításként mindig van egy alap-frame a csatornában, mivel a rendszerváltozó kompatibilis régebbi kiadásokhoz. Ha nincs csatorna-specifikus alap-frame, akkor írásnál vagy olvasásnál a "Frame: utasítás nem megengedett" vészjelzés jelenik meg.

6.8.2 Csatornában hatásos frame-k

A csatornában hatásos frame-k a munkadarabprogram által ezen frame-k megfelelő rendszerváltozóival lesznek megadva. Ezekhez tartoznak a rendszer-frame-k is. A munkadarabprogramban a rendszerváltozókkal lehet az aktuális rendszer-frame-ket olvasni és írni.

Aktuálisan a csatornában hatásos frame-k

Áttekintés

Aktuális rendszer-frame-k	következőkhöz:
$P_PARTFRAME	TCARR és PAROT
$P_SETFRAME	valósérték beállítás és megkarcolás

$P_EXTFRAME	Külső nullaponteltolás
$P_NCBFRAME[n]	aktuális NCU-globális alap-frame-k
$P_CHBFRAME[n]	aktuális csatorna alap-frame-k
$P_BFRAME	aktuális 1. alap-frame a csatornában
$P_ACTBFRAME	össz alap-frame
$P_CHBFRMASK és $P_NCBFRMASK	össz alap-frame
$P_IFRAME	aktuális beállítható frame
Aktuális rendszer-frame-k	következőkhöz:
$P_TOOLFRAME	TOROT és TOFRAME
$P_WPFRAME	szerszám vonatkoztatási pontok
$P_TRAFRAME	transzformációk
$P_PFRAME	aktuális programozható frame
Aktuális rendszer-frame	következőkhöz:
$P_CYCFRAME	ciklusok
P_ACTFRAME	aktuális össz-frame
FRAME láncolás	aktuális frame az össz alap-frame-ből tevődik össze

$P_NCBFRAME[n] Aktuális NCU-globális alap-frame-k

A $P_NCBFRAME[n] rendszerváltozó által az aktuális globális alap-frame mezőelemek írhatóak és olvashatóak. A kapott össz alap-frame az írási eljárás által a csatornában kerül beszámításra.

A megváltoztatott frame csak abban a csatornában lesz aktív, amelyikben a frame-t programozták. Ha a frame-t egy NCU minden csatornája számára meg akarjuk változtatni, akkor $P_NCBFR[n]-t és $P_NCBFRAME[n]-t egyszerre kell írni. Azután a többi csatornának a frame-t pl. G54-gyel aktiválni kell. Egy alap-frame írásánál az össz alap-frame újra ki lesz számítva.

$P_CHBFRAME[n] Aktuális csatorna alap-frame-k

A $P_CHBFRAME[n] rendszerváltozó által az aktuális csatorna alap-frame mezőelemek írhatóak és olvashatóak. A kapott össz alap-frame az írási eljárás által a csatornában kerül beszámításra. Egy alap-frame írásánál az össz alap-frame újra ki lesz számítva.

$P_BFRAME aktuális 1. alap-frame a csatornában

Az előre definiált $P_BFRAME frame-változóval lehet az alap-frame-t a mezőindex 0-val, amelyik a csatornában érvényes, a munkadarabprogramban olvasni és írni. Az írt bázis- frame azonnal be lesz számítva.

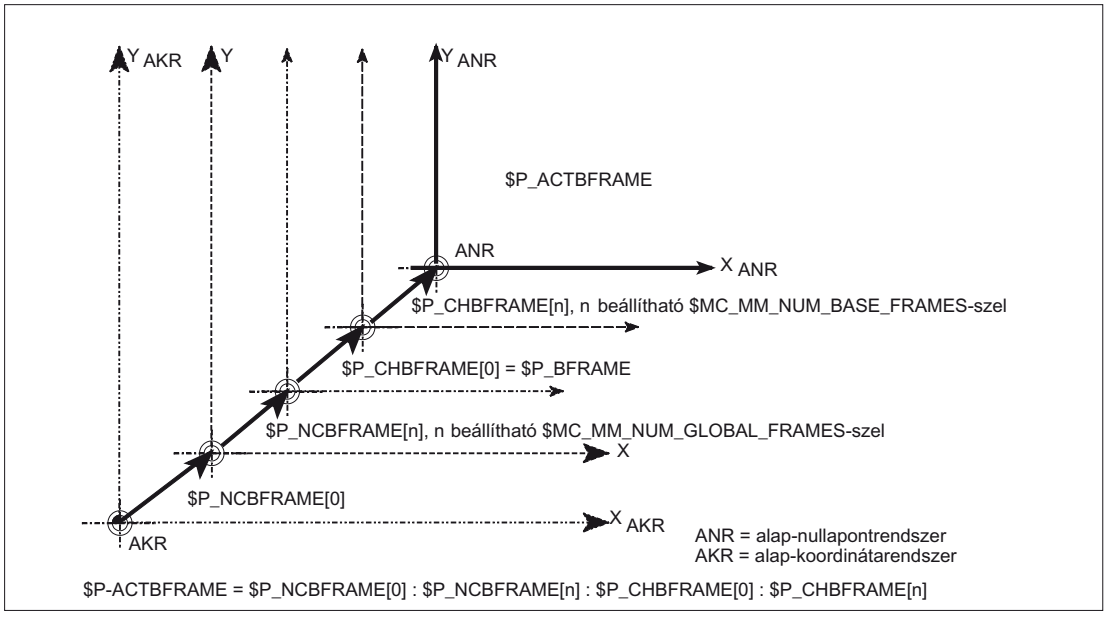
$P_BFRAME megegyezik $P_CHBFRAME[0]-val. A rendszerváltozónak alap-beállításban mindig van egy érvényes értéke. Ha nincs csatorna-specifikus alap-frame, akkor írásnál vagy olvasásnál a "Frame: utasítás nem megengedett" vészjelzés jelenik meg.

$P_ACTBFRAME Össz alap-frame

A $P_ACTFRAME változó a láncolt össz alap-frame-t határozza meg. A változó csak olvasható.

$P_ACTFRAME megfelelője:

$P_NCBFRAME[0] : ... : $P_NCBFRAME[n] : $P_CHBFRAME[0] : ... : $P_CHBFRAME[n].



$P_CHBFRMASK és $P_NCBFRMASK Össz alap-frame

A $P_CHBFRMASK és $P_NCBFRMASK rendszerváltozók által a felhasználó kiválaszthatja, hogy melyik alap-frame-ket szeretné bevonni az "össz" alap-frame kiszámításába. A változók csak a programban programozhatóak és csak BTSS által olvashatóak. A változók értéke bitmaszkként lesz értelmezve és megadja, hogy a $P_ACTFRAME melyik alap-frame mezőeleme lesz a számításba bevonva.

A $P_CHBFRMASK-kal meg lehet adni, hogy melyik csatorna-specifikus alap-frame-k, $P_NCBFRMASK-kal pedig hogy melyik NCU-globális alap-frame-k legyenek beszámítva.

A változók programozásával az össz-alap-frame és az össz-frame újra ki lesz számítva. Reset után és az alapbeállításban a $P_CHBFRMASK und $P_NCBFRMASK értéke a következő:

$P_CHBFRMASK = $MC_CHBFRAME_RESET_MASK

$P_NCBFRMASK = $MC_CHBFRAME_RESET_MASK

Példa:

```
$P_NCBFRMASK = 'H81' ;$P_NCBFRAME[0] : $P_NCBFRAME[7]
$P_CHBFRMASK = 'H11' ;$P_CHBFRAME[0] : $P_CHBFRAME[4]
```

$P_IFRAME Aktuális beállítható frame

Az $P_IFRAME előre definiált frame-változó által az aktuális beállítható frame, ami a csatornában érvényes, írható és olvasható a munkadarabprogramban. Az írt beállítható frame azonnal be lesz számítva.

Az NCU-globális beállítható frame-knél a megváltoztatott frame csak abban a csatornában hat, amelyikben a frame programozva lett. Ha a frame-t egy NCU összes csatornája számára

meg akarjuk változtatni, akkor $P_UIFR[n] és $P_IFRAME egyszerre írandó. Azután a többi csatornában a megfelelő frame-t pl. G54-gyel még aktiválni kell.

$P_PFRAME Aktuális programozható frame

$P_PFRAME egy programozható frame, amely a TRANS/ATRANS, G58/G59, ROT/AROT, SCALE/ASCALE, MIRROR/AMIRROR programozásából ill. a CTRANS, CROT, CMIRROR, CSCALE hozzárendelésből a programozható FRAME-hez adódik.

Aktuális, programozható frame változó, ami létrehozza a kapcsolatot a beállítható nullapontrendszer (BNR) és a munkadarab-koordinátarendszer (MKR) között.

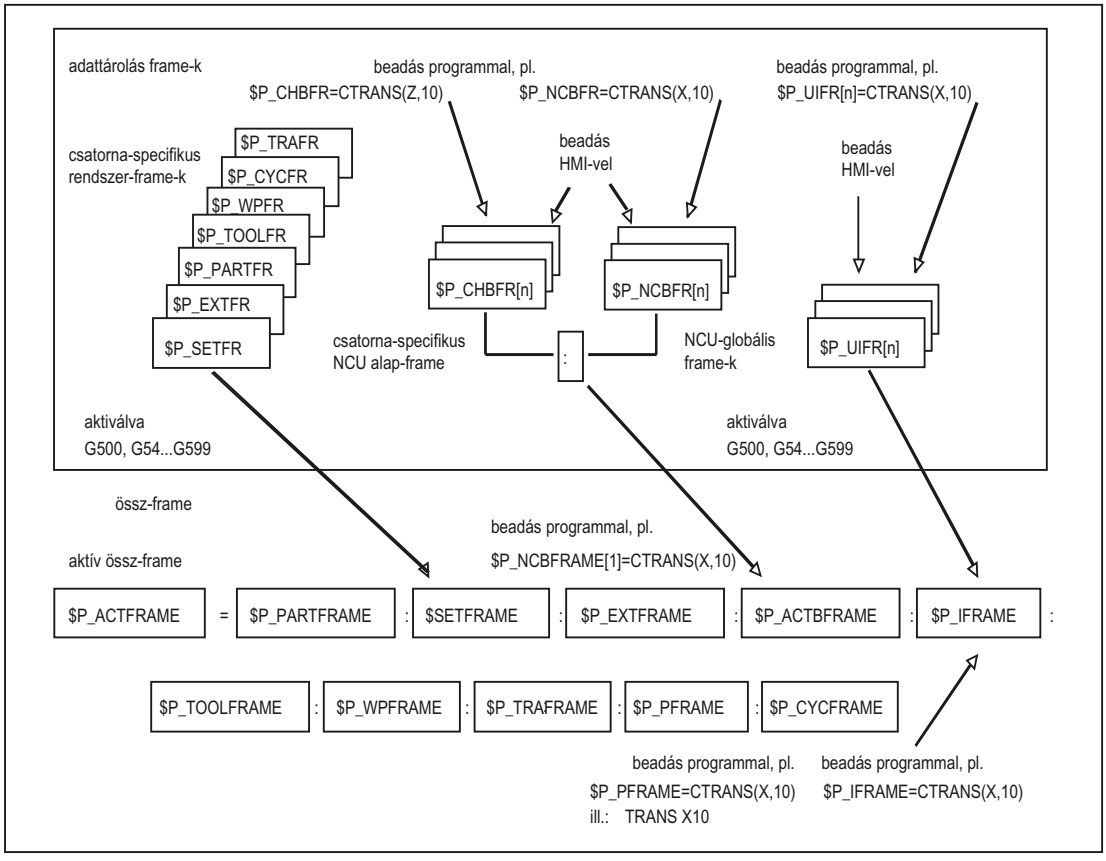
P_ACTFRAME Aktuális össz-frame

A $P_ACTFRAME az aktuális kapott össz-frame az össz alap-frame, az aktuális beállítható frame és a programozható frame láncolatából adódik. Az aktuális frame mindig akkor lesz aktualizálva, ha egy frame-rész megváltozik.

`$P_ACTFRAME` megfelelője

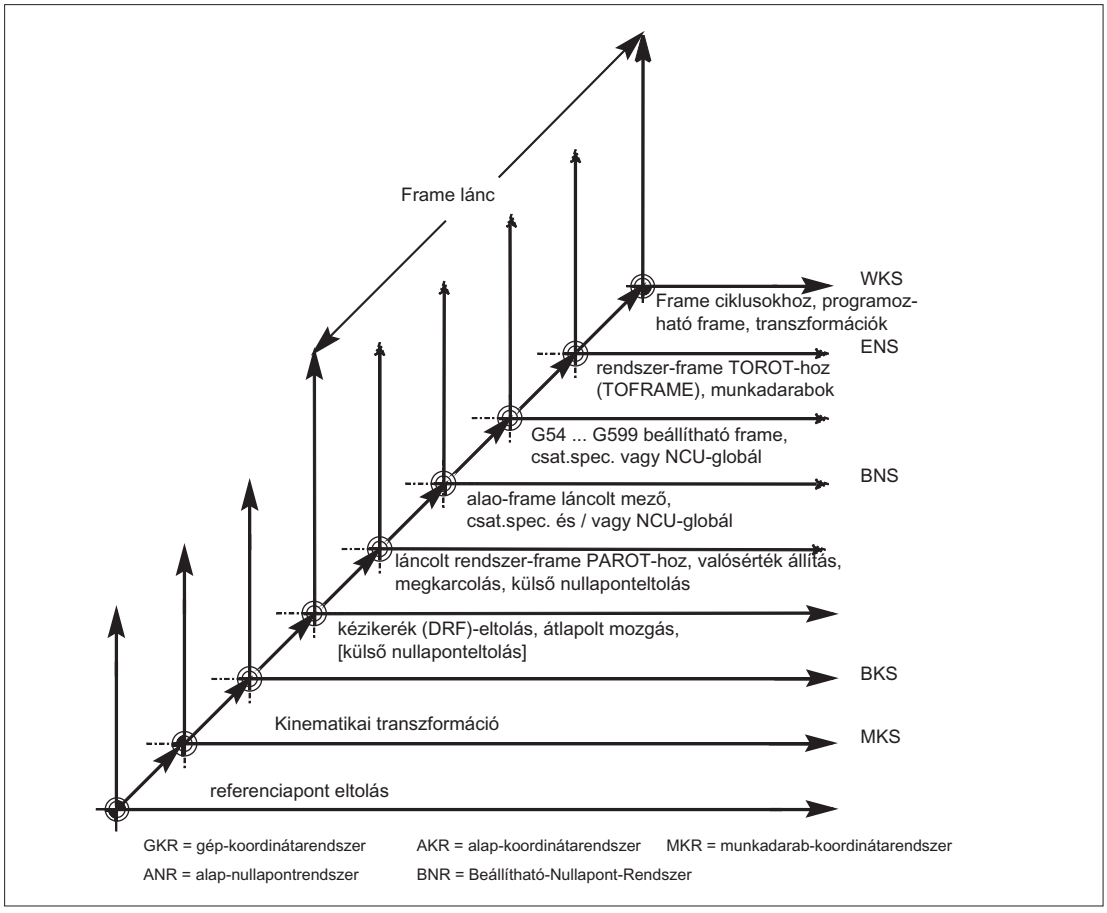
`$P_PARTFRAME : $P_SETFRAME : $P_EXTFRAME : $P_ACTBFRAME : $P_IFRAME :`

`$P_TOOLFRAME : $P_WPFRAME : $P_TRAFRAME : $P_PFRAME : $P_CYCFRAME`



FRAME láncolás

Az aktuális frame az össz alap-frame-ből, a beállítható frame-ből, a rendszer-frame-ből és a programozható frame-ből tevődik össze a fent megadott aktuális össz-frame-nek megfelelően.



Transzformációk 7

7.1 A transzformáció fajták általános programozása

Általános funkció

A vezérlés illesztése a különféle gépkinematikákhoz a megfelelő paraméterekkel programozott transzformáció fajták kiválasztásával történik. Ezekkel a paraméterekkel lehet a kiválasztott transzformációknak megadni a szerszám tájolását a térben és a körtengelyek tájolási mozgását is megfelelően megadni.

A három-, négy- és öttengelyes transzformációknál a programozott helyzetadatok mindig a szerszám csúcsára vonatkoznak, amely a térbeli megmunkálási felületre merőlegesen lesz vezetve. A derékszögű koordináták az alap koordináta-rendszerből át lesznek számítva a gép koordináta-rendszerébe és a geometriai tengelyekre vonatkoznak. Ezek írják le a munkapontot. A virtuális körtengelyek leírják a szerszám tájolását a térben és a programozásuk a TRAORI-val történik.

A kinematikai transzformációknál a pozíciókat lehet a derékszögű koordinátákban programozni. A vezérlés transzformálja a derékszögű koordináta rendszerben TRANSMIT, TRACYL és TRAANG alkalmazásával programozott mozgásokat a valós gépi tengelyek mozgásaira.

Programozás

Három-, négy- és öt-tengelyes transzformáció TRAORI

A meghatározott tájolási transzformáció a TRAORI utasítással és a három lehetséges paraméterrel (trafoszám, tájolási vektor és körtengely-offset) aktiválva.

```
TRAORI(trafoszám, tájolási vektor, körtengely-offset)
```

Kinematikai transzformáció

A kinematikai transzformációk a következők: `TRANSMIT(trafoszám)`

```
TRACYL(munka-átmérő, trafoszám)
```

```
TRAANG(ferdén álló tengely szöge, trafoszám)
```

Aktív transzformációt kikapcsolni

A `TRAFOOF`-fal ki lehet kapcsolni az éppen aktív transzformációt.

Tájolási transzformáció

Három-, négy- és öt-tengelyes transzformáció TRAORI

A térbeli felületek optimális megmunkálásához a gép munkaterében a szerszámgépeknek a három X, Y és Z lineáris tengelyen kívül még szükségük van tengelyekre. A kiegészítő

tengelyek írják le a térbeli tájolást és következőkben tájolótengelyeknek lesznek nevezve. Ezek körtengelyekként állnak rendelkezésre négy, különböző kinematikájú géptípusnál.

1. kéttengelyes billenőfej, pl. kardán szerszámfej egy körtengellyel párhuzamosan egy lineáris tengellyel fix szerszámasztalnál
2. kéttengelyes körasztal, pl. fix billenőfej két tengely körül forgatható szerszámasztallal
3. egytengelyes billenőfej és egytengelyes körasztal, pl. egy forgatható billenőfej forgatott szerszámmal egy tengely körül forgatható szerszámasztallal
4. kéttengelyes billenőfej és egytengelyes körasztal, pl. egy egy tengely körül forgatható szerszámasztalnál egy forgatható billenőfej önmaga körül forgatható szerszámmal

A **3- és 4-tengelyes transzformációk** az 5-tengelyes transzformációk különleges formái és az 5-tengelyes transzformációval azonosan programozhatók.

Az "**Általános 3-/4-/5-/6-tengelyes transzformáció**" lefedi a funkció-terjedelmével a derékszögűen elrendezett körtengelyeket és a kardán marófej transzformációit és mint az összes többi tájolási transzformáció, aktiválható a leírt négy géptípushoz a TRAORI-val. Az általános 5/6-tengelyes transzformációnál a szerszámtájolásnak van egy további harmadik szabadságfoka, amelynél a szerszám tájolásához tetszőlegesen a térben a szerszámot a saját tengelye körül forgatni lehet.

Irodalom: /FB3/ Különleges funkciók működési kézikönyv, 3...5 tengelyes transzformációk (F2)

A szerszámtájolás kinematika-független alaphelyzete

ORIRESET

Ha TRAORI-val egy tájolási transzformáció aktív, akkor az ORIRESET-tel megadható a max. 3 tájolótengely alaphelyzete az opcionális A, B, C paraméterekkel. A programozott paraméterek sorrendjének hozzárendelése a körtengelyekhez a tájolótengelyeknek a transzformációval megadott sorrendjében történik. Az ORIRESET(A, B, C) programozásának hatása hogy a tájolótengelyek lineárisan és szinkronban a pillanatnyi helyzetükből a megadott alaphelyt pozíciójába mennek.

Kinematikai transzformáció

TRANSMIT és TRACYL

A maró-megmunkálás az esztergagépeken a megadott transzformációkból vagy

1. egy homlokoldali megmunkálás eszterga-befogásban TRANSMIT-tel vagy
2. tetszőleges hornyok megmunkálása hengeres testeken TRACYL-lal

programozható.

TRAANG

Ha a fogásvételi tengely, pl. a köszörülés technológiához ferdén is állítható kell legyen, akkor a TRAANG-gal a megadott transzformációra egy paraméterezhető szöget lehet programozni.

Derékszögű PTP-mozgás

A kinematikus transzformációkhoz tartozik a "Derékszögű PTP-mozgás" is, a minél max. 8 különböző csuklóállást lehet STAT=-vel programozni. A pozíciók a derékszögű

koordinátarendszerben lesznek programozva, de a mozgás a gépen a gépi koordinátákban történik.

Irodalom:
/FB2/ Bővítő funkciók működési kézikönyv, Kinematikai transzformációk (M1)

Láncolt transzformációk

Két transzformáció egymás után lehet kapcsolni. Az így csatolt második transzformációnál a tengelyek mozgásai az első transzformációból átvételre kerülnek.

Első transzformációként lehetségesek:

- tájolási transzformáció TRAORI
- polár transzformáció TRANSMIT
- henger transzformáció TRACYL
- ferde tengely transzformáció TRAANG

A második transzformáció TRAANG ferde tengely kell legyen.

7.1.1 Tájolási mozgások a transzformációknál

Út-elmozdulások és tájolási mozgások

A programozható tájolások út-elmozdulásai alapvetően a géptípustól függenek. A három-, négy- és öttengelyes transzformációnál TRAORI-val a forgó tengelyek vagy a billenthető lineáris tengelyek írják le a szerszám tájolási mozgását.

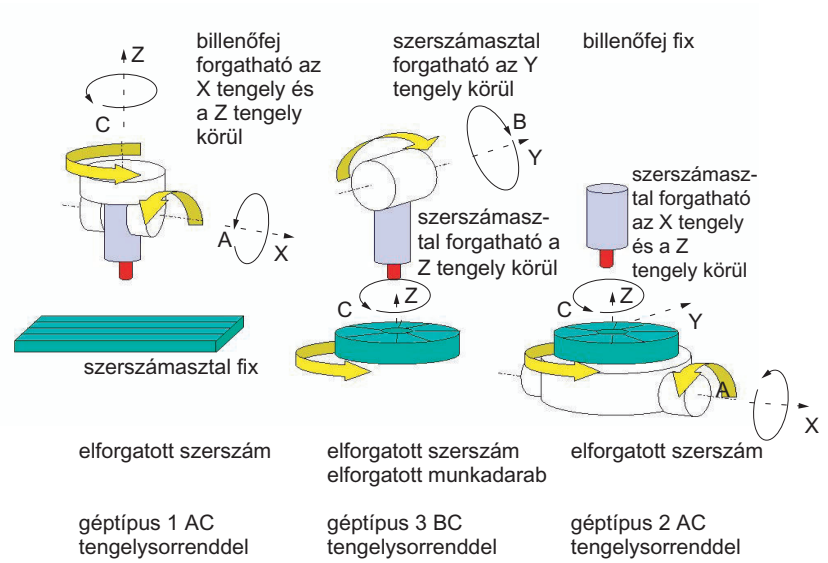
A tájolásban résztvevő körtengelyek pozícióinak változásai kiegyenlítő mozgásokat okoznak a többi géptengelyen. A szerszámcsúcs helyzete ennek során változatlan marad.

A szerszám tájolási mozgásai a virtuális A…, B…, C… körtengely-jelölőkkel az alkalmazástól függően az Euler- ill. RPY-szögek vagy irány- ill. felület-normálvektorok, egy kúp forgástengelye vagy egy kúppalást-felületen közbenső tájolás normált vektorának megadásával programozhatók.

A kinematikus transzformációknál TRANSMIT, TRACYL és TRAANG esetén a vezérlés a derékszögű út-elmozdulásait transzformálja a valós géptengelyek út-elmozdulásaira.

Gép-kinematika három-, négy- és öt-tengelyes TRAORI transzformációnál

A szerszám vagy a szerszámasztal max. két körtengellyel forgatható lehet. Lehetséges az egytengelyes billenőfej és az egytengelyes körasztal kombinációja is.



Géptípus	Tájolás programozása
géptípus 1 és 2 három-tengelyes transzformáció	Szerszámtájolás programozása csak a forgó tengelyre merőleges síkban. Van **két** egyenes tengely (lineáris tengely) és **egy** forgó tengely (körtengely).
géptípus 1 és 2 négy-tengelyes transzformáció	Szerszámtájolás programozása csak a forgó tengelyre merőleges síkban. Van **három** egyenes tengely (lineáris tengely) és **egy** forgó tengely (körtengely).
géptípus 3 öt-tengelyes transzformáció egy-tengelyes billenőfej és egy-tengelyes körasztal	Tájolási transzformáció programozása. Kinematika **három** lineáris tengellyel és **két** derékszögű körtengellyel. A körtengelyek a háromból két lineáris tengellyel párhuzamosak. Az első körtengelyt két derékszögű lineáris tengely mozgatja. Ez forgatja a harmadik lineáris tengelyt a szerszámmal. A második körtengely forgatja a munkadarabot.

Általános 5/6-tengelyes transzformáció

Géptípus	Tájolási transzformáció programozása.
géptípus 4 általános öt-/hattengelyes transzformáció két-tengelyes billenőfej önmaga körül forgatható szerszámmal és egytengelyes körasztal	Tájolási transzformáció programozása. Kinematika **három** lineáris tengellyel és **három** derékszögű körtengellyel. A körtengelyek a háromból két lineáris tengellyel párhuzamosak. Az első körtengelyt két derékszögű lineáris tengely mozgatja. Ez forgatja a harmadik lineáris tengelyt a szerszámmal. A második körtengely forgatja a munkadarabot. A szerszám alap-tájolását egy önmaga körüli kiegészítő forgatással THETA forgásszöggel programozható.

Az "általános három-/négy-/öt-/hattengelyes transzformáció" felhívásánál kiegészítőleg át lehet adni a szerszám alap-tájolását. Ekkor már nem érvényesek a körtengelyek irányaival

kapcsolatos korlátozások. Ha a körtengelyek nem pontosan merőlegesek egymásra vagy a körtengelyek nem pontosan párhuzamosak a lineáris tengelyekkel, az "általános öt-/hattengelyes transzformáció" jobb eredményt adhat a szerszámtájolásra.

TRANSMIT, TRACYL és TRAANG kinematikus transzformációk

Az esztergagépen történő maró-megmunkálásnál vagy egy ferdén állítható tengelynél köszörülésnél a transzformációtól függően alapesetben a következő tengely-hozzárendelések érvényesek:

TRANSMIT	polár transzformáció aktiválása
homlokoldali megmunkálás eszterga-felfogásban	egy körtengely egy fogásvételi tengely merőlegesen a forgótengelyre egy hossztengely párhuzamosan a forgótengellyel

TRACYL	hengerköpeny transzformáció aktiválása
tetszőleges hornyok megmunkálása egy hengeres testen	egy körtengely egy fogásvételi tengely merőlegesen a forgótengelyre egy hossztengely párhuzamosan a forgótengellyel

TRAANG	ferde tengely transzformáció aktiválása
Megmunkálás ferde fogásvételi tengellyel	egy körtengely egy fogásvételi tengely paraméterezhető szöggel egy hossztengely párhuzamosan a forgótengellyel

Derékszögű PTP-mozgás

A gép mozgása gépkoordinátákban történik és programozása a következő:

TRAORI	transzformáció aktiválása
PTP pontról pontra menet	pozíciókat a derékszögű koordináta-rendszereben (GKR) felvenni
CP	derékszögű tengelyek pályamozgása (AKR)
STAT	csuklók állása függ a transzformációtól
TU	mely szöggel mozogjanak a tengelyek a legrövidebb úton

PTP-mozgás általános 5/6-tengelyes transzformációnál

A gép mozgása gépi koordinátákban történik és a szerszám tájolása programozható körasztal pozíciókkal és a kinematika független Euler vektorral ill. RPY-szöggel is vagy az irányvektorokkal.

Ennek során lehetséges körtengely-interpoláció, vektor-interpoláció nagykör-interpolációval vagy a tájolási vektor interpolációja egy kúppalást-felületen.

Példa három...öttengelyes transzformációra egy kardán marófejnél

A szerszámgépnek legalább 5 tengelye van, ebből

- három egyenes tengely egyenesvonalú mozgásokhoz, amelyek a munkapontot a munkatérben bármely tetszőleges helyre mozgatják.
- két forgó billenőtengely, amelyek egy beállítható szöggel (többnyire 45 fok) vannak elrendezve, lehetővé teszi a szerszám tájolását a térben, ami 45 foknál egy fél-kúpra korlátozódik.

7.1.2 TRAORI tájolási transzformáció áttekintése

Lehetséges programozási módok TRAORI-val összefüggésben

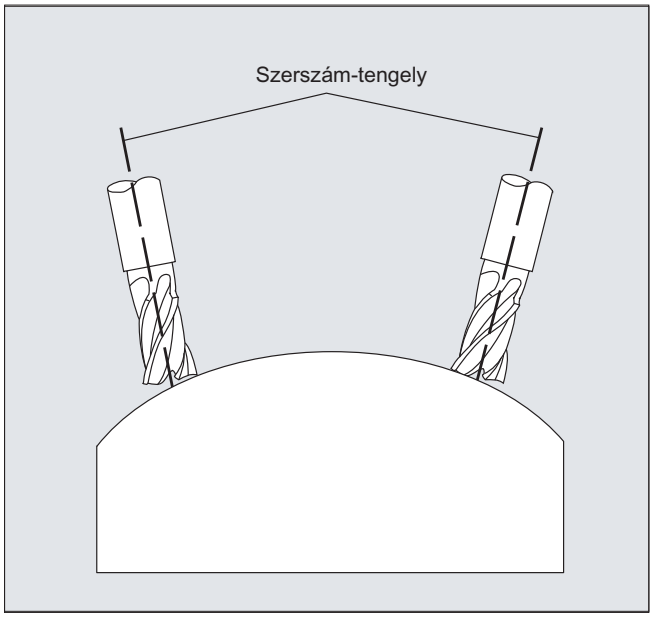
Géptípus	programozás aktív TRAORI transzformációnál
géptípus 1, 2 vagy 3 kéttengelyes billenőfej vagy kéttengelyes körasztal vagy egy egytengelyes billenőfej és egy egytengelyes körasztal kombinációja	A tájolótengelyek tengely-sorrendje és a szerszám tájolási iránya vagy **gépre-vonatkoztatottan** megadható gépadatokkal a gépkinematikától függően vagy **munkadarabra-vonatkoztatottan** programozható tájolással ua gépkinematikától függetlenül. A tájolótengelyek sorrendje a vonatkoztatási rendszerben van programozva: -ORIMKS vonatkoztatási rendszer= gép-koordinátarendszer - ORIWKS vonatkoztatási rendszer= munkadarab-koordinátarendszer Az alaphelyzet az ORIWKS. A tájolótengelyek programozása a következőkkel: A, B, C géptengely-pozíciókkal közvetlenül A2, B2, C2 szögprogramozás a virtuális tengelyeken - ORIEULER Euler-szöggel (alaphelyzet) - ORIRPY RPY-szöggel - ORIVIRT1 virtuális tájolótengely 1. definícióval - ORIVIRT2 virtuális tájolótengely 2. definícióval különféle interpolációs módokkal: **lineáris interpoláció** - ORIAXES tájolótengelyekre vagy géptengelyekre **nagykör-interpoláció** (tájolási vektor interpolációja) - ORIVECT tájolótengelyekre A tájolótengelyek programozása az irány-/felület-normálvektor A3, B3, C3 vektorkomponenseinek megadásával Az eredő szerszámtájolás programozása A4, B4, C4 a felület-normálvektorhoz a mondat elején A5, B5, C5 a felület-normálvektorhoz a mondat végén LEAD előresietés szöge szerszámtájoláshoz TILT oldalszög szerszámtájoláshoz
	Tájolási vektor interpolációja egy kúppalást-felületen Tájolás-változások egy a **térben tetszőlegesen** levő kúppalást-felületen interpolációval: - ORIPLANE a síkban (nagykör-interpoláció) - ORICONCW egy kúppalást-felületen órajárás irányában - ORICONCCW egy kúppalást-felületen órajárás irányával szemben A6, B6, C6 irányvektorok (a kúp forgástengelye) - OICONIO interpoláció egy kúppalást-felületen: A7, B7, C7 közbenső vektorral (kezdő- és végtájolás) vagy - ORICONTO egy kúppalást-felületen érintőleges átmenet Tájolás-változások **egy pályára vonatkoztatva** az - ORICURVE-val mozgás megadásával két érintőponton át PO[XH]=(xe, x2, x3, x4, x5) max. 5. fokú tájolás-polinom PO[YH]=(ye, y2, y3, y4, y5) max. 5. fokú tájolás-polinom PO[ZH]=(ze, z2, z3, z4, z5) max. 5. fokú tájolás-polinom - tájolás lefutás ORIPATHS simítása A8, B8, C8 szerszám áttájolási fázis megfelel: Szerszám iránya és úthossza leemelési mozgásnál

Géptípus	programozás aktív TRAORI transzformációnál
géptípusok 1 és 3 További géptípusok a szerszám önmaga körüli kiegészítő forgatásával egy 3. körtengelyt igényelnek Orientierungstransformation, wie z.B. generische 6-Achs-Transformation. Tájolási vektor forgatása	Szerszámtájolás **forgatásainak** programozása LEAD előre- sietési szöggel relatív a felület-normálvektorhoz PO[PHI] P egy max. 5. fokú polinom programozása TILT oldalszög forgatás a pályaérintő körül (Z-irány) PO[PSI] egy max. 5. fokú polinom programozása THETA forgásszög (forgatás a szerszámirány körül Z-ben) THETA= érték, ami a mondat végén lesz elérve THETA=AC(...) mondatonként abszolút méretmegadásra átkapcsolni THETA=IC(...) mondatonként relatív méretmegadásra átkapcsolni THETA=Θ_e programozott szög G90/G91 interpoláció PO[THT]=(..) egy max. 5. fokú polinom programozása forgásvektor programozása - ORIROTA abszolút forgatás - ORIROTR relatív forgatásvektor - ORIROTT érintőleges forgatásvektor
pályához viszonyított tájolás a pályához viszonyított tájolásváltozásokhoz vagy a forgásvektor forgatása érintőlegesen a pályához	Tájolás-változások a **pályához viszonyítva** az - ORIPATH szerszámtájolás a pályához viszonyítva - ORIPATHS kiegészítőleg egy törésnél a tájolás lefutásában Forgatási vektor programozása - ORIROTC érintőleges forgásvektor, forgatás a pályaérintőhöz

7.2 Három-, négy-, öt-tengelyes transzformációk (TRAORI)

7.2.1 Általános összefüggések kardán szerszámfejnél

Ahhoz, hogy a térben görbült felületek megmunkálásánál optimális vágási feltételeket érjünk el, a szerszám ráállási szögének változtathatónak kell lennie.

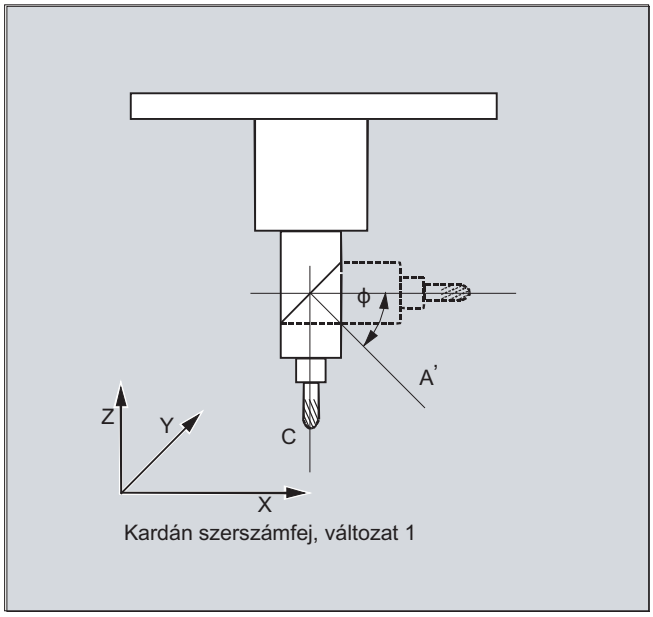


A tengelyadatokban kerül megadásra, hogy ez milyen gépkonstrukcióval érhető el.

5-tengelyes transzformáció

Kardán szerszámfej

Itt három lineáris tengely (X, Y, Z) és két tájoló-tengely adja meg a szerszám beállítási szögét és munkapontját. A két tájoló-tengelyből az egyik ferde-tengelyként van megadva, itt a példában A' - sok esetben 45°-os elrendezésként.

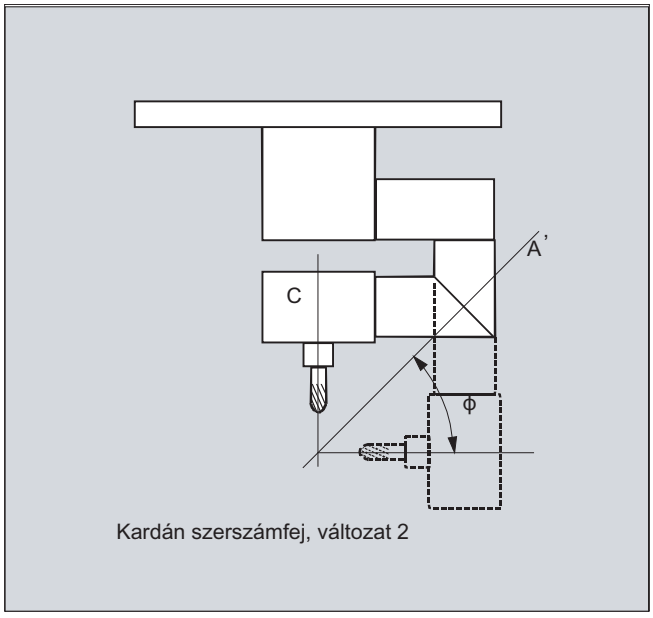


Kardán szerszámfej, változat 1

Az itt bemutatott példákban a CA gép-kinematika kardán szerszámfej példáján láthatjuk az elrendezéseket!

Gépgyártó

A szerszám tájolási iránya és a tájoló-tengelyek sorrendje a gép-kinematikától függően gépadatokkal kerül beállításra.



Kardán szerszámfej, változat 2

Ebben a példában az A' φ szöggel áll az X-tengelyhez viszonyítva.

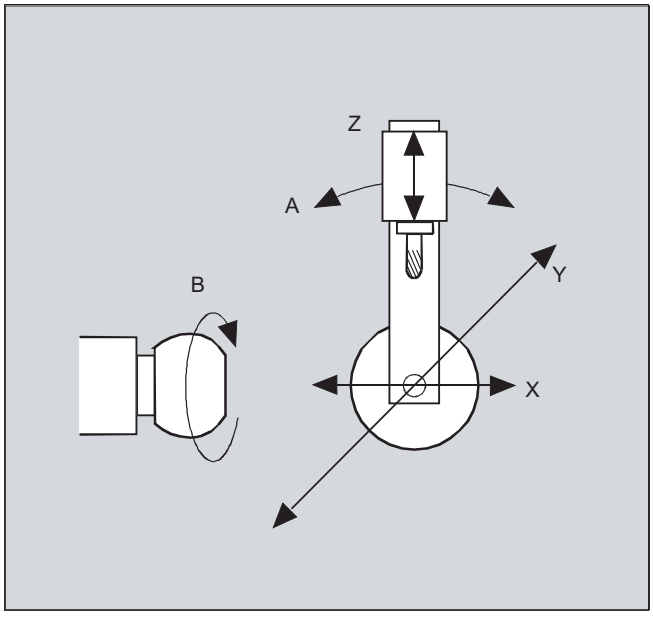
Általában az alábbi lehetséges összefüggések érvényesek:

A' φ szöget zár be az	X tengellyel
B' φ szöget zár be az	Y tengellyel
C' φ szöget zár be a	Z tengellyel

A ϕ szög gépadatokkal állítható be a 0°-tól +89°-ig terjedő tartományban.

Billenthető lineáris tengellyel

Itt egy mozgó munkadarabbal és mozgó szerszámmal történő elrendezésről van szó. A kinematika három lineáris tengelyből (X, Y, Z) és két derékszögűen elrendezett forgótengelyből áll össze. Az első körtengelyt pl. két lineáris tengely mozgatja egy keresztszán által, a szerszám párhuzamosan áll a harmadik lineáris tengellyel. A második körtengely forgatja a munkadarabot. A harmadik lineáris tengely (billenőtengely) a keresztszán síkjában található.



A forgótengelyek sorrendje és a szerszám tájolási iránya a gépkinematikától függ, a gépadatok által kerül beállításra.

Az alábbi lehetséges összefüggések érvényesek:

Tengelyek:	Tengelysorrendek:
1. körtengely	A A B B C C
2. körtengely	B C A C A B
billentett lineáris tengely	Z Y Z X Y X

További magyarázatok a szerszám tájolási irányának konfigurálható tengely-sorrendjéhez:

Irodalom: /FB3/ Különleges funkciók működési kézikönyv, 3...5 tengelyes transzformációk (F2), Kardán marófej fejezet, "Paraméterezés".

7.2.2 Három-, négy- és öt-tengelyes transzformáció (TRAORI)

A felhasználó két ill. három egyenes tengelyt és egy forgótengelyt tervezhet. A transzformációk abból indulnak ki, hogy a forgótengely merőlegesen áll a tájolási síkra.

A szerszám tájolása csak abban a síkban lehetséges, amely merőleges a forgótengelyre. A transzformáció azokat a géptípusokat támogatja, amelyek mozgatható szerszámmal és mozgatható munkadarabbal rendelkeznek.

A három- és négy-tengelyes transzformációk tervezése és programozása analóg az öt-tengelyes transzformációéval.

Irodalom:
Különleges funkciók működési kézikönyv; Többtengelyes transzformációk (F2)

Szintaxis

```
TRAORI(<n>)
TRAORI(<n>,<X>,<Y>,<Z>,<A>,<B>)
TRAFOOF
```

Jelentés

TRAORI:	aktiválja az első megadott tájolási transzformációt	
TRAORI(<n>):	aktiválja az n-nel megadott tájolási transzformációt	
<n>:	transzformáció száma	
	érték:	1 vagy 2
	Példa: TRAORI(1) aktiválja az 1-es tájolási transzformációt	
<X>,<Y>,<Z>:	tájolási vektor komponensei, amerre a szerszám mutat	
<A>,<B>:	programozható offset körtengelyekre	
TRAFOOF:	transzformáció kikapcsolása	

Szerszámtájolás

A szerszám választott tájolási irányától függően, az aktív munkasíkot (G17, G18, G19) úgy kell beállítni az NC programban, hogy a szerszámhossz-korrekció a szerszámtájolás irányába hasson.

Megjegyzés

A transzformáció bekapcsolása után a pozícióadatok (X, Y, Z) mindig a szerszám csúcsára vonatkoznak. A transzformációban résztvevő körtengelyek pozíciójának változása a többi géptengely olyan kiegyenlítő mozgásához vezet, hogy a szerszámcsúcs pozíciója változatlan marad.

A tájolási transzformáció mindig a szerszám csúcsától a szerszám befogása felé irányul.

Offset a tájoló-tengelyekhez

A tájolási transzformáció aktiválásánál a tájoló-tengelyekre lehet közvetlenül programozni egy kiegészítő offsetet.

Paraméterek elhagyhatók, ha a programozásnál a megfelelő sorrend be lesz tartva.

Példa:

`TRAORI(, , , ,A,B)` ; ha csak egyetlen offsetet kell beadni

A közvetlen programozás alternatívájaként a kiegészítő offset a tájoló-tengelyekhez automatikusan is átvehető a pillanatnyilag aktív nullaponteltolásból. Az átvétel gépadattal állítható be.

Példák

```
TRAORI(1,0,0,1)          ; A szerszám alap-tájolása a Z irányba mutat
TRAORI(1,0,1,0)          ; A szerszám alap-tájolása a Z irányba mutat
TRAORI(1,0,1,1)          ; A szerszám alap-tájolása a Y/Z irányba mutat (megfelel a
                         -45° állásnak)
```

7.2.3 Tájolás programozás és alaphelyzet változatok (OTIRESET)

Szerszámtájolás tájolás programozása TRAORI-nál

A TRAORI programozható tájolás transzformációval kapcsolatban az X, Y, Z lineáris tengelyek mellett lehet programozni A.., B..., C... körtengely-jelölőkkel tengelypozíciókat vagy virtuális tengelyeket szögekkel vagy vektor-komponensekkel. A tájolási- és géptengelyekre különféle interpolációs módok lehetségesek. Attól függetlenül, hogy éppen melyik tájolási-polinom PO[szög] és tengely-polinom PO[tengely] aktív, több különböző polinomfajta, mint pl. G1, G2, G3, CIP vagy POLY programozható.

A szerszám tájolásának változását lehet tájolási vektorokkal is programozni. Itt minden mondat végső tájolása vagy a vektor közvetlen tájolásával vagy a körtengely-pozíciók programozásával történhet.

Tájolás programozás változatok három-...öttengelyes transzformációnál

A tájolás programozás következő változatai kölcsönösen kizárják egymást.

`A, B, C`	körtengely-pozíciók közvetlen megadása
`A2, B2, C2`	virtuális tengelyek szög-programozása Euler-szöggel vagy RPY-szöggel
`A3, B3, C3`	vektor-komponensek megadása
`LEAD, TILT`	előre-sietés szög és oldalszög megadása a pályára és a felületre vonatkoztatva
`A4, B4, C4` `A5, B5, C5`	felület-normálvektor a mondat elején a mondat végén

A6, B6, C6 A7, B7, C7	tájolás-vektor interpolációja egy kúppalást-felületen
A8, B8, C8	szerszám áttájolása, leemelési mozgás iránya és úthossza

Szerszámtájolás alaphelyzetét felvenni (ORIRESET)

Az ORIRESET(...)-tel a mindenkori gépkinematika tájolási tengelyei lineárisan és szinkronban az aktuális pozíciójukból a programozott alaphelyzet pozícióba fognak mozogni. Ha egy tengelyre nincs alaphelyzet-pozíció programozva, akkor a hozzátartozó $MC_TRAFO5_ROT_AX_OFFSET_1/2 gépadatban levő pozíció lesz alkalmazva.

A körtengelyek aktív frame-jei nem lesznek figyelembe véve.

Példák CA gép-kinematikára (C, A csatornanevek)

Utasítások	Leírás
ORIRESET(90, 45)	C tengely: 90° A tengely: 45°
ORIRESET(, 30)	C tengely: $MC_TRAFO5_ROT_AX_OFFSET_1/2[0] A tengely: 30°
ORIRESET( )	C tengely: $MC_TRAFO5_ROT_AX_OFFSET_1/2[0] A tengely: $MC_TRAFO5_ROT_AX_OFFSET_1/2[1]

Példák CAC gép-kinematikára (C, A, B) csatornanevek

Utasítások	Leírás
ORIRESET(90, 45, 90)	C tengely: 90° A tengely: 45° B tengely: 90°
ORIRESET( )	C tengely: $MC_TRAFO5_ROT_AX_OFFSET_1/2[0] A tengely: $MC_TRAFO5_ROT_AX_OFFSET_1/2[1] B tengely: $MC_TRAFO5_ROT_AX_OFFSET_1/2[2]

Megjegyzés

A rámenet a szerszámtájolás alaphelyzetre mit ORIRESET(...)-tel csak aktív tájolási transzformációnál TRAORI(...) történhet.

LEAD, TILT és THETA forgatások programozása

LEAD előresietés-szög és TILT oldalszög

A szerszámtájolások forgatásai a három...öt-tengelyes transzformációknál a LEAD előresietés-szöggel és a TILT oldalszöggel lesznek programozva.

THETA forgásszög

Egy transzformációnál egy **harmadik körtengellyel** úgy a tájoláshoz vektor-komponensekkel, mint a LEAD, TILT szögekkel programozáshoz a szerszám önmaga körüli forgatását a THETA forgásszöggel lehet programozni.

7.2.4 Szerszámtájolás programozása (A..., B..., C..., LEAD, TILT)

A szerszám tájolásának programozására a következő lehetőségek vannak:

1. A körtengelyek mozgásának közvetlen programozása. A tájolás változtatása mindig az alap- ill. gép-koordinátarendszerben történik. A tájolótengelyek szinkrontengelyekként mozognak.
2. Euler-szögekben vagy RPY-szögekben történő programozás `A2, B2, C2` szögmegadással.
3. irányvektor programozása `A3, B3, C3` által. Az irányvektor a szerszámcsúcstól a szerszámfelvevő irányába mutat.
4. Felület-normálvektor programozása a mondat elején `A4, B4, C4` által és a mondat végén `A5, B5, C5` által (homlokmarás).
5. Programozás `LEAD` előresietés-szög és `TILT` oldalszög által.
6. A kúp forgástengelyének programozása normálvektorral `A6, B6, C6` által vagy közbenső-tájolással a kúppalást-felületen `A7, B7, C7,` által, lásd a "Tájolás programozása egy kúppalást-felület mentén (ORIPLANE, ORICONxx)"
7. Áttájolás programozása, a szerszám iránya és úthossza a leemelési mozgás alatt `A8, B8, C8,` által, lásd a "Tájolás lefutásának simítása (ORIPATHS A8=, B8=, C8=)" fejezetben.

Megjegyzés

A tájolási programozás minden esetben csak akkor engedhető meg, amikor egy tájolási transzformáció be van kapcsolva.

Előny: ezek a programok minden gép-kinematikára átvihetők.

Szerszámtájolás definíciója G utasítással

Megjegyzés

Gépgyártó

Gépadattal lehetséges az átkapcsolás Euler- és RPY-szögek között. Megfelelő gépadat beállításoknál az átkapcsolás a G-csoport 50 aktív G utasításától függően és függetlenül is lehetséges. A következő beállítási lehetőségekből lehet választani:

1. Ha a tájolótengelyek definíciójára és a tájolási szög G utasítással definiálására szolgáló gépadatok egyaránt nullára vannak állítva:
   Az `A2, B2, C2` által programozott szögek **a gépadattól függően** a tájolás programozás szögdefiníciójában vagy Euler- vagy RPY-szögként lesznek értelmezve.
2. Ha a tájolótengelyek G utasítással történő definíciójának gépadata egyre van beállítva, átkapcsolás történik a G-csoport 50 aktív G utasításától
   függően:
   Az `A2, B2, C2` által programozott szögek az `ORIEULER, ORIRPY, ORIVIRT1, ORIVIRT2, ORIAXPOS` és `ORIPY2` aktív G utasítások egyikének megfelelően lesznek értelmezve. A tájolótengelyekkel programozott értékek a G-csoport 50 aktív G uatsításának megfelelően szintén tájolási szögként lesznek értelmezve.
3. Ha a tájolási szög G utasítással definiálására szolgáló gépadat egyre és a tájolótengelyek G utasítással történő definíciójának gépadata nullára van állítva, átkapcsolás történik a G-csoport 50 aktív G utasításától
   független:
   Az `A2, B2, C2` által programozott szögek az `ORIEULER, ORIRPY, ORIVIRT1, ORIVIRT2 ORIAXPOS` és `ORIPY2` aktív G utasítások egyikének megfelelően lesznek értelmezve. A tájolótengelyekkel programozott értékek a G-csoport 50 aktív G utasításától függetlenül mindig körtengely-pozíciókként lesznek értelmezve.

Szintaxis

Körtengely pozíciók

```
G1 X<érték> Y<érték> Z<érték> A<érték> B<érték> C<érték>
```

Euler-szög

```
G1 X<érték> Y<érték> Z<érték> A2=<érték> B2=<érték> C2=<érték>
```

Irányvektor

```
G1 X<érték> Y<érték> Z<érték> A3=<érték> B3=<érték> C3=<érték>
```

Felület-normálvektor a mondat elején

```
G1 X<érték> Y<érték> Z<érték> A4=<érték> B4=<érték> C4=<érték>
```

Felület-normálvektor a mondat végén

```
G1 X<érték> Y<érték> Z<érték> A5=<érték> B5=<érték> C5=<érték>
```

Előre-sietési szög

```
LEAD=<érték>
```

Oldalszög

`TILT=<érték>`

Jelentés

`G1:`	egyenes interpoláció
`X, Y, Z:`	lineáris tengely pozíciók
`A, B, C:`	körtengely pozíciók
`A2=, B2=, C2=:`	szög programozás (Euler- vagy RPY-szöggel)
`A3=, B3=, C3=:`	irányvektor az MKR X, Y és Z koordinátáiban
`A4=, B4=, C4=:`	felület-normálvektor a mondat elején az MKR X, Y és Z koordinátáiban
`A5=, B5=, C5=:`	felület-normálvektor a mondat végén az MKR X, Y és Z koordinátáiban
`LEAD= :`	előresietés-szög [1)]
`TILT= :`	oldalszög [1)]
1) A szög-adatok értelmezése függ a beállításoktól a MD21094 $MC_ORIPATH_MODE-ban.	

További információk

Az 5-tengelyes programokat általában a CAD/CAM rendszerekkel hozzák létre és nem a vezérlésbe kerülnek beadásra. Ezért a következő felvilágosítások elsősorban a posztprocesszorok programozói számára lényegesek.

A tájolás programozásához a következő utasítások állnak rendelkezésre:

Utasítás	Jelentés
`ORIEULER:`	Euler-szög ZX'Z" forgatási sorrenddel
`ORIRPY:`	RPY-szög XY'Z" forgatási sorrenddel
`ORIRPY2:`	RPY-szög ZY'X" forgatási sorrenddel
`ORIVIRT1:`	virtuális tájolótengelyek szabadon definiálható forgatási sorrenddel: MD21120 $MC_ORIAX_TURN_TAB_1
`ORIVIRT2:`	virtuális tájolótengelyek szabadon definiálható forgatási sorrenddel: MD21130 $MC_ORIAX_TURN_TAB_2
`ORIAXPOS:`	virtuális tájolótengelyek körtengely-pozíciókkal

Megjegyzés

A gépgyártó gépadatokkal különféle változatokat definiálhat. Vegyük ehhez figyelembe a gépgyártó tájékoztatásait.

Programozás Euler-szögekben `ORIEULER`, forgatási sorrend Z X' Z"

Az `ORIEULER` tájolási programozásnál az `A2`, `B2`, `C2`-vel megadott értékek Euler-szögként (fokban) lesznek értelmezve.

Az új tájolási vektor az eredeti tájolási vektor következő három forgatásából adódik

1. `A2` körtengellyel az Y koordináta-tengely körül
2. `B2` körtengellyel az új X' koordináta-tengely körül
3. `C2` körtengellyel az új Z'' koordináta-tengely körül

Ebben az esetben `C2` értékének (forgatás az új Ó-tengely körül) nincsen jelentősége és nem kell programozni.

Programozás RPY-szögekben `ORIRPY`, forgatási sorrend X Y' Z''

Az `ORIRPY` tájolási programozásnál az `A2`, `B2`, `C2`-vel megadott értékek RPY-szögként (fokban) lesznek értelmezve az X Y' Z'' forgatási sorrenddel.

Megjegyzés

Az `ORIEULER` programozással ellentétben az `ORIRPY`-nál mindhárom értéknek befolyása van a tájolási vektorra.

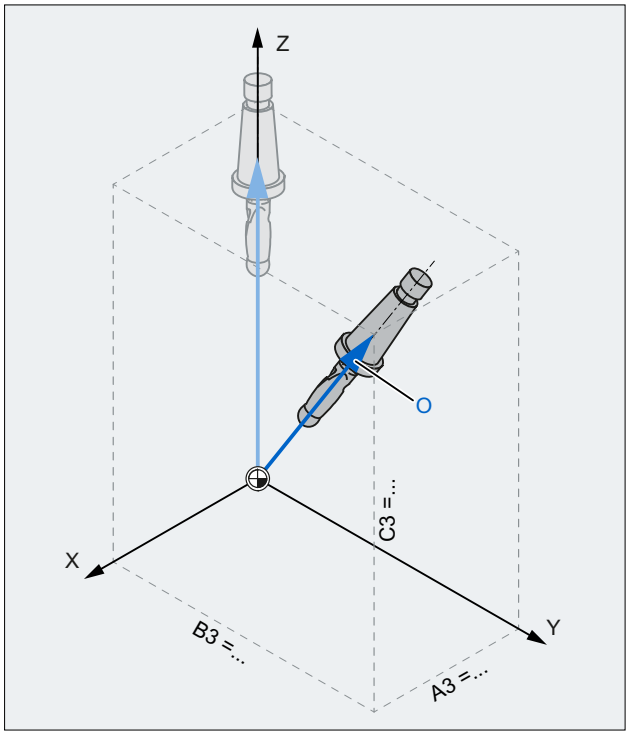
Az új tájolási vektor az eredeti tájolási vektor következő három forgatásából adódik

1. `A2` körtengellyel az X koordináta-tengely körül
2. `B2` körtengellyel az új Y' koordináta-tengely körül
3. `C2` körtengellyel az új Z'' koordináta-tengely körül

Irányvektor programozása

Az irányvektor összetevőit az `A3, B3, C3`-mal programozzuk. A vektor a szerszámfelvevő irányába mutat; a vektor hosszának itt nincsen jelentősége.

Nem programozott vektorösszetevők nullára lesznek állítva.



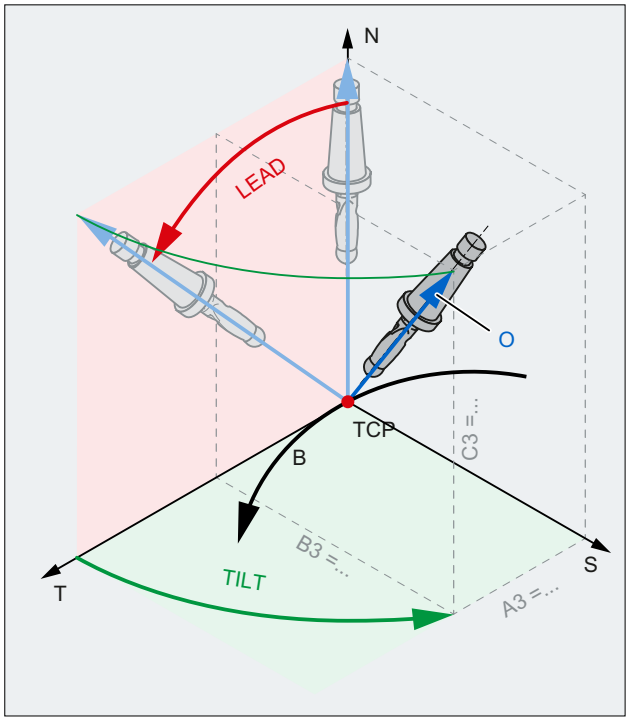
X, Y, Z	MKR koordináta-tengelyek
A3, B3, C3	irányvektor összetevők
O	tájolási vektorral

Kép 7-1 Irányvektor programozása

Szerszámtájolás programozása LEAD= és TILT=

Az eredő szerszámtájolást a következőkből kerül megállapításra:

- pálya-érintő
- felület normál-vektor
  a mondat elején `A4, B4, C4` és a mondat végén `A5, B5, C5`
- `LEAD` előre-sietés szög
  szög a pályaérintő és a felület-normálvektor által meghatározott síkban
- `TILT` oldalszög mondatvégen
  szög a síkban merőlegesen a pályaérintőre a felület-normálvektorhoz viszonyítva



T	pálya-érintő
S	merőleges a pályaérintőre
N	felület normál
B	pálya
TCP	Tool Center Point
O	tájolási vektorral

Kép 7-2 LEAD TILT programozása

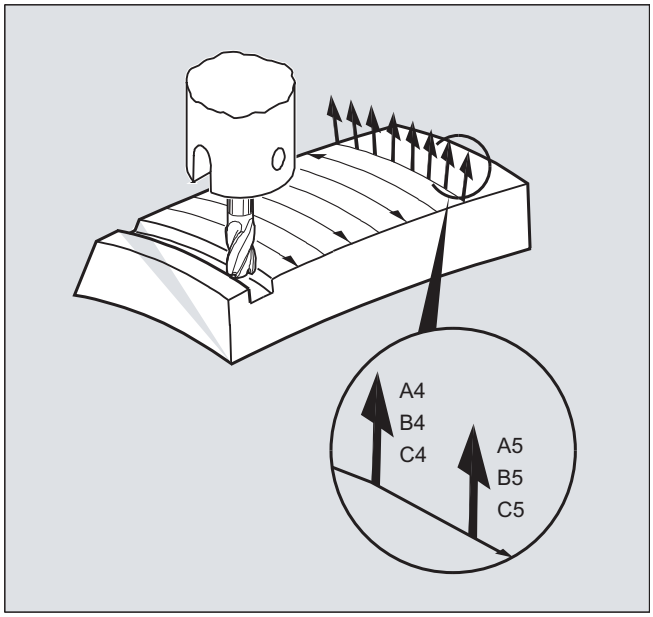
Megjegyzés

Viselkedés a belső sarkoknál 3D-s szerszámsugár-korrekciónál

Ha a mondat egy belső saroknál megrövidítésre kerül, akkor a programozott szerszámtájolás ennek ellenére a mondat végre lesz elérve.

7.2.5 Homlokmarás (A4, B4, C4, A5, B5, C5)

A homlokmarás a tetszőlegesen görbített felületek megmunkálására szolgál.



A 3D-marásnak ehhez a fajtájához szükséges a munkadarab-felületi 3D-pályák soronkénti leírása.

A számítások szokás szerint CAM-ban kerülnek végrehajtásra a szerszámforma és a szerszámméretek figyelembevételével. A készre kiszámított NC-mondatokat ezután posztprocesszorok olvassák be a vezérlésbe.

Pályagörbületek programozása

Felületek leírása

A pályagörbület leírása felület-normálvektorok által történik a következő összetevőkkel:

`A4, B4, C4` kezdet-vektor a mondat elején

`A5, B5, C5` vég-vektor a mondat végén

Ha egy mondatban csak az kezdő-vektor szerepel, akkor a felület-normálvektor az egész mondatban állandó marad. Ha egy mondatban csak a vég-vektor szerepel, akkor a megelőző mondat végső értékéből a nagykör-interpoláció által kerül interpolálásra a programozott végső érték.

Ha a kezdő- és vég-vektor van programozva, akkor a két irány között szintén a nagykör-interpolációval kerül sor interpolációra. Ezáltal folyamatosan sima pályák jönnek létre.

Az alapállásban a felület-normálvektorok függetlenül a G17.. G19 aktív síkoktól a Z-irányba mutatnak.

Egy vektor hosszának nincsen jelentősége.

Nem programozott vektorösszetevők nullára lesznek állítva.

Aktív ORIWKS esetén, (lásd "A tájolótengelyek vonatkoztatása (ORIWKS, ORIMKS) (Oldal 336)") a felület-normálvektorok az aktív frame-re vonatkoznak és a frame forgatásánál együtt forognak.

Gépgyártó

A felület-normálvektornak, egy gépadatok által beállított határértéken belül, a pályaérintőre merőlegesen kell állnia, különben vészjelzés kerül kiadásra.

7.2.6 A tájolótengelyek vonatkoztatása (ORIWKS, ORIMKS)

A tájolási programozásnál a munkadarab-koordinátarendszerben

- Euler- ill. RPY-szöggel vagy
- tájolási vektorral

a forgómozgás folyamata az `ORIMKS`/`ORIWKS` által állítható be.

Megjegyzés

Gépgyártó

Az interpoláció fajta a következő gépadattal lesz megadva:

MD21104 $MC_ORI_IPO_WITH_G_CODE

= FALSE: Vonatkozás az ORIWKS és ORIMKS G utasítások

= TRUE: Vonatkozás a G utasítások az 51. csoportból (ORIAXES, ORIVECT, ORIPLANE, ...)

Szintaxis

```
ORIMKS=...
ORIWKS=...
```

Jelentés

`ORIMKS`:	forgatás a gép-koordinátarendszerben
`ORIWKS`:	forgatás a munkadarab-koordinátarendszerben

Megjegyzés

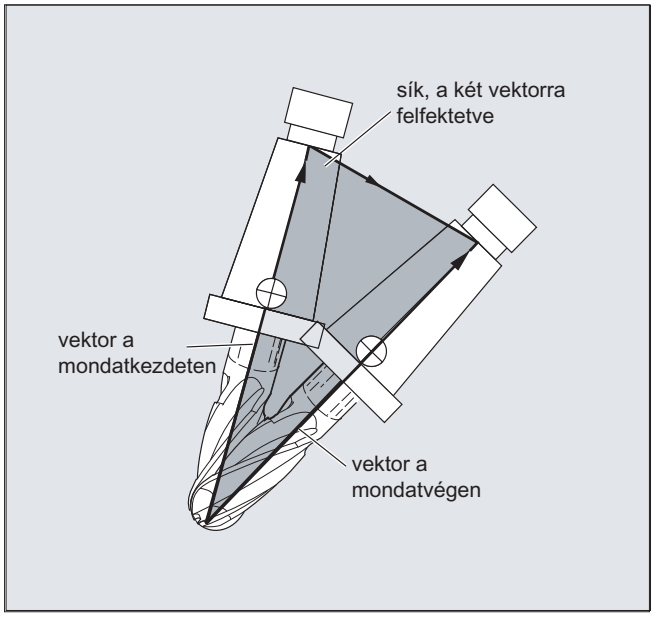
Az ORIWKS alapbeállítás. Ha egy öt-tengelyes programnál eleve nem világos, milyen gépen kell futtatni, akkor alapvetően célszerű az ORIWKS-t választani. A gépkinematikától függ, hogy a gép valójában milyen mozgásokat hajt végre.

Az `ORIMKS`-sal valóságos gépi mozgásokat lehet programozni, pl. a berendezésekkel történő ütközések elkerülése céljából.

További információ

Az ORIMKS-nél a végrehajtott szerszámmozgás a gépkinematikától **függő**. Térben rögzített szerszámcsúccsal történő tájolás-változtatásnál a körtengely-pozíciók között történik lineáris interpoláció.

Az ORIWKS-nél a végrehajtott szerszámmozgás a gépkinematikától **független**. Térben rögzített szerszámcsúccsal történő tájolás-változtatásnál a szerszám a kezdő- és végvektor által kifeszített síkban mozog.



Szinguláris helyek

Megjegyzés

ORIWKS

Tájolási mozgások az öt-tengelyes gépek szinguláris helyeinek tartományában a géptengelyek nagy mozgásait igénylik. (Például egy forgó-, billenőfejnél, C-vel, mint forgótengellyel és A-val, mint billenőtengellyel minden A = 0 hely szinguláris.)

Gépgyártó

Ahhoz, hogy ne terheljük túl a géptengelyeket, a sebességvezérlés erősen lecsökkenti a pályasebességet a szinguláris helyek közelében.

A

$MC_TRAFO5_NON_POLE_LIMIT

$MC_TRAFO5_POLE_LIMIT

gépadatokkal úgy lehet paraméterezni a transzformációt, hogy a tájolási mozgások a pólus közelében a póluson át történnek és ezáltal egy gyors megmunkálás lehetséges.

A szinguláris helyeket csak az $MC_TRAFO5_POLE_LIMIT gépadattal lehet kezelni.

Irodalom:
/FB3/ Különleges funkciók működési kézikönyv; 3...5-tengelyes transzformációk (F2), "Szinguláris helyek és jelentőségük" fejezet.

7.2.7 Tájolótengelyek programozása (ORIAXES, ORIVECT, ORIEULER, ORIRPY, ORIRPY2, ORIVIRT1, ORIVIRT2)

A "Tájolótengelyek" funkció a szerszám térben történő tájolását írja le és ez a körtengelyek offset programozásával lesz elérve. Egy további, harmadik szabadságfok kerül bevezetésre, amely a saját maga körüli mozgást írja le. Ez a szerszámtájolás egy körtengellyel a térben tetszőlegesen történik és igényli a hat-tengelyes transzformációt. A szerszám saját maga körüli forgása a forgásvektorok interpolációs módjától függően a THETA forgásszöggel lesz megadva, lásd ("Szerszámtájolás forgatásai (ORIROTA, ORIROTR, ORIROTT, ORIROTC, THETA) (Oldal 347)".

A tájolótengelyek a az A2, B2, C2 tengelyjelölőkkel lesznek programozva.

Szintaxis

```
N... ORIAXES/ORIVECT            ; lineáris vagy nagykör-interpoláció
N... G1 X Y Z A B C

N... ORIPLANE                   ; síkok tájolás-interpolációja

N... ORIEULER/ORIRPY/ORIRPY2    ; tájolási szög Euler-/RPY-szög
N... G1 X Y Z A2= B2= C2=       ; virtuális tengelyek szög-programozása

N... ORIVIRT1/ORIVIRT2          ; virtuális tájolás-tengelyek def. 1/2
N... G1 X Y Z A3= B3= C3=       ; irányvektor programozás
```

Megjegyzés

A tájolás változásokhoz egy térbeli kúppalást-felület mentén a tájolótengelyeknél további körtengely-offsetek programozhatók (lásd "Tájolás programozása egy kúppalást-felület mentén (ORIPLANE, ORICONCW, ORICONCCW, ORICONTO, ORICONIO) (Oldal 340)".

Jelentés

ORIAXES:	gép- vagy tájolótengelyek lineáris interpolációja
ORIVECT:	nagykör-interpoláció (azonos ORIPLANE-nel)
ORIMKS: ORIWKS:	forgatás a gép-koordinátarendszerben forgatás a munkadarab-koordinátarendszerben Leírást lásd "A tájolótengelyek vonatkoztatása (ORIWKS, ORIMKS) (Oldal 336)".
A= B= C=:	géptengely-pozíció programozása

ORIEULER:	tájolás programozása Euler-szöggel
ORIRPY:	tájolás programozása RPY-szöggel A forgatás sorrendje XYZ, ahol: • A2 forgásszög X körül • B2 forgásszög Y körül • C2 forgásszög Z körül
ORIRPY2:	tájolás programozása RPY-szöggel A forgatás sorrendje ZYX, ahol: • A2 forgásszög Z körül • B2 forgásszög Y körül • C2 forgásszög X körül
A2= B2= C2=:	virtuális tengelyek szög-programozása
ORIVIRT1/ORIVIRT2:	tájolás programozása virtuális tájolástengelyekkel definíció 1: megadás MD21120 $MC_ORIAX_TURN_TAB_1 szerint definíció 2: megadás MD21130 $MC_ORIAX_TURN_TAB_2 szerint
A3= B3= C3=:	iránytengelyek irányvektor-programozása

További információk

Gépgyártó

Az MD21102 $MC_ORI_DEF_WITH_G_CODE gépadattal kerül meghatározásra, hogyan definiáljuk az A2, B2, C2 programozott szögeket:

A definíció az MD21100 $MC_ORIENTATION_IS_EULER gépadat után következik (alaphelyzet) vagy az 50-es G-csoportot követi (ORIEULER, ORIRPY, ORIVIRT1, ORIVIRT2).

Az MD21104 $MC_ORI_IPO_WITH_G_CODE gépadattal kerül meghatározásra, hogy az interpoláció melyik fajtája hatásos: ORIWKS/ORIMKS vagy ORIAXES/ORIVECT.

JOG üzemmód

A tájolási szögek ebben az üzemmódban mindig lineárisan vannak interpolálva. A mozgatóbillentyűk általi folyamatos és inkrementális mozgatásnál csak egy tájolótengelyt lehet mozgatni. A kézikerekekkel egyidejűleg lehet mozgatni a tájolótengelyeket.

A tájolótengelyek kézi mozgatására hat a csatorna-specifikus előtolás-override kapcsoló ill. a gyorsmenet-override kapcsoló a gyorsmenet-átlapolásnál.

Az alábbi gépadatokkal egy külön sebesség-előmegadás lehetséges:

MD21160 $MC_JOG_VELO_RAPID_GEO

MD21165 $MC_JOG_VELO_GEO

MD21150 $MC_JOG_VELO_RAPID_ORI

MD21155 $MC_JOG_VELO_ORI

Megjegyzés

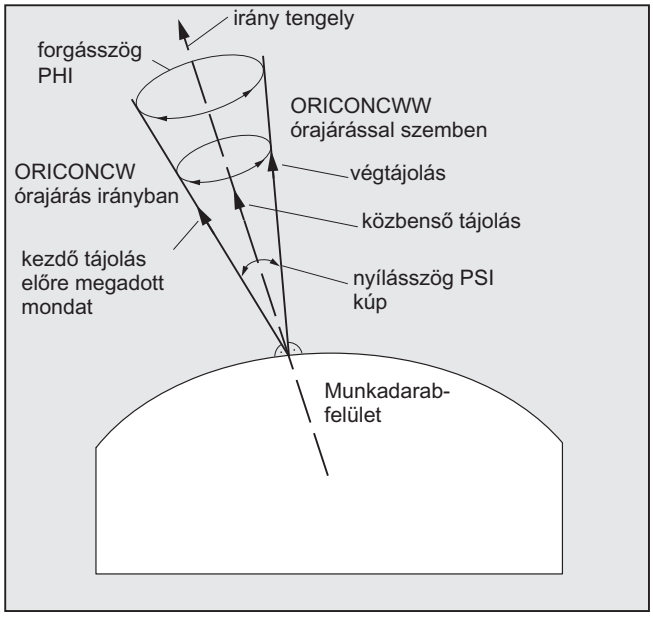
SINUMERIK 840D sl "Handling transzformációs csomag"-gal

A "Derékszögű kézi-mozgatás" funkcióval a JOG-üzemben a geometria-tengelyek mozgatása a GKR, MKR és TKR vonatkoztatási rendszerekben egymástól külön beállítható.

Irodalom:
Bővítő funkciók működési kézikönyv; Kinematikai transzformációk (M1)

7.2.8 Tájolás programozása egy kúppalást-felület mentén (ORIPLANE, ORICONCW, ORICONCCW, ORICONTO, ORICONIO)

A kibővített tájolással lehetséges tájolás változásokat végrehajtani egy térbeli kúppalást-felület mentén. A tájolásvektor interpolációja egy kúppalást-felületen az ORICONxx modális utasításokkal történik. A végtájolást egy síkban történő interpolációnál lehet ORIPLANE-nel programozni. Általában a kezdő-tájolást az előző mondatok határozzák meg.



Programozás

A végtájolás vagy a szögprogramozás megadásával Euler- vagy RPY-szögben A2, B2, C2 által vagy a körtengely-pozíciók programozásával A, B, C által lesz rögzítve. A tájolótengelyekre a kúppalást-felület mentén további programozások szükségesek:

- a kúp forgástengelye vektorként az `A6, B6, C6` által
- nyílásszög `PSI` a `NUT` jelölővel
- közbenső tájolás a kúppaláston az `A7, B7, C7`által

Megjegyzés

A6, B6, C6 irányvektor programozása a kúp forgástengelyéhez

A végtájolás programozása nem feltétlenül szükséges. Ha nincs megadva végtájolás, akkor egy teljes kúppalást lesz interpolálva 360 fokkal.

Kúp nyílásszögének programozása NUT=szög-gel

A végtájolás programozása feltétlenül szükséges.

Ezen módon nem lehet egy teljes kúppalást lesz interpolálni 360 fokkal.

A7, B7, C7 közbenső tájolás programozása a kúppaláston

A végtájolás programozása feltétlenül szükséges. A tájolás változást és a forgásirányt egyértelműen meghatározza a kezdő-, vég- és közbenső-tájolás három vektora. Mind a három vektor különböző kell legyen. Ha a programozott közbenső tájolás párhuzamos a kezdő- és végtájolással, akkor a tájolásra egy lineáris nagykör-interpoláció lesz végrehajtva a kezdő- és vég-vektor által meghatározott síkban.

Kibővített tájolás-interpoláció egy kúppalást-felületen

`N... ORICONCW` vagy `ORICONCCW`	**interpoláció** egy kúppaláston
`N... A6= B6= C6= A3= B3= C3=`	irány-vektorral a kúp órajárás-irányába/ szembe és végtájolás vagy
vagy	
`N... ORICONTO`	érintőleges átmenet és
`N... G1 X Y Z A6= B6= C6=`	a végtájolás megadása
vagy	vagy
`N... ORICONIO`	végtájolás és egy közbenső tájolás megadása a kúppaláston
`N... G1 X Y Z A7= B7= C7=`	
`N... PO[PHI]=(a2, a3, a4, a5)`	forgásszög polinommal és
`N... PO[PSI]=(b2, b3, b4, b5)`	nyílásszög polinommal

Paraméter

`ORIPLANE`:	interpoláció a síkban (nagykör-interpoláció)
`ORICONCW`:	interpoláció egy kúppalást-felületen az órajárás irányában
`ORICONCCW`:	interpoláció egy kúppalást-felületen az órajárás irányával szemben
`ORICONTO`:	interpoláció egy kúppalást-felületen érintőleges átmenet
`A6= B6= C6=`:	kúp forgástengelyének programozása (normált vektor)

NUT=szög:	kúp nyílásszöge fokban
NUT=+179:	elmozdulás-szög kisebb vagy egyenlő 180 fokkal
NUT=-181:	elmozdulás-szög nagyobb vagy egyenlő 180 fokkal
ORICONIO:	interpoláció egy kúppalást-felületen
A7= B7= C7=:	közbenső tájolás(programozás normált vektorként)
PHI:	tájolás forgásszöge a kúp iránytengelye körül
PSI:	kúp nyílásszöge
lehetséges polinomok PO[PHI]=(a2, a3, a4, a5) PO[PSI]=(b2, b3, b4, b5)	A mindenkori szögeken kívül max.5. fokú polinomok is programozhatók

Példa: Különféle tájolás változások

Programkód	Kommentár
…	
N10 G1 X0 Y0 F5000	
N20 TRAORI(1)	; Tájolás-transzformáció be.
N30 ORIVECT	; Szerszámtájolást vektorként interpolálni.
…	; Szerszámtájolás a síkban.
N40 ORIPLANE	; Nagykör-interpolációt kiválasztani.
N50 A3=0 B3=0 C3=1	
N60 A3=0 B3=1 C3=1	; Tájolás az Y/Z-síkban 45 fokkal elforgatva, tájolás a mondat végén lesz elérve (0,1/√2,1/√2).
…	
N70 ORICONCW	; Tájolás-programozás kúppalástón:
N80 A6=0 B6=0 C6=1 A3=0 B3=0 C3=1	; A tájolás-vektor egy kúppalástón lesz (0,0,1) iránnyal a tájolásig (1/√2,0,1/√2) órajárás irányában interpolálva, a forgásszög itt 270 fok.
N90 A6=0 B6=0 C6=1	; A szerszámtájolás egy teljes fordulatot tesz meg ugyanazon a kúppalástón.

További információ

Ha a tájolás-változásokat egy tetszőleges térbeli kúppalást-felületen kell leírni, akkor ismert kell legyen a vektor, ami körül a szerszámtájolást el kell forgatni. Ezen kívül meg kell adni a kezdő- és a végtájolást. A kezdő-tájolás a megelőző mondatból adódik és a végtájolást vagy programozni kell vagy más feltételekkel kell megadni.

ORIPLANE programozás a síkban megfelel ORIVECT-nek

A nagykör-interpoláció programozása a szög-polinomokkal együtt megfelel a kontúrok egyenes- és polinom-interpolációjának. A szerszámtájolás egy, a kezdő- és a végtájolás által kijelölt síkban lesz interpolálva. Ha további polinomok lesznek programozva, akkor a tájolás-vektor a síkból kibillenhet.

Körök programozása egy síkban G2/G3, CIP és CT

A kibővített tájolás megfelel a körök interpolációjának egy síkban. A körök megfelelő programozási lehetőségeit középpont megadással vagy sugár megadásával mint G2/G3, kör közbenső ponton át CIP és érintőkör CT, lásd

Irodalom: Alapok programozási kézikönyv , "Út-utasítások programozása".

Tájolás programozása

Tájolás-vektor interpolációja egy kúppalást-felületen ORICONxx

A tájolások interpolációjához egy kúppalást-felületen négy különböző interpolációs módot lehet kiválasztani a G-csoport 51 G utasításaiból:

1. Interpoláció egy kúppalástón az órajárás irányában `ORICONCW` a végtájolás és a kúpirány vagy a nyílásszög megadásával. Az irányvektor az `A6, B6, C6` jelölőkkel és a kúp nyílásszöge a NUT= jelölővel a 0 ...180 fok értéktartományban lesz programozva.
2. Interpoláció egy kúppalástón az órajárás irányával szemben `ORICONCWW` a végtájolás és a kúpirány vagy a nyílásszög megadásával. Az irányvektor az `A6, B6, C6` jelölőkkel és a kúp nyílásszöge a NUT= jelölővel a 0 ...180 fok értéktartományban lesz programozva.
3. Interpoláció egy kúppalástón `ORICONIO` a végtájolás és egy közbenső tájolás megadásával, ami a `A7, B7, C7` jelölőkkel lesz programozva.
4. Interpoláció egy kúppalástón `ORICONTO` érintőleges átmenettel és a végtájolás megadásával.. Az irányvektor az `A6, B6, C6` jelölőkkel lesz programozva.

7.2.9 Két érintőpont tájolás-megadása (ORICURVE, PO[XH]=, PO[YH]=, PO[ZH]=)

Tájolás változásának programozása a második térgörbével ORICURVE

A tájolás-változások programozásának egy további lehetősége a szerszám csúcsán kívül a szerszám egy másik érintőpontjának mozgását is programozni egy térgörbe mentén `ORICURVE`-val. Ezzel a szerszám tájolás-változásait, akárcsak a szerszámvektor programozásánál egyértelműen meg lehet adni.

Gépgyártó

Vegye figyelembe a gépgyártó tájékoztatásait a gépadattal beállítható tengely-jelölőről a szerszám 2. tájolási pályájának programozásához.

Programozás

Ennél az interpolációs módnál mindkét térgörbe számára programozhatók pontok `G1` -gyel ill. polinomok `POLY`-val. Körök és evolvensek nem megengedettek. Kiegészítőleg lehet egy Spline-interpolációt `BSPLINE`-nal és a "Rövid Spline-mondatok összefogása" funkciót aktiválni.

Irodalom:
/FB1/ Alapfunkciók működési kézikönyv; Pályavezérlő üzem, Pontos-állj, Look Ahead (B1), fejezet: Rövid Spline-mondatok összefogása

A más ASPLINE és CSPLINE Spline-fajták és egy kompresszor aktiválása COMPON, COMPCURV vagy COMPCAD által nem megengedett.

A szerszám két érintőpontjának mozgását a koordináták tájolási polinomjainak programozásánál max 5. fokig lehet megadni.

Kibővített tájolás-interpoláció kiegészítő térgörbével és koordináta polinomokkal

```
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)
```

A szerszám második érintőpontja mozgásának megadása és a mindenkori koordináták kiegészítő polinomjai

Paraméter

ORICURVE	tájolás interpolációja a szerszám második érintőpontja mozgásának megadásával
XH YH ZH	a szerszám második érintőpontja koordinátáinak jelölője kiegészítő kontúr térgörbeként
lehetséges polinomok PO[XH]=(xe, x2, x3, x4, x5) PO[YH]=(ye, y2, y3, y4, y5) PO[ZH]=(ze, z2, z3, z4, z5)	a térgörbék a mindenkori végpontokon kívül kiegészítőleg programozhatók polinomokkal
xe, ye, ze	térgörbe végpontja
xi, yi, zi	max. 5. fokú polinom együtthatói

Megjegyzés

XH YH ZH jelölők egy 2. tájolási pálya programozásához

A jelölőket úgy kell választani, hogy ne legyen konfliktus a más jelölőkkel, mint

X Y Z tengelyek

és körtengelyekkel, mint

A2 B2 C2 Euler-szög ill. RPY-szög

A3 B3 C3 irányvektorok

A4 B4 C4 ill. A5 B5 C5 felület-normálvektorok

A6 B6 C6 forgásvektorok ill. A7 B7 C7 közbenső-pont koordináták

vagy más interpolációs paraméterek.

7.3 Tájolási polinomok (PO[szög], PO[koordináta])

Függetlenül attól, hogy melyik polinom-interpoláció aktív éppen a G-csoport 1-ből, egy három-...öt-tengelyes transzformációnál két különböző típusú, max. 5. fokú tájolási polinom programozható.

1. polinom **szögre:** LEAD előre-sietési szög, TILT oldalszög a síkra vonatkoztatva, amit a kezdő- és végtájolás kijelöl.
2. polinom **koordinátákra:** XH, YH, ZH a második térgörbéhez szerszámtájolásnál egy szerszámon levő vonatkoztatási ponttal.

Egy hat-tengelyes transzformációnál a szerszámtájoláshoz kiegészítőleg programozható a THT forgásvektor forgatása egy max. 5. fokú polinom a szerszám forgatására.

Szintaxis

Tájolási polinom típus 1 **szögre**

`N… PO[PHI]=(a2, a3, a4, a5)` `N… PO[PSI]=(b2, b3, b4, b5)`	három-...öt-tengelyes transzformáció

Tájolási polinom típus 2 **koordinátákra**

`N… PO[XH]=(xe, x2, x3, x4, x5)`
`N… PO[YH]=(ye, y2, y3, y4, y5)`
`N… PO[ZH]=(ze, z2, z3, z4, z5)`
	szerszámtájolás második tájolási pálya koordinátáinak jelölése

Mindkét esetben lehet kiegészítőleg egy polinomot a **forgatásra** hat-tengelyes transzformációnál

`N… PO[THT]=(c2, c3, c4, c5)`
vagy
`N… PO[THT]=(d2, d3, d4, d5)`
	a forgatás pályára vonatkoztatott interpolációja
	
	abszolút, relatív és érintőleges interpoláció tájolás-változáshoz

a tájolás-vektorra programozni. Ez akkor lehetséges, ha a transzformáció támogatja a THETA forgásszöggel programozható forgásvektort interpolálható offsettel.

Jelentés

`PO[PHI]`	szög a kezdő- és végtájolás között a síkban
`PO[PSI]`	tájolás kezdő- és végtájolás közötti síkból kibillenésének szöge
`PO[THT]`	G-csoport 54 egyik THETA-val programozott G utasításának forgásvektor forgásának forgásszöge
`PHI`	LEAD előre-sietés szög
`PSI`	TILT oldalszög
`THETA`	forgatás a Z szerszámirány körül

PO[XH]	szerszámon levő vonatkoztatási pont X-koordinátája
PO[YH]	szerszámon levő vonatkoztatási pont Y-koordinátája
PO[ZH]	szerszámon levő vonatkoztatási pont Z-koordinátája

További információ

A tájolási polinomok nem programozhatók

- ha a ASPLINE, BSPLINE, CSPLINE Spline-interpolációk aktívak.
  Típus 1 polinomok tájolási szögre minden interpolációs módnál lehetségesek a Spline-on kívül, azaz lineáris interpolációnál G00 gyorsmenettel ill. G01 előtolással polinom-interpolációnál POLY-val és
  kör- ill. evolvens-interpolációnál G02, G03, CIP, CT, INVCW és INCCCW esetében.
  Típus 2 polinomok tájolási koordinátákra ezzel szemben csak akkor lehetségesek, ha lineáris interpolációnál G00 gyorsmenettel ill. G01 előtolással vagy polinom-interpolációnál POLY-val aktív.
- ha a tájolás ORIAXES tengely-interpolációval lesz interpolálva. Ebben az esetben közvetlenül lehet polinomokat programozni PO[A] és PO[B] alkalmazásával az A és B tájolási tengelyekhez.

Tájolási polinom típus 1 ORIVECT, ORIPLANE és ORICONxx használatával

Nagykör-interpolációnál és kúppalást-interpolációnál ORIVECT, ORIPLANE és ORICONxx használatával csak típus 1-es tájolási polinomok lehetségesek.

Tájolási polinom típus 2 ORICURVE használatával

Ha az interpoláció az ORICURVE kiegészítő térgörbével aktív, a tájolási vektor derékszögű komponensei lesznek interpolálva és csak típus 2-es tájolási polinomok lehetségesek.

7.4 Szerszámtájolás forgatásai (ORIROTA, ORIROTR, ORIROTT, ORIROTC, THETA)

Ha mozgatható szerszámmal rendelkező géptípusoknál a szerszám tájolása is változtatható kell legyen, akkor minden mondat egy végtájolással lesz programozva. A gép-kinematikától függően vagy a tájolótengelyek tájolásiránya vagy a THETA tájolás-vektor forgásiránya programozható. Ezekhez a forgásvektorokhoz különféle interpolációs módok programozhatók:

- ORIROTA: forgásszög egy abszolút megadott forgásirányhoz
- ORIROTR: forgásszög a kezdő- és végtájolás közötti síkhoz viszonyítva
- ORIROTT: forgásszög a tájolási vektor változásához viszonyítva
- ORIROTC: érintőleges forgásszög a pályaérintőhöz.

Szintaxis

Csak ha az ORIROTA interpolációs már aktív, lehetséges a forgásszög vagy forgásvektor programozása az alábbi négy módon:

1. közvetlenül az `A, B, C` körtengely-pozíciókkal
2. Euler-szöggel (fokban) `A2, B2, C2` által
3. RPY-szöggel (fokban) `A2, B2, C2` által
4. irányvektorral `A3, B3, C3` által (forgásszög `THETA=<érték>`)

Ha az ORIOTR vagy ORIOTT aktív, a forgásszög csak közvetlenül `THETA`-val programozható.

Egy forgatást lehet egyedül is programozni egy mondatban, anélkül hogy tájolás-változás történne. Ekkor az ORIROTR-nek és ORIROTT -nak nincs jelentősége. Ebben az esetben a forgásszög mindig az abszolút irányra vonatkoztatva lesz értelmezve (ORIROTA).

`N... ORIROTA`	forgásvektor interpolációját megadni
`N... ORIROTR`	
`N... ORIROTT`	
`N... ORIROTC`	
`N... A3= B3= C3= THETA=<érték>`	tájolási vektor forgatását megadni
`N... PO[THT]=(`d_2`, `d_3`, `d_4`, `d_5`)`	forgásszöget 5. fokú polinommal interpolálni

Jelentés

`ORIROTA`:	forgásszög egy abszolút megadott forgásirányhoz
`ORIROTR`:	forgásszög a kezdő- és végtájolás közötti síkhoz viszonyítva
`ORIROTT`:	forgásszög érintőleges forgásvektorként tájolás-változáshoz
`ORIROTC`:	forgásszög érintőleges forgásvektorként a pályaérintőhöz
`THETA`:	tájolás-vektor forgatása
`THETA=<érték>`:	forgásszög fokban, ami a mondat végén lesz elérve
`THETA=Θe`:	forgásszög a forgásvektor Θ_e végszögével

THETA=AC(…):	mondatonként abszolút méretmegadásra átkapcsolni
THETA=AC(…):	mondatonként relatív méretmegadásra átkapcsolni
Θe:	forgásvektor végszöge G90 abszolút és G91 relatív (láncméret) aktív esetében is
PO[THT]=(....):	polinom forgásszöghöz

Példa: Tájolások forgatásai

```
Programkód                                  Kommentár
N10 TRAORI                                  ; tájolás-transzformációt aktiválni
N20 G1 X0 Y0 Z0 F5000                       ; szerszám tájolása
N30 A3=0 B3=0 C3=1 THETA=0                  ; Z-irányban 0 forgásszöggel
N40 A3=1 B3=0 C3=0 THETA=90                 ; X-irányban és 90 fokos forgatással
N50 A3=0 B3=1 C3=0 PO[THT]=(180,90)         ; tájolás
N60 A3=0 B3=1 C3=0 THETA=IC(-90)            ; Y-irányban és 180 fokos forgatással
N70 ORIROTT                                 ; állandó marad és forgatás 90 fokra
N80 A3=1 B3=0 C3=0 THETA=30                 ; forgásszög a tájolás-változáshoz viszonyít-
                                            va
                                            ; forgásvektor 30 fokos szögben az X-Y síkhoz
```

Az interpolációnál az N40 mondatban a forgásszög a 0 fok kezdőértékről a 90 fok végértékre lineárisan lesz interpolálva. Az N50 mondatban a forgásszög 90 fokról 180 fokra változik a parabola θ(u) = +90u2-nek megfelelően Az N60-ban végre lehet hajtani egy forgatást is anélkül, hogy tájolás-változás történne.

Az N80-nál a szerszámtájolás az Y-irányból az X-irányba lesz forgatva. Ennél a tájolás-változás az X–Y síkban van és a forgásvektor ehhez a síkhoz egy 30 fokos szöget zár be.

További információ

ORIROTA

A THETA forgásszög egy a térben abszolút megadott irányra vonatkoztatva lesz interpolálva. Az alap-forgásirányt gépadat adja meg

ORIROTR

A THETA forgásszög a kezdő- és végtájolás közötti síkhoz viszonyítva lesz értelmezve.

ORIROTT

A THETA forgásszög a tájolás-változáshoz viszonyítva lesz értelmezve. A THETA=0 esetén a forgásvektor a tájolás-változáshoz érintőlegesen lesz interpolálva és csak akkor különbözik az ORIROTR-tól, ha a tájolásra legalább egy polinom lett programozva "PSI billenési szögre". Ezzel egy olyan tájolás-változás adódik, amelyik nem a síkban fut le. Egy kiegészítőleg programozott THETA forgásszöggel pl. a forgásvektort úgy lehet interpolálni, hogy mindig egy adott értéket képez a tájolás-változáshoz.

ORIROTC

A forgásvektor a pályaérintőhöz viszonyítva egy a `THETA` szöggel programozható offsettel lesz interpolálva. Az offset-szögre lehet egy max. 5. fokú `PO[THT]=(c2, c3, c4, c5)` polinomot is lehet programozni.

7.5 Pálya-viszonyítású tájolás

7.5.1 Pályához viszonyított tájolási módok

Ezzel a kibővített funkcióval a relatív tájolás nemcsak a mondat végén, hanem a teljes pálya mentén elérhető. Az előző mondatban elért tájolás egy nagykör-interpolációval át lesz vezetve a programozott végtájolásba. Alapvetően két lehetőség van a kívánt tájolást a pályához viszonyítva programozni:

1. A szerszámtájolás és a szerszám forgatása is ORIPATH, ORPATHTS alkalmazásával a pályához viszonyítva lesz interpolálva.
2. A tájolási vektor úgy lesz programozva és interpolálva, mint eddig. Az ORIROTC-cal a tájolás-vektor forgatása a pályaérintőhöz viszonyítva lesz beállítva.

Szintaxis

A tájolás interpolációs módja és a szerszám forgatása a következőkkel lesz programozva:

`N... ORIPATH`	pályához viszonyított tájolás
`N... ORIPATHS`	pályához viszonyított tájolás a tájolás lefutásának simításával
`N... ORIROTC`	forgásvektor pályához viszonyított interpolációja

A pálya lefutásában egy saroknál keletkező törés a tájolásban az `ORIPATHS` alkalmazásával simítható. A leemelő mozgás iránya és úthossza az `A8=X, B8=Y C8=Z` komponensű vektorral lesz programozva.

Az `ORIPATH/ORIPATHS` alkalmazásával különböző vonatkozásokat lehet a pályaérintőhöz a következő három szöggel

- `LEAD=` előreszög megadása a pályára és a felületre vonatkoztatva
- `TILT=` oldalszög megadása a pályára és a felületre vonatkoztatva
- `THETA=` forgásszög

a teljes pálya mentén programozni. A `THETA` forgásszöghöz a `PO[THT]=(...)` max. 5. fokú kiegészítő polinom programozható.

Megjegyzés

Gépgyártó

Vegyük ehhez figyelembe a gépgyártó tájékoztatásait. A megtervezhető gép- és beállítási adatokkal a pályához viszonyított tájolási módokhoz további beállítások adhatók meg. További magyarázatok:

Irodalom:
/FB3/ Különleges funkciók működési kézikönyv; 3...5 tengelyes transzformáció (F2), "Tájolás" fejezet

---

Jelentés

A LEAD és TILT szögek interpolációja gépadatokkal különbözőképp beállítható:

- A LEAD és TILT segítségével programozott szerszámtájolás vonatkoztatás az egész mondatban megmarad.
- előreszög LEAD: forgatás az érintőre és a normálvektorra merőleges irány körül TILT: tájolás forgatása a normálvektor körül
- előreszög LEAD: forgatás az érintőre és a merőleges irány körül és normálvektor oldalszög TILT: tájolás forgásszöge a kúp iránytengelye körül
- forgásszög THETA: szerszám forgatása saját maga körül egy kiegészítő harmadik körtengellyel tájolótengelyként hat-tengelyes transzformációnál

Megjegyzés

Pályára vonatkoztatott tájolás együtt OSC, OSS, OSSE, OSD, OST-vel nem megengedett

Az ORIPATH ill. ORIPATHS és ORIOTC pályára vonatkoztatott tájolási interpolációk nem programozhatók együtt a tájolás lefutásának simítására egy G utasítással a 34-es G-csoportból. Ehhez az OSOF kell aktív legyen.

7.5.2 Szerszámtájolás pályára vonatkoztatott forgatása (ORIPATH, ORIPATHS, forgásszög)

Egy hat-tengelyes transzformációnál a szerszámtájoláshoz tetszőlegesen a térben a szerszámot is saját maga körül lehet forgatni egy harmadik körtengellyel. A szerszámtájolás pályára vonatkoztatott forgatásánál ORIPATH ill. ORIPATHS-szal a kiegészítő forgatást a THETA forgásszöggel lehet programozni. Ennek alternatívájaként LEAD és TILT szögeket lehet programozni egy vektorral, amelyik a síkban merőlegesen áll a szerszámiránya.

Gépgyártó

Vegyük ehhez figyelembe a gépgyártó tájékoztatásait. A LEAD és TILT szögek interpolációját gépadattal eltérőre lehet beállítani.

Szintaxis

Szerszámtájolás forgatása a szerszám körül

A pályára vonatkoztatott szerszámtájolási mód ORIPATH vagy ORIPATHS megadásával lesz aktiválva.

N... ORIPATH	pályára vonatkoztatott tájolási mód aktiválása
N... ORIPATHS	pályára vonatkoztatott tájolási mód aktiválása a tájolás lefutásának simításával

A három lehetséges forgás-hatású szög aktiválása:

N... LEAD=	szög a programozott tájolásra a felület-normálvektorhoz viszonyítva

`N... TILT=`	szög a programozott tájolásra a pályaérintőre merőleges síkban a felület-normálvektorhoz viszonyítva
`N... THETA=`	forgásszög a tájolás-változáshoz viszonyítva a harmadik körtengely szerszámiránya körül

A szögek értékei a mondat végén `LEAD=érték`, `TILT=érték` ill. `THETA=érték` megadásával lesznek programozva. Kiegészítőleg az állandó szögekhez mind s három szögre lehet max. 5. fokú polinomokat programozni.

`N... PO[PHI]=(a2, a3, a4, a5)`	polinom a LEAD előreszögre
`N... PO[PSI]=(b2, b3, b4, b5)`	polinom a TILT oldalszögre
`N... PO[THT]=(d2, d3, d4, d5)`	polinom a THETA forgásszögre

A programozásnál a nulla értékű magasabb polinom-együtthatók elhagyhatóak. Például `PO[PHI]=a2` a `LEAD` előreszögre egy parabolát ad.

Jelentés

Pályára vonatkoztatott szerszámtájolás

`ORIPATH`:	pályára vonatkoztatott szerszámtájolás
`ORIPATHS`:	pályára vonatkoztatott szerszámtájolás, törés a tájolás lefutásában simítva lesz
`LEAD`:	szög a felület-normálvektorhoz viszonyítva, a pályaérintő és a felület-normálvektor által meghatározott síkban
`TILT`:	tájolás forgatása a Z-irány ill. a pályaérintő körül
`THETA`:	forgatás a szerszámirány körül Z felé
`PO[PHI]`:	tájolási polinom a LEAD előreszöghöz
`PO[PSI]`:	tájolási polinom a TILT oldalszöghöz
`PO[THT]`:	tájolási polinom a THETA forgásszöghöz

Megjegyzés

THETA forgásszög

A szerszám forgatásához önmaga körül a harmadik körtengellyel, mint tájolótengellyel szükséges a hat-tengelyes transzformáció.

7.5.3 Szerszámforgatás pályára vonatkozatott interpolációja (ORIROTC, THETA)

Interpoláció forgásvektorokkal

A szerszámnak az ORIROTC-cal programozott pályaérintőhöz viszonyított forgatásához lehet a forgásvektort egy THETA forgásszöggel programozható offsettel is interpolálni. Ehhez az offset-szögre a PO[THT]-tal lehet egy max. 5. fokú polinomot programozni.

Szintaxis

`N... ORIROTC`	szerszám forgatását a pályaérintőhöz viszonyítva beállítani
`N... A3= B3= C3= THETA=érték`	tájolási vektor forgatását megadni
`N... A3= B3= C3= PO[THT]=(c2, c3, c4, c5)`	offset-szöget max. 5. fokú polinommal interpolálni

Egy forgatást lehet egyedül is programozni egy mondatban, anélkül hogy tájolás-változás történne.

Jelentés

Szerszám forgatásának pályára vonatkoztatott interpolációja hat-tengelyes transzformációnál

`ORIROTC`:	érintőleges forgásvektort a pályaérintőhöz beállítani
`THETA=érték`:	forgásszög fokban, ami a mondat végén lesz elérve
`THETA=θe`:	forgásszög a forgásvektor Θ_e végszögével
`THETA=AC(…)`:	mondatonként abszolút méretmegadásra átkapcsolni
`THETA=IC(…)`:	mondatonként relatív méretmegadásra átkapcsolni
`PO[THT]=(c2, c3, c4, c5)`:	offset-szöget 5. fokú polinommal interpolálni

Megjegyzés

Forgásvektor interpolációja ORIROTC

Ha a szerszám tájolási irányával szemben a szerszám forgatását is a pályairányhoz viszonyítva kell beállítani, akkor ez csak egy hat-tengelyes transzformációnál lehetséges.

Aktív ORIROTC esetén

Az ORIROTA forgásvektort nem lehet programozni. A programozás esetén a 14128 "Szerszámforgatás abszolút programozása aktív ORIROTC esetén" vészjelzés lesz kiadva.

A szerszám tájolási iránya három-...öt-tengelyes transzformációnál

A szerszám tájolási irányát a három-...öt-tengelyes transzformációnál megszokott módon Euler-szöggel ill. RPY-szöggel vagy irányvektorral lehet programozni. Lehetséges az ORIVECT szerszám tájolás-változása a térben nagykör-interpoláció programozásával, az ORIAXES tájolási tengelyek lineáris interpolációja, az összes interpolációs mód ORICONxx egy kúppalást-felületen, a térbeli görbéhez kiegészítőleg az ORICURVE interpoláció a szerszám két érintőpontjával .

`G....`:	körtengelyek mozgásmódjának megadása
`X, Y, Z`:	lineáris tengelyek megadása
`ORIAXES`:	gép- vagy tájolótengelyek lineáris interpolációja
`ORIVECT`:	nagykör-interpoláció (azonos ORIPLANE-nel)

ORIMKS: ORIWKS:	forgatás a gép-koordinátarendszerben forgatás a munkadarab-koordinátarendszerben leírást lásd Szerszámtájolás forgatásai fejezet
A= B= C=:	géptengely-pozíció programozása
ORIEULER:	tájolás programozása Euler-szöggel
ORIRPY:	tájolás programozása RPY-szöggel
A2= B2= C2=:	virtuális tengelyek szög-programozása
ORIVIRT1: ORIVIRT2:	tájolás programozása virtuális tájolástengelyekkel (definíció 1), megadás MD $MC_ORIAX_TURN_TAB_1 szerint (definíció 2), megadás MD $MC_ORIAX_TURN_TAB_2 szerint
A3= B3= C3=:	iránytengelyek irányvektor-programozása
ORIPLANE:	interpoláció a síkban (nagykör-interpoláció)
ORICONCW:	interpoláció egy kúppalást-felületen az órajárás irányában
ORICONCCW:	interpoláció egy kúppalást-felületen az órajárás irányával szemben
ORICONTO:	interpoláció egy kúppalást-felületen érintőleges átmenet
A6= B6= C6=:	kúp forgástengelyének programozása (normált vektor)
NUT=szög	kúp nyílásszöge fokban
NUT=+179	elmozdulás-szög kisebb vagy egyenlő 180 fokkal
NUT=-181	elmozdulás-szög nagyobb vagy egyenlő 180 fokkal
ORICONIO:	interpoláció egy kúppalást-felületen
A7= B7= C7=:	közbenső tájolás(programozás normált vektorként)
ORICURVE XH YH ZH pl. polinomokkal PO[XH]=(xe, x2, x3, x4, x5)	tájolás interpolációja a szerszám második érintőpontja mozgásának megadásával A mindenkori végpontokon kívül kiegészítőleg térgörbék programozhatók polinomokkal.

Megjegyzés

Ha a szerszámtájolás aktív ORIAXES-szel a tájolótengelyekkel lesz interpolálva, akkor a forgásszög pályára vonatkoztat beállítása csak a mondat végén teljesül.

7.5.4 A tájolás lefutásának simítása (ORIPATHS A8=, B8=, C8=)

Az állandó gyorsulású tájolás-változásoknál a kontúron pályamozgások megszakításai, amelyek a kontúr sarkainál léphetnek fel, nem kívánatosak. Ez ebből adódó törést a tájolás lefutásában egy közbenső mondat beszúrásával lehet simítani. A tájolás-változások akkor történnek állandó gyorsulással, ha az áttájolás alatt ORIPATHS is aktív. Ebben a fázisban végre lehet hajtani egy leemelő mozgást a szerszámra.

Gépgyártó

Kérjük vegye figyelembe a gépgyártó tájékoztatásait az esetlegesen előre-definiált gépadatokról és beállítási adatokról, amelyekkel ez a funkció aktiválva lesz.

Gépadattal beállítható, hogyan lesz értelmezve a leemelési vektor:

1. A szerszám koordináta-rendszerben a Z-koordinátát a szerszámirány határozza meg.
2. A munkadarab koordináta-rendszerben a Z-koordinátát az aktív sík határozza meg.

További magyarázatok "Pályára vonatkoztatott tájolás" funkcióhoz lásd
Irodalom:
Különleges funkciók működési kézikönyv, Többtengelyes transzformációk (F2)

Szintaxis

A teljes pályára vonatkoztatott állandó szerszámtájolásokhoz a kontúr egy sarkánál további programozások szükségesek. Ennek a mozgásnak az iránya és úthossza A8=X, B8=Y, C8=Z komponensű vektorral lesz programozva:

```
N... ORIPATHS A8=X B8=Y C8=Z
```

Jelentés

`ORIPATHS`:	pályára vonatkoztatott szerszámtájolás, egy törés a tájolás lefutásában simítva lesz
`A8= B8= C8=`:	irány és úthossz vektor-komponensei
`X, Y, Z`:	leemelő-mozgás szerszámirányban

Megjegyzés

A8, B8, C8 irányvektor programozása

Ha ezen vektor hossza nulla, nincs leemelő-mozgás.

ORIPATHS

A pályára vonatkoztatott szerszámtájolás ORIPATHS-szal lesz aktív. Különben a tájolás a kezdettől a végtájolásig lineáris nagykör-interpolációval lesz végrehajtva.

7.6 A tájolás komprimálása (COMPON, COMPCURV, COMPCAD, COMPSURF)

Az NC programokat, amelyekben egy tájolás transzformáció (TRAORI) aktív és a szerszámtájolás van programozva (bármilyen fajta), lehetséges előre megadott tűrések betartásával komprimálni.

Programozás

Szerszámtájolás

Ha egy tájolási transzformáció (TRAORI) aktív, 5-tengelyes gépeknél a szerszámtájolást a következők szerint (kinematika függetlenül) lehet programozni:

- Az irány**vektor** programozása:
  `A3=<...> B3=<...> C3=<...>`
- Az Euler-**szög** ill. RPY-**szög** programozása:
  `A2=<...> B2=<...> C2=<...>`

A szerszám forgatása

A **6-tengelyes** gépeknél a szerszám tájolása mellett még lehet programozni a szerszám forgatását.

A forgásszög programozása:

```
THETA=<...>
```

Lásd " Szerszámtájolás forgatásai (Oldal 347) ".

Megjegyzés

Az NC-mondatok, amelyekben kiegészítőleg egy forgatás van programozva, csak akkor komprimálhatóak, ha a forgásszög **lineárisan** változik. Vagyis a forgásszögre nem szabad polinomot PO[THT]=(...)-vel programozni.

Egy komprimálható NC-mondat általános formája

Egy komprimálható NC-mondat általános formája a következők szerint nézhet ki:

```
N... X=<...> Y=<...> Z=<...> A3=<...> B3=<...> C3=<...> THETA=<...> F=<...>
```

ill.:

```
N... X=<...> Y=<...> Z=<...> A2=<...> B2=<...> C2=<...> THETA=<...> F=<...>
```

Megjegyzés

A pozíció-értékeket lehet közvetlenül, pl. X90 vagy paraméter-hozzárendeléssel, pl. X=R1*(R2+R3) közvetetten megadni.

A szerszámtájolás programozása körasztal-pozíciókkal

A szerszámtájolást meg lehet adni körasztal-pozíciókkal is a következők szerint:

```
N... X=<...> Y=<...> Z=<...> A=<...> B=<...> C=<...> THETA=<...> F=<...>
```

Ebben az esetben a komprimálás két különböző módon lesz végrehajtva attól függően , hogy egy nagykör-interpoláció lesz végrehajtva vagy nem. Ha nincs nagykör-interpoláció, a komprimált tájolás-változás axiális polinomokkal lesz a szokott módon a körtengelyekre megadva.

Kontúr-pontosság

A beállított kompresszor modustól (MD20482 $MC_COMPRESSOR_MODE) függően a geometria-tengelyekre és a tájolótengelyekre a komprimálásnál vagy a beállított tengely-specifikus tűrések (MD33100 $MA_COMPRESS_POS_TOL) vagy a következőkben beállítási adatokkal beállítható csatorna-specifikus tűrések lesznek hatásosak.

SD42475 $SC_COMPRESS_CONTUR_TOL (maximális kontúreltérés)

SD42476 $SC_COMPRESS_ORI_TOL (maximális szögeltérés szerszámtájolásra)

SD42477 $SC_COMPRESS_ORI_ROT_TOL (maximális szögeltérés a szerszám forgásszögére) (csak 6-tengelyes gépeknél)

Irodalom:
Működési kézikönyv Alapfunkciók; 3 ... 5-tengelyes transzformáció (F2),
fejezet: "Tájolás komprimálása"

Aktiválás / deaktiválás

A kompresszor funkciókat a `COMPON`, `COMPCURV`, `COMPCAD` ill. `COMPSURF` modális G-funkciókkal lehet bekapcsolni.

A kompresszor funkció a `COMPOF`-fal lesz befejezve.

Lásd " NC-mondat kompresszió (COMPON, COMPCURV, COMPCAD, COMPOF) (Oldal 257) ".

Megjegyzés

A tájolási mozgás csak aktív nagykör-interpolációnál lesz komprimálva (vagyis a szerszámtájolás változása a kezdeti és a vég-tájolás által kijelölt síkban történik).

Egy nagykör-interpoláció a következő feltételekkel lesz végrehajtva:

- MD21104 $MC_ORI_IPO_WITH_G_CODE = 0,
  ORIWKS aktív és
  a tájolás vektorokkal van programozva (A3, B3, C3 ill. A2, B2, C2).
- MD21104 $MC_ORI_IPO_WITH_G_CODE = 1 és
  ORIVECT ill. ORIPLANE aktív.
  A szerszámtájolás lehet irányvektorként vagy körasztal-pozíciókkal programozva. Ha egy `ORICONxx` vagy `ORICURVE` G utasítás aktív vagy a tájolási szögre polinomok (`PO[PHI]` és `PO[PSI]`) vannak programozva, nem lesz végrehajtva nagykör-interpoláció.

Példa

A következő programpéldában egy polinommal közelített kör lesz komprimálva. A szerszámtájolás ennek során szinkronban mozog hozzá egy kúppaláston. Bár az egymás után programozott tájolás-változások nem egyenletes lefutásúak, a kompresszor a tájolásra egy sima lefutást generál.

```
Programozás                                Kommentár
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               ; kontúr maximális eltérése: 0.05 mm
$SC_COMPRESS_ORI_TOL=5                     ; tájolás maximális eltérése = 5 fok

TRAORI
COMPCURV                                   ; A mozgás egy poligonból képzett körön tör-
                                           ; ténik. A tájolás ennek során egy kúpon mozog
                                           ; a Z-tengely körül 45 fokos nyílásszöggel
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 Tájolás lefutás simítását be-/kikapcsolni (ORISON, ORISOF)

A "Tájolás lefutás simítása" a munkadarab-programban a G-csoport 61 utasításaival lesz be-/kikapcsolva. Az utasítások modálisan hatnak.

Előfeltételek

- Rendszer 5/6 tengelyes transzformációval.
- COMPCAD kompresszor funkció aktív.

Szintaxis

```
ORISON
...
ORISOF
```

Jelentés

ORISON:	Tájolás lefutás simítását bekapcsolni
ORISOF:	Tájolás lefutás simítását kikapcsolni

Példa

```
Programkód                    Kommentár
...
TRAORI()                      ; Tájolási transzformáció bekapcsolása.
COMPCAD                       ; COMPCAD kompresszor funkció bekapcsolása.
ORISON                        ; Tájolás simítás bekapcsolása.
$SC_ORISON_TOL=1.0            ; Szerszámtájolás maximális szögeltérése = 1,0 fok.
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                        ; Tájolás simítás kikapcsolása.
...
```

A tájolás 90 fokkal az XZ síkban -45-től +45 fokig lesz elforgatva. A tájolás lefutás simításával a tájolás már nem éri el a maximális -45 ill. +45 fokot.

7.8 Kinematikus transzformációk

7.8.1 Homlokoldal-transzformációt bekapcsolni (TRANSMIT)

A homlokoldal-transzformáció (TRANSMIT) a munkadarab-programban vagy szinkron-akcióban a `TRANSMIT` utasítással lesz bekapcsolva.

Szintaxis

```
TRANSMIT
TRANSMIT(<n>)
```

Jelentés

`TRANSMIT`:	TRANSMIT bekapcsolása az első TRANSMIT adatkészlettel
`TRANSMIT(n)`:	TRANSMIT bekapcsolása az <n>-edik TRANSMIT adatkészlettel

Megjegyzés

Csatornában aktív TRANSMIT transzformáció kikapcsolása:

- transzformáció kikapcsolása: `TRAFOOF`
- egy másik transzformáció bekapcsolása: pl. `TRACYL`, `TRAANG`, `TRAORI`

7.8.2 Hengerpalást-transzformációt bekapcsolni (TRACYL)

A hengerpalást-transzformáció (TRACYL) a munkadarab-programban vagy szinkron-akcióban a `TRACYL` utasítással lesz bekapcsolva.

Szintaxis

```
TRACYL(<d>)
TRACYL(<d>,<n>)
TRACYL(<d>,<n>,<k>)
```

Jelentés

`TRACYL(<d>)`:	TRACYL bekapcsolása az első TRACYL adatkészlettel és <d> munkaát-mérővel
`TRACYL (<d>,<n>)`:	TRACYL bekapcsolása az <n>-edik TRACYL adatkészlettel és <d> mun-kaátmérővel
`<d>`:	vonatkozási- ill. munka-átmérő Az érték nagyobb kell legyen 1-nél.

<n>:	TRACYL adatkészlet-szám (opcionális)	
	értéktartomány:	1, 2
<k>:	A <k> paraméter csak az 514-es transzformáció-típusnál számít	
	k = 0:	vájatfal-korrekció nélkül
	k = 1:	vájatfal-korrekcióval
	Ha a paraméter nincs megadva, a paraméterezett alaphelyzet hatásos: $MC_TRACYL_DEFAULT_MODE_<n> <n> = TRACYL adatkészlet szám	

Megjegyzés

Csatornában aktív TRACYL transzformáció kikapcsolása:

- transzformáció kikapcsolása: `TRAFOOF`
- egy másik transzformáció bekapcsolása: pl. `TRAANG`, `TRANSMIT`, `TRAORI`

Példa

Programkód	Kommentár
...	
N40 TRACYL(40.)	; TRACYL bekapcsolása az első TRACYL adatkészlet-tel és 40 mm-es munkaátmérővel.
...	

További információ

Program struktúra

Egy vájat marására szolgáló munkadarab-program az 513-as TRACYL transzformációval (TRACYL vájatfal-korrekcióval) rendszerint a következő lépésekből áll:

1. Szerszám kiválasztása
2. TRACYL kiválasztása.
3. Megfelelő koordináta-eltolás (FRAME) felhívása.
4. Pozícionálás.
5. OFFN programozása.
6. Szerszámsugár-korrekció felhívása.
7. Rámeneteli mondat (szerszámsugár-korrekció megtétele és rámenetel a vájatfalra).
8. Vájat-középvonal kontúrja.
9. Szerszámsugár-korrekció kikapcsolása.
10. Lemeneteli mondat (szerszámsugár-korrekció megtétele visszafelé és elmenetel a vájatfaltól).
11. Pozícionálás.

12. TRAFOOF.

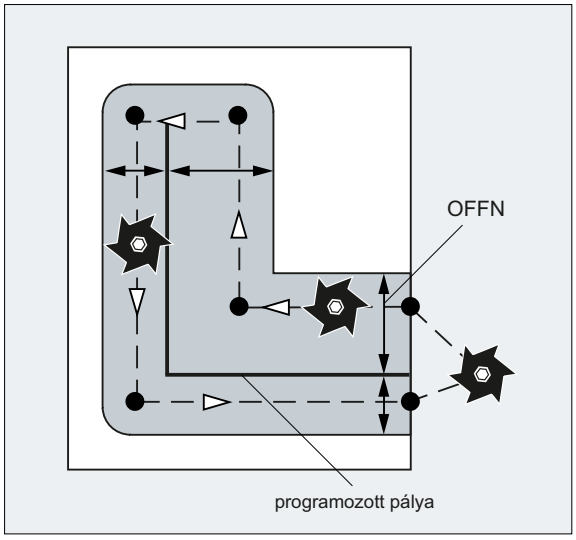
13. Eredeti koordináta-eltolás (FRAME) újra felhívása.

Kontúr-offset (OFFN)

A vájatok marásához az 513-as TRACYL transzformációval a munkadarabprogramban a vájat középvonala az OFFN **fél vájat-szélesség** címmel lesz programozva.

Az OFFN csak felhívott szerszámsugár-korrekcióval lesz hatásos, hogy elkerüljük a vájatfal sérülését.

Az OFFN megváltoztatása a munkadarab-programon belül lehetséges. Ezzel a vájat-középvonalat el lehet tolni a középről:



Megjegyzés

Az OFFN legalább akkora kell legyen, mint a szerszámsugár, hogy kizárjuk a szemközti vájatfal sérülését.

Megjegyzés

Az OFFN a TRACYL-lal másképp hat, mint TRACYL nélkül. Mivel az OFFN aktív szerszámsugár-korrekciónál TRACYL nélkül is be lesz számítva, ezért az OFFN a TRAFOOF után újra nullába kell állítani.

FIGYELEM

Az OFFN hatása a transzformáció típustól függ.

Az 513-as TRACYL transzformáció (TRACYL vájatfal-korrekcióval) esetén az OFFN-ra a fél vájat-szélesség van programozva.

Az 512-es TRACYL transzformáció (TRACYL vájatfal-korrekció nélkül) az OFFN értéke a szerszámsugár-korrekcióra ráhagyásként hat.

Szerszámsugár-korrekció (SZSK)

Az 513-as TRACYL transzformációnál a szerszámsugár-korrekció nem a vájatfalhoz, hanem a vájat programozott közepéhez viszonyítva lesz beszámítva. Ahhoz, hogy a szerszám a vájatfaltól balra haladjon, a G41 helyett a G42-t kell programozni vagy az OFFN értékét negatív előjellel megadni.

Szerszámátmérő

A TRACYL-lal és egy szerszámmal, amelynek átmérője kisebb a vájat-szélességnél, nem ugyanaz a vájat-geometria lesz létrehozva, mint egy szerszámmal, amelynek átmérője egyenlő a vájat-szélességgel. A pontosság javításához ajánlott a szerszám átmérőjét csak kevéssel kisebbre választani a vájat-szélességnél.

Tengelyek használata

Megjegyzés

A következő tengelyeket nem lehet pozícionáló-tengelyként ill. inga-tengelyként használni:

- hengerpalást-felület kerületi irányának geometria-tengelye (Y tengely)
- a kiegészítő lineáris tengely vájatfal-korrekciónál (Z tengely)

7.8.3 Ferdeszög-transzformációt programozható szöggel bekapcsolni (TRAANG)

A ferdeszög-transzformáció a munkadarabprogramban vagy szinkron-akcióban a `TRAANG` utasítással lesz bekapcsolva.

Szintaxis

```
TRAANG
TRAANG()
TRAANG(,<n>)
TRAANG(<α>)
TRAANG(<α>,<n>)
```

Jelentés

`TRAANG`: `TRAANG()`:	TRAANG bekapcsolása az első TRAANG adatkészlettel és az utoljára érvényes <α> szöggel
`TRAANG(,<n>)`:	TRAANG bekapcsolása az <n>-edik TRAANG adatkészlettel és az utoljára érvényes <α> szöggel
`TRAANG(<α>)`:	TRAANG bekapcsolása az első TRAANG adatkészlettel és az <α> szöggel
`TRAANG(<α>,<n>)`:	TRAANG bekapcsolása az <n>-edik TRAANG adatkészlettel és az <α> szöggel

<a>:	ferdén álló tengely szöge (opcionális)	
	értéktartomány:	-90° < α < + 90°
	Ha nincs szög megadva, a gépadatban paraméterezett alaphelyzet hatásos: MD2xxxx $MC_TRAANG_ANGLE_<n>	
<n>:	TRAANG adatkészlet-szám (opcionális)	
	értéktartomány:	1, 2

Megjegyzés

A csatornában aktív `TRAANG` ferdeszög transzformáció kikapcsolása:

- transzformáció kikapcsolása: `TRAFOOF`
- egy másik transzformáció bekapcsolása: pl. `TRACYL`, `TRANSMIT`, `TRAORI`

Példa

Programkód	Kommentár
`N20 TRAANG(45)`	`; TRAANG bekapcsolása az első TRAANG adatkészlettel és az 45°-os szöggel`

7.8.4 Ferde beszúrás köszörűgépeken (G5, G7)

A G7 és G5 G-funkciók a ferde beszúrás programozásának egyszerűsítését szolgálják a köszörűgépeken a "ferde tengely" (TRAANG) transzformációval úgy, hogy a beszúrásnál kizárólag a ferde tengely mozog.

Ennél csak a beszúrási mozgás kívánt X és Y vég-pozícióját kell programozni. A hozzá tartozó kezdő pozíciót G7-nél az NC az X tengely aktuális pozíciójából, a programozott vég-pozícióból és a ferde tengely és veszi fel α szögéből kiszámítja majd felveszi.

A kezdő pozíció a két egyenes metszéspontjából adódik ki:

- Z tengellyel párhuzamos egyenes az X tengely aktuális pozíciójának távolságára
- ferde tengellyel párhuzamos egyenes a programozott vég-pozíción át

Az ezután következő G5-tel a ferde tengely a programozott végpozícióra megy.

Szintaxis

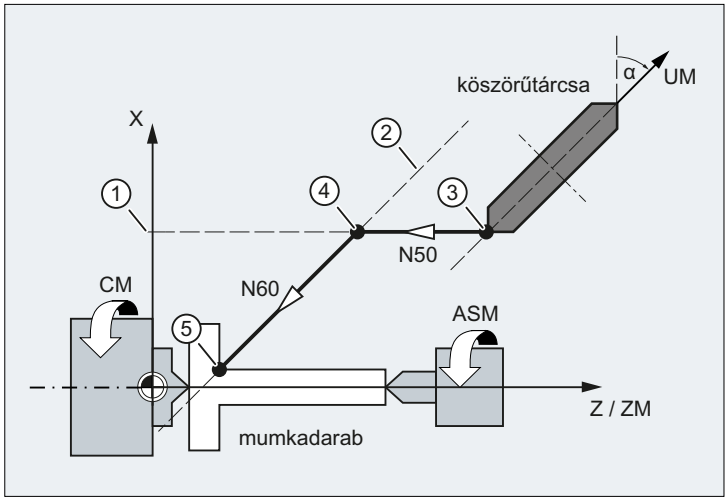
```
G7 <végpoz_X> <végpoz_Z>
G5 <végpoz_X>
```

Jelentés

`G7`:	kezdő pozíciót a ferde beszúráshoz kiszámítani és felvenni
`G5`:	ferde tengely mozgása a programozott vég-pozícióra

<végpoz_X>:	X tengely vég-pozíciója
<végpoz_Z>:	Z tengely vég-pozíciója

Példa



① Z tengellyel párhuzamos az X tengely aktuális pozíciójának távolságára
② ferde tengellyel párhuzamos a programozott vég-pozíción át
③ kiinduló pozíció
④ beszúrás: kezdő pozíció
④ beszúrás: végpozíció
X geometria-tengely
Z geometria-tengely
ZM géptengely
UM géptengely

Programkód	Kommentár
N... G18	; XZ sík választása.
N40 TRAANG(45.0)	; TRAANG transzformációt bekapcsolni, szög = 45°.
N50 G7 X40 Z70 F4000	; Kezdő pozíciót kiszámítani és felvenni.
N60 G5 X40 F100	; Ferde tengelyt vég-pozícióra mozgatni.
N70 ...	

7.9 Láncolt transzformáció bekapcsolása (TRACON)

A láncolt transzformáció a munkadarab-programban vagy szinkron-akcióban a `TRACON` utasítással lesz bekapcsolva.

Szintaxis

```
TRACON(<Trafo_Nr>,<Par_1>,<Par_2,...)
...
TRAFOOF
```

Jelentés

`TRACON`:	láncolt transzformáció bekapcsolása Egy másik előzőleg aktivált transzformáció a `TRACON()` által implicit ki lesz kapcsolva.		
`<Trafo_Nr>`:	láncolt transzformáció száma		
	típus:	INT	
	értéktartomány:	0 ... 2	
	érték:	0, 1	első/egyetlen láncolt transzformáció
		2	második láncolt transzformáció
		nincs adat	ugyanaz, mint 0 ill. 1
		Utalás: 0, 1, 2-től eltérő értékek hibajelzést eredményeznek.	
`<Par_1>,<Par_2,...`:	Paraméterek a láncolt transzformációkhoz (pl. ferde tengely szöge) Nem megadott paramétereknél az előbeállítások vagy az utoljára használt paraméterek lesznek hatásosak. A vessző kitételével kell arról gondoskodni, hogy a megadott paraméterek abban a sorrendben legyenek kiértékelve, amiben várják őket, amikor előtte álló paramétereknél előbeállítások kell hassanak. Különösen legalább egy paraméter megadásánál ezelőtt egy vesszőnek kell állnia, akkor is, ha `<Trafo_Nr>` megadása nem szükséges, tehát pl. `TRACON( , 3.7)`. TRACON-nál TRACYL-lal vagy TRAORI-val (5. tengely megmunkálás) a második Par_2 paraméter nem ferde tengely szögeként TRACON-nál TRACYL-lal (hengerpalást megmunkálás) az egység-átmérő par lesz használva.		
`TRAFOOF`:	Az utoljára bekapcsolt (láncolt) transzformáció ki lesz kapcsolva.		

Példa

Programkód	**Kommentár**
`...`	

Programkód	Kommentár
N230 TRACON(1,45.)	; Első láncolt transzformáció bekapcsolása.
	;Az előzőleg aktív transzformáció automatikusan ki lesz kapcsolva.
	; Paraméter a ferde tengelyre 45°.
...	
N330 TRACON(2,40.)	; Második láncolt transzformáció bekapcsolása.
	; Paraméter a ferde tengelyre 40°.
...	
N380 TRAFOOF	; Második láncolt transzformáció kikapcsolása.
...	

7.10 PTP-mozgás derékszögű koordinátarendszerben

7.10.1 Derékszögű PTP-mozgás be-/kikapcsolása (PTP, PTPG0, PTPWOC, CP)

A derékszögű pont-pont ill. PTP-mozgás az NC programban a 49-es G-csoporthoz tartozó utasításokkal lesznek be-/kikapcsolva.

Az utasítások modálisan hatnak. Elő-beállítás a mozgás egy derékszögű pályamozgással (CP).

A CP-től eltérően az aktív PTP-mozgásnál csak még egy derékszögű célpont transzformáció lesz végrehajtva és a géptengelyek szinkronban lesznek mozgatva.

A derékszögű célpont egyértelmű átszámításához a géptengely értékekre a pozíció és szög adatokon túlmenően még információk szükségesek a tengely helyzetének jellemzéséhez. Ezek a megadások a STAT (Oldal 370) és TU (Oldal 374) beállítható címekkel történnek.

Előfeltétel

TRAORI, TRANSMIT, RCTRA vagy ROBX transzformáció aktív.

Szintaxis

```
PTP / PTPG0 / PTPWOC
...
CP
```

Jelentés

PTP:	PTP pontról pontra mozgás bekapcsolás A programozott derékszögű pozíció a G0 és G1 mondatokban szinkrontengely mozgással lesz felvéve.
PTPG0:	PTPG0 pontról pontra mozgás bekapcsolás Csak a G0 mondatokban lesz a programozott derékszögű pozíció szinkrontengely mozgással felvéve. A G1 mondatokban a pályamozgás CP-re lesz átkapcsolva.
PTPWOC:	PTPWOC pontról pontra mozgás bekapcsolása (csak aktív tájolás-transzformációnál lehetséges) Mint a PTP, de kiegyenlítő mozgások nélkül, amelyeket a kör-tengelyek és a tájolási tengelyek mozgásai okoznak.
CP:	Pontról pontra mozgás kikapcsolás és CP pályamozgás bekapcsolása A CP-vel egy derékszögű pályamozgás lesz végrehajtva.

Megjegyzés

PTPWOC

Egy RCTRA vagy ROBX transzformációval kombinálva a PTPWOC alkalmazásának nincs értelme!

Példák

Lásd:

- Példa 1: Egy 6 tengelyes robot PTP mozgatása ROBX transzformációval (Oldal 377)
- Példa 2: PTP-mozgás általános 5-tengelyes transzformációnál (Oldal 378)
- Példa 3: PTPG0 és TRANSMIT (Oldal 378)

7.10.2 Csukló állását megadni (STAT)

A pozíció megadása derékszögű koordinátákban és a szerszámtájolás megadása nem elég a gép állásának egyértelmű megadásához, mert azonos szerszámtájolásnál több csukló-állás lehetséges. Attól függően, hogy melyik kinematikáról van szó, maximum 8 különböző csuklóállás létezik. Ezek a különböző csuklóállások transzformáció-specifikusak.

A több-értelműség elkerüléséhez a csukló állása a STAT címmel lesz megadva.

Megjegyzés

A vezérlés a programozott STAT értékeket csak a PTP mozgásoknál veszi figyelembe. A CP mozgásoknál nem lesznek figyelembe véve, mivel egy aktív transzformációnál a mozgásnál egy állás váltás általában nem lehetséges. Az aktív CP-vel mozgásnál a helyzet a célponthoz a kezdőponttól lesz átvéve.

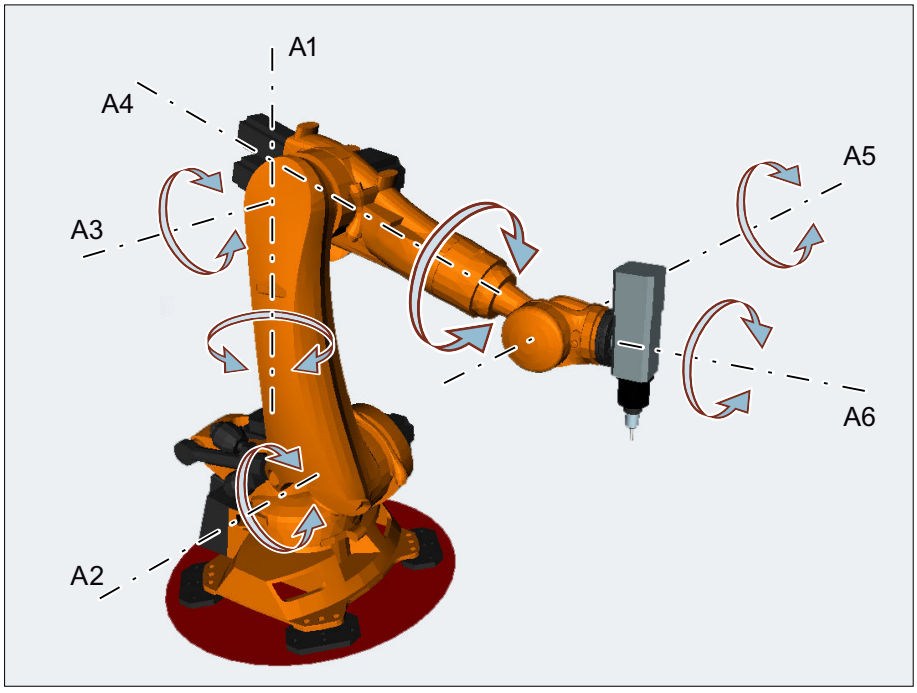
Szintaxis

```
STAT=<érték>
```

Jelentés

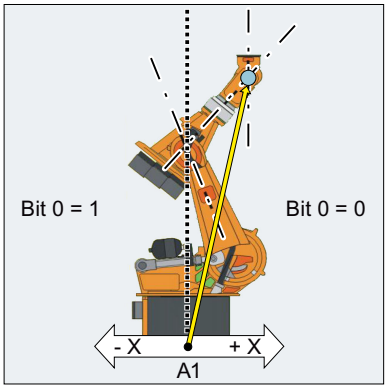
STAT:	beállítható cím a csukló állások megadására
<érték>:	bináris vagy decimális érték Minden lehetséges állás számára tartalmaz egy bitet. A bitek jelentését az adott transzformáció adja meg.

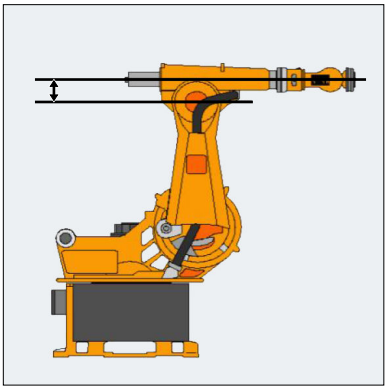
A STAT alkalmazását egy 6 tengelyes csukló-gémes robot maró-orsóval példáján szemléltetjük. A kinematikai transzformáció a ROBX robot transzformációval történjen (Előfeltétel: Az "RMCC/ROBX robot transzformáció bővítve" compile ciklus be van töltve és aktív).



A A1, A2 és A3 tengelyek a csukló-gémes robot főtengelyei. A főtengelyekkel az A4, A5 és A6 tengelyek, amelyeket fej vagy kézi tengelyeknek neveznek, a munkatérben lesznek pozicionálva. A kézi tengelyek további mozgatási lehetőségeivel a maró-orsót a térben úgy lehet tájolni, ahogyan az a megmunkálási feladathoz szükséges. Azonos szerszámtájolásnál különböző csukló-állások lehetségesek.

A megmunkáláshoz használandó csukló-állás kiválasztása a STAT beállítható cím bit 0 ... 2 programozásával történik:

bit 0	kézi tengelyek metszéspontjának pozíciója (A4, A5, A6)		
	= 0	alap-tartomány (**Shoulder Right**) A robot az alap-tartományban van, ha kézi tengelyek metszéspontjának X értéke, az A1 koordinátarendszerre vonatkoztatva, pozitív.	 Példa: A kézi tengelyek metszéspontja az alap-tartományban van
	= 1	fej feletti tartomány (**Shoulder Left**) A robot a fej feletti tartományban van, ha kézi tengelyek metszéspontjának X értéke, az A1 koordinátarendszerre vonatkoztatva, negatív.	

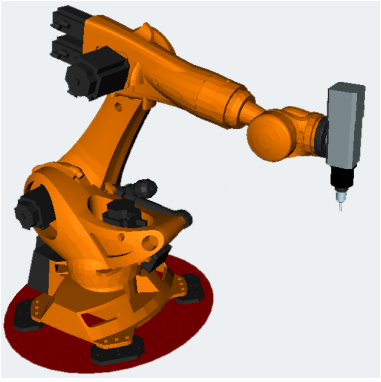
bit 1	tengely 3 pozíciója Az érték, amelynél a bit 1 értéke változik, a robot típusától függ. A robotokra, amelyek 3 és 4 tengelyei metszik egymást, érvényes:		
	= 0	A3 < 0° (**Elbow Down**)	
	= 1	A3 ≥ 0° (**Elbow Up**)	
	Utalás: Robotoknál, a 3 és 4 tengelyek között offset-tel, a szög, amelynél a bit 1 értéke változik, az offset nagyságától függ.		 offset A3 és A4 között
bit 2	tengely 5 pozíciója		
	= 0	A5 ≤ 0° (**No Handflip**)	
	= 1	A5 > 0° (**Handflip**)	

Programpélda:

Programkód	**Kommentár**
`...`	
`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`	`; A STAT érték meghatározza a csukló helyzeteket (lásd lent).`
`...`	

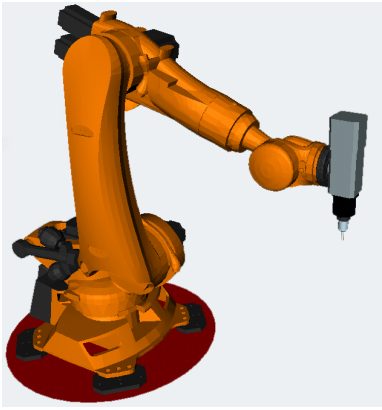
`STAT=1 ('B001')`

→ Shoulder Left
→ Elbow Down
→ No Handflip



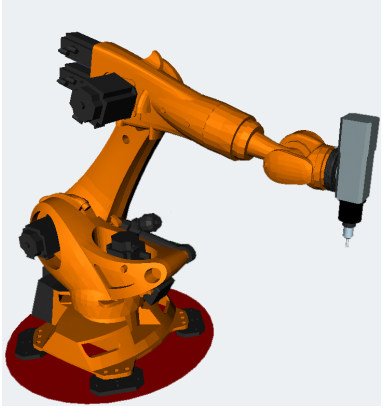
`STAT=2 ('B010')`

→ Shoulder Right
→ Elbow Up
→ No Handflip



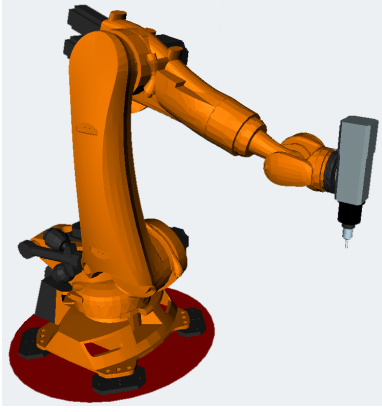
`STAT=5 ('B101')`

→ Shoulder Left
→ Elbow Down
→ Handflip



`STAT=6 ('B110')`

→ Shoulder Right
→ Elbow Up
→ Handflip



TRANSMIT

TRANSMIT-nál a STAT cím lesz használva a pólus több-értelműségének a feloldásához.

Ha a körtengely 180º-kal el kell forduljon ill. a kontúr CP-nél a póluson át vezetne:

bit 0	Csak $MC_TRANSMIT_POLE_SIDE_FIX_1/2 = 1 ill. 2 esetén számít:	
	= 0	Körtengely +180º-ra elmozdul ill. az órajárás irányában forog.
	= 1	Körtengely -180º-ra elmozdul ill. az órajárás irányával szemben forog.
bit 1	Csak $MC_TRANSMIT_POLE_SIDE_FIX_1/2 = 0 esetén számít:	
	= 0	Mozgás a póluson keresztül. A körtengely nem forog.
	= 1	Forgás a pólus körül. Ennél a STAT bit 0 számít.

7.10.3 Tengelyszög előjelét megadni (TU)

Forgó tengelyeknél a +180°-nál nagyobb vagy a -180°-nál kisebb tengely-szögek eléréséhez külön mozgási stratégia (pl. közbenső pontok) nélkül, a TU beállítható címmel meg kell előre adni a tengely-szög előjelét.

Megjegyzés

A vezérlés a programozott TU értékeket csak a PTP mozgásoknál veszi figyelembe. Ezek a CP mozgásoknál nem lesznek figyelembe véve.

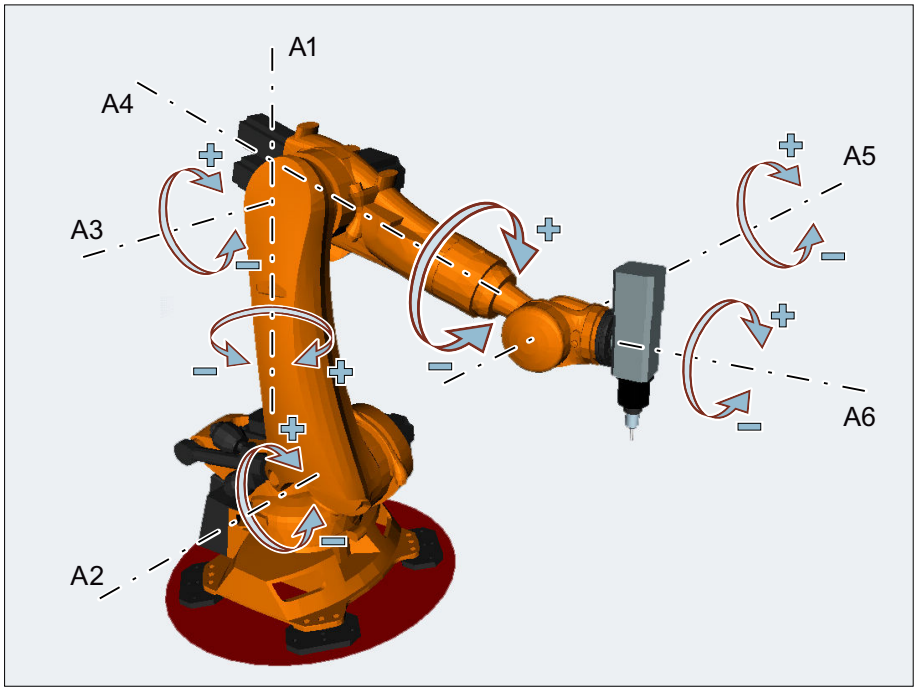
Szintaxis

```
TU=<érték>
```

Jelentés

TU:	Beállítható cím a tengely-szög előjel megadására			
<érték>:	bináris vagy decimális érték Minden, a transzformációban résztvevő tengelyre van egy bit, a melyik kijelzi a tengely-szög (θ) és ezzel a mozgásirányt.			
	bit	= 0	tengely-szög előjel: +	tengely-szögtartomány: 0° ≤ θ < 360°
		= 1	tengely-szög előjel: -	tengely-szögtartomány: - 360° < θ < 0°

Példa: 6 tengelyes csuklós karú robot



Bit	Jelentés	érték	tengely-szög előjel	Tengelyszög
bit 0 [1]	A1 tengelyszög előjele	= 0	+	≥ 0°
		= 1	-	< 0°
bit 1 [1]	A2 tengelyszög előjele	= 0	+	≥ 0°
		= 1	-	< 0°
bit 2 [1]	A3 tengelyszög előjele	= 0	+	≥ 0°
		= 1	-	< 0°
bit 3 [1]	A4 tengelyszög előjele	= 0	+	≥ 0°
		= 1	-	< 0°
bit 4 [1]	A5 tengelyszög előjele	= 0	+	≥ 0°
		= 1	-	< 0°
bit 5 [1]	A6 tengelyszög előjele	= 0	+	≥ 0°
		= 1	-	< 0°

[1] A tényleges TU bit-számok a robot tengelyek csatornatengely számaiból adódnak. A példában a robot tengelyek (A1 ... A6) az első hat tengely a csatornában, tehát a TU bit 0 ... 5 lesznek használva. A robot tengelyek más csatorna-hozzárendelésénél a robot tengelyek TU bit-számai megfelelően változnak (pl.: robot tengelyek a 3. .. 8. csatorna-tengely, vagyis a TU bit 2 ... 7 lesznek használva a robot tengelyekhez).

TU=19 (TU='B010011) jelentése:

Bit	Érték		Tengelyszög
0	= 1	⇒	$\theta_{A1} < 0°$
1	= 1	⇒	$\theta_{A2} < 0°$

2	= 0	⇒	$\theta_{A3} \geq 0°$
3	= 0	⇒	$\theta_{A4} \geq 0°$
4	= 1	⇒	$\theta_{A5} < 0°$
5	= 0	⇒	$\theta_{A6} \geq 0°$

Megjegyzés

A tengelyeknél > ±360°-nál nagyobb mozgási tartománnyal a mozgás mindig a legrövidebb úton történik, mert a tengely állása a TU információkkal nem határozható meg egyértelműen.

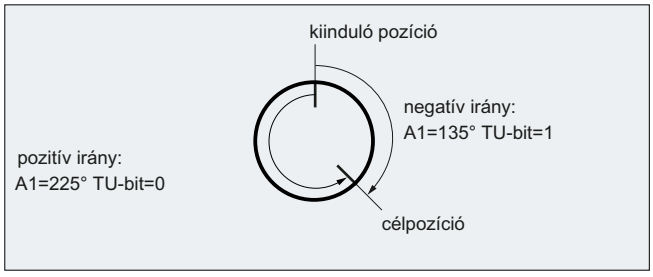
Ha egy pozíciónál nincs TU programozva, az MD30455 $MA_MISC_FUNCTION_MASK-tól függően rövidebb vagy hosszabb út lesz megtéve (lásd "Szoftver határok figyelembe vétele PTP mozgásnál" fejezetet a Bővítő funkciók működési kézikönyvben).

TRANSMIT

PTP mozgásnál aktív TRANSMIT-tal a TU címnek nincs jelentése!

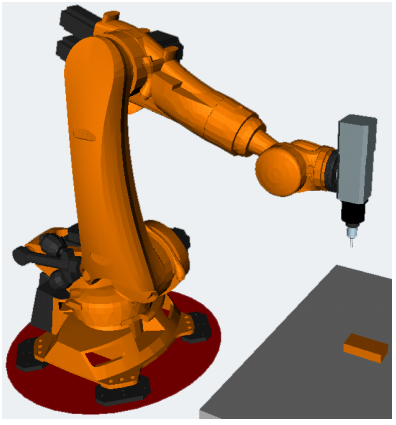
Példa

A képen megadott körtengely pozícióra negatív vagy pozitív irányba lehet rámenetelt végrehajtani. A szög pozíció az A1 cím alatt lesz programozva. Csak a TU megadással lesz a mozgásirány egyértelmű.



7.10.4 Példa 1: Egy 6 tengelyes robot PTP mozgatása ROBX transzformációval

A következő alkalmazási példában a derékszögű PTP mozgás és az azzal összefüggésben levő NC utasítások példaként lesznek megmutatva.



Kép 7-3 6 tengelyes csuklós karú robot maró-orsóval

```
N1 G90
N2 T="T8MILLD20" D1 M6
N3 TRAORI
;$P_UIFR[1]=CTRANS(X,1500,Y,0,Z,400):CROT(X,0,Y,0,Z,-90)
N4 G54
N5 M3 S20000
N6 ORIWKS
N7 ORIVIRT1
N8 CYCLE832(0.01,_FINISH,1)
;HOME
N9 TRAFOOF
N10 G0 RA1=0.0000 RA2=-90.0000 RA3=90.0000 A=0.0000 B=90.0000 C=0.0000
N11 TRAORI
N12 G54
N13 G0 PTP X1369.2426 Y956.7528 Z502.5517 A=135.5761 B=-33.2223 C=161.1435
STAT='B010' TU='B001011'
N14 G0 X1355.1242 Y1014.9394 Z424.9695 A=135.8491 B=-33.1439 C=160.9941
STAT='B010' TU='B001011'
N15 G1 CP X1354.8361 Y1016.1269 Z423.3862 A=136.0635 B=-33.0819 C=160.8770
F1000
N16 G1 X1336.4283 Y1016.1269 Z426.6311 A=136.0484 B=-32.2151 C=160.9643
F2000
N17 G1 X1317.9831 Y1016.1269 Z429.6730 A=136.0175 B=-31.3394 C=161.0655
;HOME
N18 TRAFOOF
N19 G0 RA1=0.0000 RA2=-90.0000 RA3=90.0000 A=0.0000 B=90.0000 C=0.0000
N20 M30
```

7.10.5 Példa 2: PTP-mozgás általános 5-tengelyes transzformációnál

Feltétel: Az alap egy derékszögű CA kinematika.

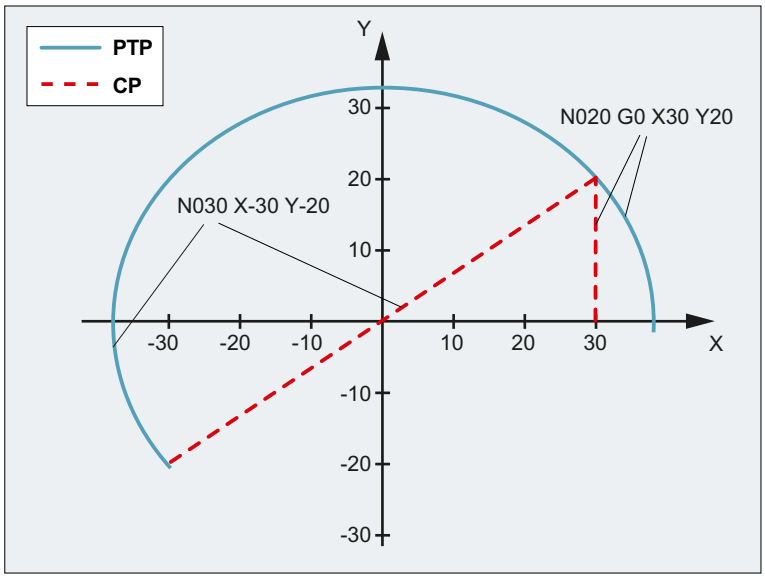
Programkód	Kommentár
TRAORI	; CA kinematika transzformáció be
PTP	; PTP-mozgás bekapcsolása
N10 A3=0 B3=0 C3=1	; körtengely pozíciók C=0 A=0
N20 A3=1 B3=0 C3=1	; körtengely pozíciók C=90 A=45
N30 A3=1 B3=0 C3=0	; körtengely pozíciók C=90 A=90
N40 A3=1 B3=0 C3=1 **STAT=1**	; körtengely pozíciók C=270 A=-45

A körtengely-pozíciók egyértelmű rámenti helyét kiválasztani:

Az N40 mondatban a körtengelyek a **STAT=1** programozása miatt a hosszabb úton mennek a kezdőpontjukból (C=90, A=90) a végpontjukhoz (C=270, A=–45). **STAT=0** estén a körtengelyek a legrövidebb úton mennének a végponthoz (C=90, A=45).

7.10.6 Példa 3: PTPG0 és TRANSMIT

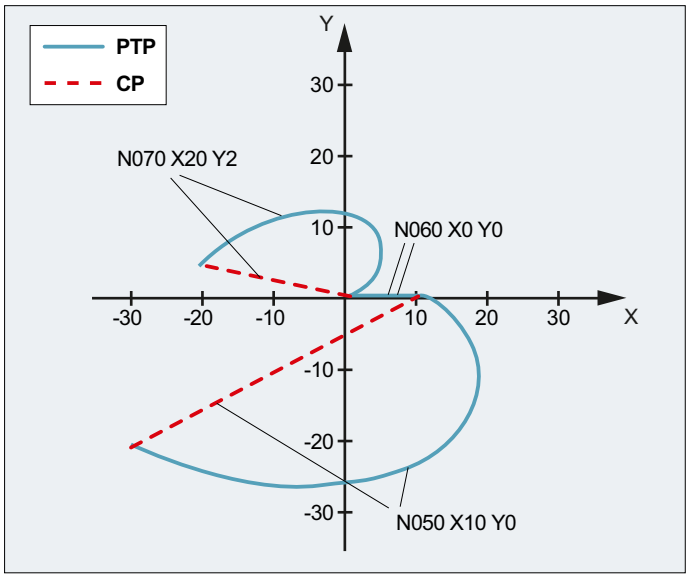
Pólus megkerülése PTPG0 és TRANSMIT esetén



Programkód	Kommentár
N001 G0 X30 Z0 F10000 T1 D1 G90	; kiinduló helyzet abszolút méret
N002 SPOS=0	
N003 **TRANSMIT**	; TRANSMIT transzformáció
N010 **PTPG0**	; minden G0 mondathoz automatikus PTP és utána ismét CP
N020 G0 X30 Y20	

Programkód	Kommentár
N030 X-30 Y-20	
N120 G1 X30 Y20	
N110 X30 Y0	
M30	

Pólusból kijövetel PTPG0 és TRANSMIT esetén



Programozás	Kommentár
N001 G0 X90 Z0 F10000 T1 D1 G90	; kiinduló állapot
N002 SPOS=0	
N003 **TRANSMIT**	; TRANSMIT transzformáció
N010 **PTPG0**	; minden G0 mondathoz automatikus PTP és utána ismét 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 Peremfeltételek egy transzformáció felhívásánál

Funkció

Transzformációk felhívása munkadarab-program, ill. MDA által lehetséges. Ennél a következőket kell figyelembe venni:

- Egy mozgás-közbenső mondat nem lesz beiktatva (letörés/sugár).
- Egy Spline-mondat sorrendnek lezárva kell lennie; ha nem, akkor megjelenik egy jelentés.
- Szerszám-finomkorrekciónak kikapcsolva kell lennie (FTOCOF); ha nem, akkor megjelenik egy jelentés.
- Szerszámsugár-korrekciónak kikapcsolva kell lennie (G40); ha nem, akkor megjelenik egy jelentés.
- Egy aktivált szerszámhossz-korrekciót a vezérlés átvesz a transzformációba.
- A transzformáció előtt hatásos aktuális frame-t a vezérlés kikapcsolja.
- Egy aktív munkamező-behatárolást a vezérlés kikapcsolja a transzformáció által érintett tengelyek számára (megfelelője WALIMOF).
- A védelmi tartomány felügyelete ki lesz kapcsolva.
- Pályavezérlés-üzem és átmenet-simítás meg lesznek szakítva.
- Minden, a gépadatban megadott tengelynek mondatra vonatkoztatva kell szinkronizálva lennie.
- Kicserélt tengelyek vissza lesznek cserélve; ha nem, akkor megjelenik egy jelentés.
- Függő tengelyeknél egy jelentés lesz kiadva.

Szerszámcsere

Egy szerszámcsere csak kikapcsolt szerszámsugár-korrekciónál megengedett.

Egy szerszámhossz-korrekció váltás és a szerszámsugár-korrekció ki-/bekapcsolása nem lehetnek ugyanabban a mondatban programozva.

Frame váltás

Minden utasítás, ami csak az alap-koordinátarendszerre vonatkozik, megengedett (FRAME, szerszámsugár-korrekció). Egy frame-csere G91-nél (láncméret) azonban – másképp, mint inaktív transzformációnál – nem lesz külön kezelve. A megteendő inkremens az új frame munkadarab-koordinátarendszerében lesz kiértékelve – függetlenül attól, hogy melyik frame hatott az előző mondatban.

Kizárások

A transzformációban érintett tengelyeket nem lehet használni:

- Preset-tengelyként (vészjelzés),
- fixpontra menetelre (vészjelzés),
- referálásra (vészjelzés).

7.12 Transzformáció kikapcsolása (TRAFOOF)

Az előre definiált TRAFOOF utasítással minden aktív transzformáció és frame ki lesz kapcsolva.

Megjegyzés

Transzformáció kikapcsolásánál ugyanazok a peremfeltételek (Oldal 380) érvényesek, mint a felhívásra.

Ezután szükséges frame-ket újabb programozással kell aktivizálni.

Szintaxis

```
...
TRAFOOF
```

Jelentés

TRAFOOF:	Összes aktív transzformáció/frame kikapcsolása

Kinematikai láncok 8

8.1 Komponensek törlése (DELOBJ)

A `DELOBJ()` funkció komponenseket "töröl" a hozzárendelt rendszerváltozók visszaállításával az alapértékükre:

- kinematikai láncok elemei
- védőtartományok, védőtartomány elemek és ütközés párok
- transzformáció adatok

Szintaxis

```
[<RetVal>=] DELOBJ(<CompType>[,,,<NoAlarm>)])
[<RetVal>=] DELOBJ(<CompType>,<Index1>[,,<NoAlarm>])
[<RetVal>=] DELOBJ(<CompType>[,<Index1>][,<Index2>][,<NoAlarm>])
```

Jelentés

DELOBJ:	Kinematikai láncok elemei, védelmi tartományok és védelmi tartomány elemei, ütközési párok és transzformációs adatok törlése	
<CompType>:	törlendő komponensek típusa	
	adattípus:	STRING
	Érték: "KIN_CHAIN_ELEM" **Jelentés**: összes kinematikai elem rendszerváltozói: $NK_...	
	Érték: "KIN_CHAIN_SWITCH" **Jelentés**: $NK_SWITCH[<i>] rendszerváltozó	
	Érték: "KIN_CHAIN_ALL" **Jelentés**: összes kinematikai elem és kapcsoló Azonos jelentésű a DELOBJ felhívásával "KIN_CHAIN_ELEM" és "KIN_CHAIN_SWITCH"-csel	
	Érték: "PROT_AREA" **Jelentés**: védelmi tartományok rendszerváltozói: • $NP_PROT_NAME • $NP_CHAIN_NAME • $NP_CHAIN_ELEM • $NP_1ST_PROT	
	Érték: "PROT_AREA_ELEM" **Jelentés**: gép védőtartományok védőtartomány elemeinek rendszerváltozói és/vagy automatikus szerszám-védőtartományok: • $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	
	Érték: "PROT_AREA_COLL_PAIRS" **Jelentés**: ütközés párok rendszerváltozói: • $NP_COLL_PAIR • $NP_SAFETY_DIST	
	Érték: "PROT_AREA_ALL" **Jelentés**: összes védőtartomány, védőtartomány elem és ütközés pár ($NP_... rendszerváltozó) Azonos jelentésű a DELOBJ felhívásával "PROT_AREA", "PROT_AREA_ELEM" und "PROT_AREA_COLL_PAIRS"-szel	
	Érték: "TRAFO_DATA" **Jelentés**: összes transzformáció rendszerváltozói $NT_...	

<Index1>:	az első törlendő komponens indexe (**opcionális**)	
	adattípus:	INT
	alapérték:	-1
	értéktartomány:	-1 ≤ x ≤ (létrehozott komponensek maximális száma -1)
	Érték	**Jelentés**
	0, 1, 2	törlendő komponens indexe
	-1	A megadott típus összes komponense törölve lesz. <Index2> nem lesz kiértékelve.
<Index2>:	az utolsó törlendő komponens indexe (**opcionális**) Ha <Index2> nincs programozva, csak a <Index1>-ben hivatkozott komponensek rendszerváltozói lesznek törölve.	
	adattípus:	INT
	alapérték:	Csak a <Index1>-ben hivatkozott komponensek rendszerváltozói lesznek törölve.
	értéktartomány:	<Index1> < x ≤ (létrehozott komponensek maximális száma -1)
<NoAlarm>:	vészjelzés elnyomása (**opcionális**)	
	adattípus:	BOOL
	alapérték:	FALSE
	Érték	**Jelentés**
	FALSE	Hiba esetén (<RetVal> < 0) a program feldolgozása megszakad és egy vészjelzés lesz kijelezve .
	TRUE	Hiba esetén a program feldolgozása nem szakad meg és vészjelzés nem lesz kijelezve. Alkalmazási eset: alkalmazás-specifikus reakciónak megfelelő visszaadási érték
<RetVal>:	funkció visszaadási érték	
	adattípus:	INT
	értéktartomány:	0, -1, -2, ... -7
	Érték	**Jelentés**
	0	Nem lépett fel hiba.
	-1	A funkció felhívása paraméter nélkül. Legalább a <CompType> paramétert meg kell adni.
	-2	<CompType> ismeretlen komponenst jelöl
	-3	<Index1> kisebb mint -1
	-4	<Index1> nagyobb mint a komponensek beállított száma
	-5	<Index1> egy komponens **csoport** törlésénél egy érték nem egyenlő -1-gyel
	-6	<Index2> kisebb mint <Index1>
	-7	<Index2> nagyobb mint a komponensek beállított száma

8.2 Index megállapítása névvel (NAMETOINT)

A STRING típusú rendszerváltozó mezőkbe alkalmazás-specifikus nevek vannak bevive. A rendszerváltozó jelölők és nevek alapján a `NAMETOINT()` funkció megállapítja a nevekhez tartozó index értékeket, amelyek alatt a a rendszerváltozó mező tárolva van.

Szintaxis

```
<RetVal> = NAMETOINT(<SysVar>,<Name>[,<NoAlarm>])
```

Jelentés

NAMETOINT:	rendszerváltozó index megállapítása	
<SysVar>:	STRING típusú rendszerváltozó mező neve	
	adattípus:	STRING
	értéktartomány:	Az NC összes STRING típusú rendszerváltozó mezőinek nevei
<Name>:	Karakterlánc vagy név, amihez meg kell állapítani a rendszerváltozó indexet.	
	adattípus:	STRING
<NoAlarm>:	vészjelzés elnyomása (**opcionális**)	
	adattípus:	BOOL
	alapérték:	FALSE
	Érték	**Jelentés**
	TRUE	Hiba esetén a program feldolgozása nem szakad meg és vészjelzés nem lesz kijelezve. Alkalmazási eset: alkalmazás-specifikus reakciónak megfelelő visszaadási érték
	FALSE	Hiba esetén (<RetVal> < 0) a program feldolgozása megszakad és egy vészjelzés lesz kijelezve.
<RetVal>:	rendszerváltozó index vagy hiba-jelentés	
	adattípus:	INT
	értéktartomány:	-1 ≤ x ≤ (létrehozott komponensek max. száma -1)
	Érték	**Jelentés**
	≥ 0	A keresett név a megadott rendszerváltozó index alatt lett megtalálva.
	-1	A keresett név nem található ill. egy hiba lépett fel.

Példa

Programkód	**Kommentár**
`DEF INT INDEX`	
`$NP_PROT_NAME[27] = "Abdeckung"`	
`...`	
`INDEX = NAMETOINT("$NP_PROT_NAME","Abdeckung")`	`; INDEX == 27`

Ütközés elkerülése kinematikai láncokkal 9

Megjegyzés

Védőtartományok

A következő fejezetekben megnevezett védelmi tartományok a "Geometriai gép-modellezés" funkcióra vonatkoznak

Irodalom:
Különleges funkciók védelmi tartomány, "Geometriai gép-modellezés" fejezet

9.1 Ütközés párok vizsgálata (COLLPAIR)

A `COLLPAIR(...)` funkció megállapítja, hogy két védelmi tartomány egy ütközési párt képez-e.

Szintaxis

```
[<RetVal> =] COLLPAIR(<Name_1>,<Name_2>[,<NoAlarm>)])
```

Jelentés

`COLLPAIR`:	Egy ütközés pár összetartozásának vizsgálata		
`<RetVal>`:	funkció visszaadási érték		
	adattípus:	INT	
	érték:	≥ 0	A két védelmi tartomány egy ütközés párt képez. visszaadási érték == ütközés pár index m (lásd $NP_COLL_PAIR)
		-1	Kettőnél kevesebb string lett megadva, vagy a két stringből legalább egy nulla-string.
		-2	Az első paraméterben megadott védelmi tartomány nem található.
		-3	A második paraméterben megadott védelmi tartomány nem található.
		-4	Egyik megadott védelmi tartomány sem található.
		-5	Mindkét megadott védelmi tartomány meg lett találva, de nem együtt egy ütközés párban.
`<Name_1>`:	első védelmi tartomány neve		
	adattípus:	STRING	
	értéktartomány:	paraméterezett védelmi tartomány nevek	
`<Name_2>`:	második védelmi tartomány neve		
	adattípus:	STRING	
	értéktartomány:	paraméterezett védelmi tartomány nevek	
`<NoAlarm>`:	Vészjelzés elnyomása (**opcionális**)		
	adattípus:	BOOL	
	érték:	FALSE (alapérték)	Hiba esetén <RetVal> < 0) a program feldolgozása megszakad és egy vészjelzés lesz kijelezve.
		TRUE	Hiba esetén a program feldolgozása nem szakad meg és vészjelzés nem lesz kijelezve. Alkalmazási eset: alkalmazás-specifikus reakciónak megfelelő visszaadási érték

9.2 Ütközés elkerülés gép-modell újra számítását igényelni (PROTA)

Ha az $NK_... kinematikai lánc, a geometriai gép-modellezés vagy az $NP_... ütközés elkerülés rendszerváltozói a munkadarabprogramban vannak beírva, utána fel kell hívni a `PROTA` eljárást, amivel az ütközés elkerülés NC-beni gép-modellje hatásos lesz.

Szintaxis

```
PROTA[(<Par>)]
```

Jelentés

`PROTA`:	ütközés elkerülés gép-modell újra számítását igényelni		
	• Előrefutás-álljt vált ki. • Egyedül kell álljon a mondatban.		
`<Par>`:	Paraméterek (**opcionális**)		
	adattípus:	STRING	
	érték:	---	Paraméterek nélkül. A gép modell újra számítása végre lesz hajtva. A védelmi tartományok állapotai megmaradnak.
		"R"	A gép modell újra számítása végre lesz hajtva. A védelmi tartományok a kiindulási állapotban a $NP_INIT_STAT-nak megfelelően lesz beállítva.

Peremfeltételek

Szimuláció

A `PROTA` eljárást nem szabad a munkadarabprogramokban a (simNC) szimulációval összefüggésben használni.

Példa: `PROTA`, felhívásának elkerülése aktív szimuláció közben.

```
Programkód                    Kommentár
...
IF $P_SIM == FALSE            ; IF szimuláció nem aktív
   PROTA                      ;  THEN ütközésmodellt újra számítani
ENDIF                         ; ENDIF
...
```

Lásd még

Védelmi tartomány állapotot beállítani (PROTS) (Oldal 390)

9.3 Védelmi tartomány állapotot beállítani (PROTS)

A `PROTS(...)` eljárás beállítja a védelmi tartományok állapotát a megadott értékre.

Szintaxis

```
PROTS(<State>[,<Name_1>,...,<Name_n>])
```

Jelentés

`PROTS`:	védelmi tartományok állapotát beállítani		
	• Egyedül kell álljon a mondatban.		
`<State>`:	állapot, amire a megadott védelmi tartományt be kell állítani		
	adattípus:	CHAR	
	érték:	"A" vagy "a"	Állapot Aktív
		"I" vagy"i"	Állapot nem aktív
		"P" vagy "p"	Állapot előre aktivált ill. PLC vezérelt [1]
		"R" vagy "r"	Állapot inicializálási állapot NC belső értéke[2]
`<Name_1> ... <Name_n>`:	egy vagy több védelmi tartomány neve, amelyeket a megadott állapotba kell állítani (**opcionális**) Ha nincs megadva név, a megadott állapot az összes definiált védelmi tartományra be lesz állítva.		
	adattípus:	STRING	
	értéktartomány:	paraméterezett védelmi tartomány nevek	
	Utalás: A védelmi tartományok maximális száma, amit paraméterként meg lehet adni, függ a programsoronként maximálisan lehetséges karakterek számától.		
[1]Aktiválás / deaktiválás a következőn keresztül történik: DB10.DBX234.0 - DBX241.7 [2] Az állapot az inicializálási állapot NC-n belüli értékére lesz beállítva, vagyis az értékre, ami az $NP_INIT_STAT rendszerváltozónak a PROTA(...) (Oldal 389) utolsó felhívásának időpontjában volt.			

9.4 Két védelmi tartomány távolságának meghatározása (PROTD)

A `PROTD(...)` funkció kiszámítja két védelmi tartomány távolságát.

Funkció tulajdonságok:

- A távolság kiszámítása a védelmi tartományok állapotától (aktivált, nem aktivált, előre aktivált) függetlenül történik.
- Két védelmi tartomány távolságának kiszámításához csak olyan védelmi tartomány elemek lesznek figyelembe véve, amelyek $NP_USAGE = "C" vagy "A"-val jelöltek. A védelmi tartomány védelmi tartomány elemei, amelye $NP_USAGE = "V"-vel van megjelölve, nem lesznek figyelembe véve.
- A védelmi tartományokat, amelyeknél a védelmi tartomány összes védelmi tartomány eleme $NP_USAGE = "V"-vel van megjelölve, nem lehet bevonni a távolság számításba.
- A távolság kiszámítása az előző mondat végén érvényes pozíciókkal történik.
- Átfedések, amelyeket a főfutamban be kell számítani pl. DRF-eltolás vagy külső nullaponteltolás a funkció **értelmezési időpontjában** érvényes értékeikkel lesznek figyelembe véve a távolság számításában.

Megjegyzés

Szinkronizáció

A `PROTD(...)` funkció alkalmazásánál kizárólag a felhasználó felelőssége a főfutam és az előre-futás szinkronizálása esetlegesen `STOPRE` előre-futás állj-jal.

Ütközés

Ha a megadott védelmi tartományok között ütközés van, a funkció a 0,0 távolságot adja. Ütközés akkor van, ha a két védelmi tartomány érinti vagy metszi egymást.

Az ütközés vizsgálat biztonsági távolsága (MD10622 $MN_COLLISION_SAFETY_DIST) a távolság számításánál nem lesz figyelembe véve.

Szintaxis

```
[<RetVal> =] PROTD([<Name_1>],[<Name_2>],VAR <Vector>[,<System>])
```

Jelentés

`PROTD`:	A két megadott védelmi tartomány távolságát kiszámítani	
	• Egyedül kell álljon a mondatban.	
`<RetVal>`:	funkció visszaadási érték: a két védelmi tartomány távolságának abszolút értéke vagy 0,0 ütközésnél (lásd fent: ütközés bekezdés)	
	adattípus:	REAL
	értéktartomány:	0,0 ≤ x ≤ +max. REAL érték

`<Name_1>`, `<Name_2>`:	a két védelmi tartomány neve, amelyek távolságát ki kell számítani (**opcionális**)		
	adattípus:	STRING	
	értéktartomány:	paraméterezett védelmi tartomány nevek	
	alapérték:	"" (üres-string) Ha nincsenek megadva védelmi tartományok, a funkció az aktuális legkisebb távolságot számítja ki az ütközés modellben levő összes aktivált és előre aktivált védelmi tartományból.	
`<Vector>`:	visszaadási érték 3-dimenziós távolság vektor a <Name_2> védelmi tartománytól a <Name_1> védelmi tartományhoz a következőkkel: • <Vector>[0]: X koordináta a világ koordinátarendszerben • <Vector>[1]: Y koordináta a világ koordinátarendszerben • <Vector>[2]: Z koordináta a világ koordinátarendszerben ütközésnél: <Vector> == nulla-vektor		
	adattípus:	VAR REAL [3]	
	értéktartomány:	<Vector> [n]: 0,0 ≤ x ≤ ±max. REAL-érték	
`<System>`:	távolság és távolság vektor mértékrendszere (hüvelyk / metrikus) (**opcionális**)		
	adattípus:	BOOL	
	érték:	FALSE (alapérték)	A mértékrendszer megfelel a G-csoport 13 (G70, G71, G700, G710) aktuálisan aktív G utasításának.
		TRUE	Mértékrendszer a beállított alaprendszernek megfelelően: MD52806 $MN_ISO_SCALING_SYSTEM

Transzformációk kinematikai láncokkal 10

10.1 Transzformáció bekapcsolása (TRAFOON)

Egy kinematikai láncokkal definiált transzformáció az előre definiált TRAFOON eljárással lesz aktiválva. A felhívásnak egyedül kell állnia egy mondatban.

Megjegyzés

Alternatívaként egy kinematikai láncokkal definiált transzformációt lehet a hagyományos nyelvi utasításokkal, mint pl. TRAORI vagy TRANSMIT, aktiválni. Ehhez a $NT_TRAFO_INDEX rendszerváltozóba egy nullától eltérő megfelelő érték kell legyen beadva.

További információkat a $NT_TRAFO_INDEX-hez lásd "Jegyzékes kézikönyv Rendszerváltozók".

Szintaxis

```
TRAFOON(<Trafoname>,<Diameter>,<k>)
```

Jelentés

TRAFOON:	eljárás a kinematikai láncokkal definiált transzformációt bekapcsolására		
`<Trafoname>`:	transzformációs adatkészlet neve		
	adattípus:	STRING	
	értéktartomány:	transzformációs adatkészletek összes $NT_NAME által definiált neve	
	Utalás: A transzformációs adatkészlet neve egyértelmű kell legyen. Az $NT_NAME-ben csak egyszer fordulhat elő.		
`<Diameter>`:	vonatkozási- ill. munka-átmérő (csak TRACYL)		
	adattípus:	REAL	
	Az érték > 1 kell legyen.		
`<k>`:	A vájatfal-korrekció alkalmazását definiálja (csak TRACYL).		
	adattípus:	BOOL	
	érték:	FALSE	vájatfal-korrekció nélkül
		TRUE	vájatfal-korrekcióval
	Megfelel az 514-es TRACYL transzformáció típusnak (programozható vájatfal korrekció). Ha <k> nincs megadva, a $NT_CNTRL[<n>] bit 10-ben paraméterezett beállítás hat.		

Példa

Programkód	Kommentár
`TRAFOON["Trans_1"]`	`Aktiválva a Trans_1 nevű transzformáció.`

10.2 Tájolási transzformáció a gép bemérése után módosítani (CORRTRAFO)

A kinematikai láncokkal definiált tájolási transzformációs gépeknél a felhasználó használhatja a CORRTRAFO előre definiált funkciót, hogy egy gépi bemérés után az offset-vektorokat vagy a tájolótengelyek irányvektorait a gép kinematikai modelljében módosítsa.

Szintaxis

```
<Corr_Status> = CORRTRAFO(<Corr_Vect>, <Corr_Index>, <Corr_Mode> [,
<No_Alarm>])
```

Jelentés

CORRTRAFO:	funkció felhívás

<Corr_Status>:	funkció visszaadási érték		
	adattípus:	INT	
	értékek:	0	A funkció hiba nélkül végre lett hajtva.
		1	Nincs aktív transzformáció.
		2	Az aktuálisan aktív transzformáció nem tájolási transzformáció.
		3	Az aktív tájolási transzformáció nem kinematikai láncokkal lett definiálva.
		10	A <Corr_Index> felhívás paraméter negatív.
		11	A <Corr_Mode> felhívás paraméter negatív.
		12	Érvénytelen utalás egy rész-láncra egy szakaszára (<Corr_Index> 1-es helye). Az érték nem lehet nagyobb a tájolótengelyek számánál a rész-láncban.
		13	Érvénytelen utalás egy rész-lánc tájolási tengelyére (<Corr_Index> 1-es helye). Az érték kisebb kell legyen a tájolótengelyek számánál a rész-láncban.
		14	Érvénytelen utalás egy rész-láncra (<Corr_Index> 10-es helye). Csak a 0 és 1 értékek megengedettek (utalás rész- ill. szerszám-láncra). Ez a hibaszám akkor lép fel, ha a rész-elem, amelyre a <Corr_Index> utal, nem létezik.
		15	A szakaszban, amelyre a <Corr_Index> paraméter utal, nincs kontúr-elem ($NT_CORR_ELEM_P ill. $NT_CORR_ELEM_T) definiálva.
		20	Érvénytelen korrekció módus (<Corr_Mode> 1-es helye). Csak a 0 és 1 értékek megengedettek.
		21	Érvénytelen korrekció módus (<Corr_Mode> 10-es és/vagy 100-as helye). Egy tengely-irány írásánál csak az 1-es hely lehet nullától eltérő.
		30	A <Corr_Mode> 100-as helye érvénytelen. Csak a 0 és 1 értékek megengedettek.
		31	A <Corr_Mode> 1000-es helye érvénytelen. Csak a 0 és 1 értékek megengedettek.
		40	Az irányvektor, amelyet tengelyirányként kell átvenni a nulla-vektor. A hiba csak akkor léphet fel, ha a <Corr_Mode> 1000-es helye 0-val egyenlő. Ha ezen paraméter 1000-es helye 1 (maximális korrekció felügyelet kikapcsolva), lehet írni a nulla-vektort is.
		41	Egy offset vektor korrekciójánál az eltérés az aktuális értékhez képest legalább egy koordinátával nagyobb, mint a SD41610 $SN_CORR_TRAFO_LIN_MAX beállítási adattal megadott maximális érték. A <Corr_Vect> paramétert egy hiba-vektor felülírja. Ez akkor is érvényes, ha a megmunkálás vészjelzéssel megszakad (lásd <No_Alarm> paraméterek). A hiba-vektor azokban a komponensekben, amelyek korrekcióértéke a megengedett határt túllépte, a megállapított korrekcióérték és a határ közötti előjelhelyes eltérést tartalmazza. Azon komponensek tartalma, amelyek nem lépték túl a határukat, nulla.
		42	Egy irány-vektor korrekciójánál a szögeltérés az aktuális irányhoz képest nagyobb, mint a SD41611 $SN_CORR_TRAFO_DIR_MAX beállítási adattal megadott maximális érték.

		43	Egy rendszerváltozó írásának kísérlete a hiányzó írási jog miatt el lett utasítva.
`<Corr_Vect>`:	korrekció-vektor A korrekció-vektor tartalma a következő **<**Corr_Index**>** és `<Corr_Mode>` paraméterekkel lesz definiálva. Ha `<Corr_Status> = 41`, a vektor tartalma felül lesz írva (l. f.).		
	adattípus:	REAL	
<`Corr_Index`**>**:	Szakasz, melynek a kontúrelemeit módosítani kell / Tájolási tengely indexe, melynek az irányvektorát módosítani kell.		
	adattípus:	INT	
	A `<Corr_Index>` paraméter decimálisan kódolt (1-estől 10-es helyig):		
	1-es hely:	Tartalmazza a szakasz ill. a tájolási tengely indexét a rész-láncban.	
	10-es hely:	Rész-láncokra utal.	
		0x	munkadarab-lánc
		1x	szerszám-lánc

<Corr_Mode>:	korrekció módus		
	adattípus:	INT	
	A <Corr_Mode> paraméter decimálisan kódolt (1-estől 1000-es helyig):		
	1-es hely:	Meghatározza, hogy melyik elem legyen korrigálva.	
		xxx**0**	Egy lineáris offset-vektor korrekciója
		xxx**1**	Egy tájolási tengely irányvektorának korrekciója
	10-es hely:	Meghatározza, hogy a korrekció-elem, amelyre a <Corr_Index> tartalma utal, módosítva legyen-e.	
		xx**0**x	A korrekció-vektor közvetlenül a korrekció-elembe lesz írva. Ezt a változatot lehet használni a korrekció-elem közvetlen írására, anélkül, hogy az érintett rendszeradat ($NK_OFF_DIR[<n>, ...]) <n> indexe ismert lenne.
		xx**1**x	Mint 0, de azzal a különbséggel, hogy az átadott korrekció-érték a világ koordinátarendszerben lesz értelmezve. A 0 és 1 változatok között eltérés mindig akkor keletkezhet, ha a kinematikai lánc az alaphelyzetben (összes tájolási tengely pozíciója 0-val egyenlő) további forgatásokat tartalmaz.
		xx**2**x	Mint 1, de azzal a különbséggel, hogy a korrekcióérték a teljes szakaszra vonatkozik, azaz a korrekció-elembe egy olyan érték lesz bevive, hogy a teljes szakasz eléri a korrekcióértékkel definiált hosszat.
		Utalás: Az 1 és 2 értékek egy tájolási tengely irányvektorának az írásánál nem megengedettek.	
	100-as hely:	Meghatározza, hogy a <Corr_Vect> paraméter tartalmát hogyan kell értelmezni.	
		x**0**xx	Az átadott <Corr_Vect> korrekció-vektor tartalmazza a korrekció-elem teljes új hosszát, ill. a szakaszét, amelyre a <Corr_Index> a <Corr_Mode> 10-es helyével kapcsolatban utal (abszolút korrekció).
		x**1**xx	Az átadott <Corr_Vect> korrekció-vektor csak az eltérést tartalmazza a korrekció-elem aktuális hosszától, ill. a szakaszét, amelyre a <Corr_Index> a <Corr_Mode> 10-es helyével kapcsolatban utal (növekményes korrekció).
		Utalás: Egy tájolási tengely irányvektorának korrekciójánál a 100-as hely tartalma 0 kell legyen.	
	1000-es hely:	Meghatározza, hogy a korrekció a következő maximum értékkel korlátozva legyen-e. • SD41610 $SN_CORR_TRAFO_LIN_MAX ill.: • SD41611 $SN_CORR_TRAFO_DIR_MAX	
		0xxx	A maximális korrekció felügyelete aktív.
		1xxx	A maximális korrekció felügyelete nem aktív.

<No_Alarm>:	Viselkedés hiba esetén (visszaadási érték > 0) (**opcionális**)		
	adattípus:	BOOL	
	érték:	FALSE (alapér-ték)	Hiba esetén a program feldolgozása meg lesz szakítva és a 14103 vészjelzés ki lesz jelezve.
		TRUE	Hiba esetén a program feldolgozása nem szakad meg és vészjelzés nem lesz kijelezve. Alkalmazási eset: alkalmazás-specifikus reakciónak megfelelő visszaadási érték

Megjegyzés

Ha a funkció felhívásánál egy hiba lép fel, vagy egy vészjelzés lesz kiadva vagy egy hibaszám lesz visszaadva (lásd a <No_Alarm> paramétert), így a felhasználó maga tud a hiba állapotra megfelelően reagálni. A hiba okát egy vészjelzés paraméter írja le részletesebben. A vészjelzés helyett visszaadott hibaszám azonos a vészjelzés paraméterrel.

További információk a CORRTRAFO-hoz

Egy tájolási transzformációs gép kinematikai struktúrája egy vagy két kinematikai lánccal (rész-láncok) lesz leírva, amelyek a világ koordinátarendszer nullapontjából indulnak ki. A két lánc egyike, a **szerszám-lánc**, a szerszám vonatkoztatási pontján végződik, a másik, a **munkadarab-lánc**, az alap koordinátarendszer nullapontján.

A CORRTRAFO funkció emelőkar hosszakat és tengely irányokat hiányol tájolási transzformációs gépeknél és ezeket az értékeket speciális korrekció-elemekbe írja. Egy kinematikai lánc többek között OFFSET típusú elemekkel lesz írva, amelyek az $NK_TYPE által lesznek definiálva.

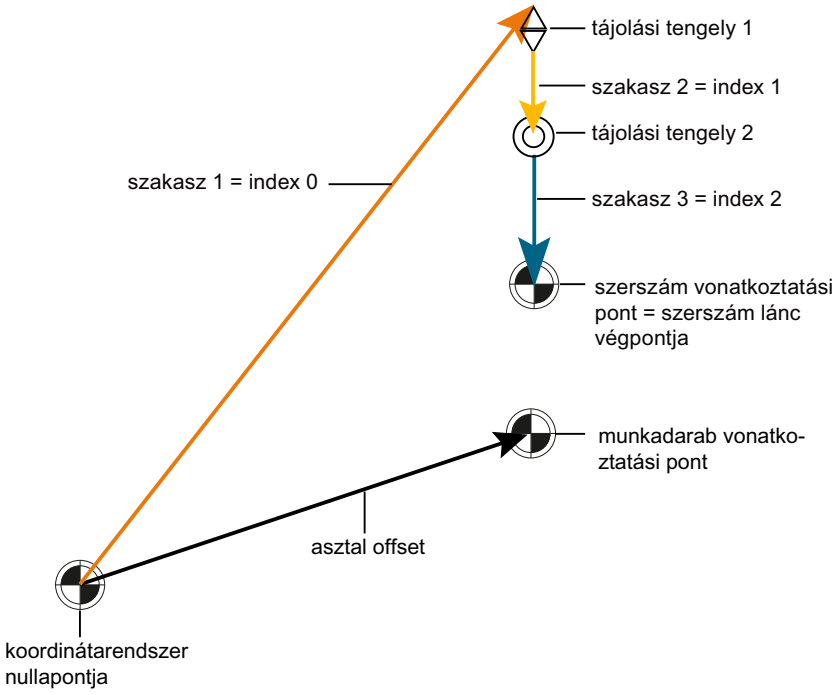
CORRTRAFO szakaszokkal dolgozik

Mindkét rész-lánc egyenként maximum négy szakaszra osztódhat:

- Szakasz 1 a lánc kezdőpontján indul és az első tájolási tengelyen végződik.
- Szakasz 2 a tájolási tengely 1 és a tájolási tengely 2 közötti szakasz.
- Szakasz 3 a tájolási tengely 2 és a tájolási tengely 3 közötti szakasz.
- Szakasz 4 a tájolási tengely 3 és a szerszám- ill. munkadarab-lánc vége közötti szakasz.

A szakaszok tartalmazhatják az OFFSET vagy ROT_CONST állandó lánc-elemeket.

A következő ábra egy tájolási transzformációt mutat 2 tájolási tengellyel:



Kép 10-1 CORRTRAFO példa

A szakaszok egyértelműen vannak definiálva. Ha végig futunk a kinematikai láncon a kezdőpontjától a végpontjáig, akkor az első szakasz indexe 0, a következőé az index 1 stb. Az utolsó szakasz indexe ekkor mindig egyenlő a tájolási tengelyek számával.

Kontúr-elemek

Minden egyes szakaszban lehet egy állandó kinematikai elemre ($NK_TYPE[<n>] = "OFFSET" típusú lánc-elem) az $NT_CORR_ELEM_T[<n>, 0 ... 3] ill. $NT_CORR_ELEM_P[<n>, 0 ... 3] rendszerváltozóval utalni. Az így megadott elemekbe a CORRTRAFO funkció segítségével azok a korrekció értékek lesznek írva, amelyek a gép bemérésénél lettek megállapítva.

Példa transzformáció index = 1-gyel:

- $NT_CORR_ELEM_T[1,0] = "C_AXIS_OFFSET"; C tengely offset (tájolótengely 1) a szakasz 1-ben korrekció elemként van definiálva.
- $NT_CORR_ELEM_T[1,1] = "B_AXIS_OFFSET"; B tengely offset (tájolótengely 2) a szakasz 2-ben korrekció elemként van definiálva.
- $NT_CORR_ELEM_T[1,2] = "BASE_TOOL_OFFSET"; B tengely offset a szerszám vonatkoztatási ponthoz a szakasz 3-ban korrekció elemként van definiálva.

Az utalások sorrendje a $NT_CORR_ELEM_T/P[<n>, 0 ... 3]-ban meg kell egyezzen a fent leírt szakaszokéval, azaz a $NT_CORR_ELEM_T/P [<n>, 0]-ban csak egy lánc-elem állhat, amelyik az első tájolási tengely előtt van, stb..

A CORRTRAFO funkció az így definiált kontúrelemekbe azokat az értékeket írja, amelyek a gép bemérésével lettek meghatározva. A korrekciós értékek módosítása a CORRTRAFO-ban a <Corr_Mode> paraméterrel lesz definiálva.

Lánc lezárása

Ha a $NT_CNTRL[<n>] rendszerváltozóban a bit 7 vagy bit 8 be van állítva, a szerszám-lánc végén (bit 7) ill. a szerszám-lánc kezdőpontja előtt (bit 8) belül automatikusan további állandó lánc-elemek lesznek beszúrva, amelyek létrehoznak egy összeköttetést a lánc végpontjától a gép nullapontjához ("lánc lezárása").

Az automatikusan beszúrt elemeket nem lehet kívülről írni, hanem csak olvasni (lásd ehhez a $AC_TRAFO_CORR_ELEM_P/T rendszerváltozót).

Pont a szerszám-lánc bezárásához

Ha a $NT_CLOSE_CHAIN_T rendszerváltozó nem üres, a szerszám-lánc nem a lánc végpontján, hanem a megadott lánc-elem végpontján lesz lezárva. Ezen pont utáni lánc-elemek a transzformáció aktiválásánál egy megfelelő nullaponteltolást okoznak.

Egy tájolási tengely indexe

A tájolótengelyek közötti állandók offset kivételével a CORRTRAFO funkcióval a tájolótengelyek irányvektorait is lehet írni. Egy tájolási tengely indexe itt az az index értendő, amelyik akkor adódik, ha a kinematikai láncon az elejétől a végéig megyünk. ami során a számlálás nullánál kezdődik. Egy tájolási tengely indexe ezért mindig azonos az előző szakasz indexével.

Egy tájolási tengely indexét a $AC_TRAFO_ORIAX_LOC rendszerváltozóval is meg lehet állapítani.

Egy lánc-elem maximálisan megengedett változásai

Egy lánc-elem maximálisan megengedett változását lehet határolni az SD41610 $SN_CORR_TRAFO_LIN_MAX-szal az offset-vektorokra és az SD41611 $SN_CORR_TRAFO_DIR_MAX-szal a tájolási tengelyek irányvektoraira. Az SD41610 $SN_CORR_TRAFO_LIN_MAX megadja, hogy az egyes vektor-komponenseket a vonatkoztatási értékükkel szemben maximum milyen értékkel szabad megváltoztatni. Az SD41611 $SN_CORR_TRAFO_DIR_MAX megadja, hogy a tengely-vektort a vonatkoztatási értékével szemben maximum milyen szöggel szabad megváltoztatni. A vonatkozási érték érték ennél mindig a megfelelő érték, amelyik a CORRTRAFO felhívásánál aktív transzformációban hatásos. Azaz a transzformáció aktiválása után az adattárolás kinematikai adatai tartalmának esetleges megváltoztatása nem hat a CORRTRAFO funkció hatásmódjára.

Szerszámkorrekciók 11

11.1 Korrekció-tároló

A korrekció-tároló felépítése

Minden adatmező felhívható egy T- és D-számmal (a "lapos D-szám"-on kívül) és a szerszám geometriai adatai mellett még további beviteleket tartalmaz, pl. a szerszámtípust.

Lapos D-szám struktúra

A "lapos D-szám struktúra" akkor kerül alkalmazásra, ha a szerszámkezelés az NC-n kívül történik. Ebben az esetben a D-számok a hozzátartozó szerszámkorrekció adatokkal a szerszámokhoz hozzárendelés nélkül vannak létrehozva.

A munkadarab-programban lehet továbbra is T-t programozni. Ennek a T-nek azonban nincs kapcsolata a programozott D-számhoz.

Felhasználói vágóél-adatok

A felhasználói vágóél-adatokat gépadatokkal lehet konfigurálni. Vegyük ehhez figyelembe a gépgyártó tájékoztatásait.

Szerszám-paraméter

Megjegyzés

Egyes értékek a korrekciós tárolóban

A P1...P25 korrekció-tárolók egyes értékei rendszerváltozókkal a programból olvashatók és írhatók. Az összes többi paraméter foglalt.

A $TC_DP6 ... $TC_DP8, $TC_DP10 és $TC_DP11 ill. $TC_DP15 ... $TC_DP17, $TC_DP19 és $TC_DP20 paramétereknek a szerszámtípustól függően más jelentésük van.

Szerszámparaméter-szám (DP)	Rendszerváltozók jelentése	Megjegyzés
$TC_DP1	szerszámtípus	áttekintés a listában
$TC_DP2	vágóélhelyzet	csak esztergaszerszámokra
geometria	**hosszkorrekció**	
$TC_DP3	hossz 1	beszámítás
$TC_DP4	hossz 2	típus és sík
$TC_DP5	hossz 3	
geometria	**sugár**	
$TC_DP6 [1)] $TC_DP6 [2)]	sugár 1 / hossz 1 átmérő d	maró-/eszterga- /köszörűszerszám vájatfűrész

Szerszámparaméter-szám (DP)	Rendszerváltozók jelentése	Megjegyzés
$TC_DP7 [1] $TC_DP7 [2]	hossz 2 / kúpos-maró saroksugár vájatszélesség b saroksugár	marószerszámok vájatfűrész
$TC_DP8 [1] $TC_DP8 [2]	lekerekítési sugár 1 marószerszámokra átállás k	marószerszámok vájatfűrész
$TC_DP9 [1] [3]	lekerekítési sugár 2	foglalt
$TC_DP10 [1]	szög 1 szerszám homlokoldala	kúpos marószerszámok
$TC_DP11 [1]	szög 2 szerszám hossztengely	kúpos marószerszámok
kopás	**hossz-és sugárkorrekció**	
$TC_DP12	hossz 1	
$TC_DP13	hossz 2	
$TC_DP14	hossz 3	
$TC_DP15 [1] $TC_DP15 [2]	sugár 1 / hossz 1 átmérő d	maró-/eszterga- /köszörűszerszám vájatfűrész
$TC_DP16 [1] $TC_DP16 [3]	hossz 2 / kúpos maró saroksugár vájatszélesség b saroksugár	marószerszámok vájatfűrész
$TC_DP17 [1] $TC_DP17 [2]	lekerekítési sugár 1 marószerszámokra átállás k	marás / 3D homlokmarás vájatfűrész
$TC_DP18 [1] [3]	lekerekítési sugár 2	foglalt
$TC_DP19[1]	szög 1 szerszám homlokoldala	kúpos marószerszámok
$TC_DP20[1]	szög 2 szerszám hossztengely	kúpos marószerszámok
alapméret/adapter	**hosszkorrekciók**	
$TC_DP21	hossz 1	
$TC_DP22	hossz 2	
$TC_DP23	hossz 3	
technológia		
$TC_DP24	szabadszög	csak esztergaszerszámokra
$TC_DP25		foglalt

[1] Marószerszámoknál is érvényes a 3D-s homlokmaráshoz

[2] Csak vájatfűrész szerszámtípusnál

[3] foglalt (SINUMERIK 840D sl nem használja)

Megjegyzések

A geometriai méretek (pl. hossz 1 vagy sugár) több beviteli összetevőből állnak. Ezek összeadódnak egy eredő nagysággá (pl. összhossz1, összsugár), és ez lesz hatásos.

Nem szükséges korrekciók nulla értékkel vannak feltöltve.

$TC-DP1 ... $TC-DP23 szerszámparaméterek kontúrszerszámoknál

Megjegyzés

A táblázatban nem megadott szerszámparaméterek, mint pl. $TC_DP7,nem lesznek kiértékelve, vagyis a tartalmuknak nincs jelentősége.

Szerszámparaméter-szám (DP)	Jelentés	Dn vágóélek		Megjegyzés
$TC_DP1	Szerszámtípus			400 ... 599
$TC_DP2	vágóélhelyzet			
geometria	hosszkorrekció			
$TC_DP3	hossz 1			
$TC_DP4	hossz 2			
$TC_DP5	hossz 3			
geometria	sugár			
$TC_DP6	sugár			
geometria	határszög			
$TC_DP10	minimális határszög			
$TC_DP11	maximális határszög			
kopás	hossz-és sugárkorrekció			
$TC_DP12	hossz 1 kopás			
$TC_DP13	hossz 2 kopás			
$TC_DP14	hossz 3 kopás			
$TC_DP15	sugár kopás			
kopás	határszög			
$TC_DP19	kopás min. határszög			
$TC_DP20	kopás max. határszög			
alapméret/adapter	hosszkorrekciók			
$TC_DP21	hossz 1			
$TC_DP22	hossz 2			
$TC_DP23	hossz 3			

Alapérték és kopásérték

Az eredő érték az alapérték és a kopásérték összege (pl. $TC_DP6 + $TC_DP15 a sugárra). Az első vágóél szerszámhosszához ezen kívül még hozzá lesz adva a bázisméret ($TC_DP21 – $TC_DP23). Ezen túlmenően a szerszámhosszakra hat minden más érték, amelyek egy szokásos szerszámnál a hatásos szerszámhosszakat befolyásolhatják (adapter, tájolható szerszámtartó, beállítási adatok).

Határszög 1 és 2

A határszög 1 ill. 2 a vágóél középpontjából a vágóél vonatkoztatási pontjába mutató vektorra vonatkoznak és az órajárás irányába und értendők.

11.2 Additív korrekciók

11.2.1 Additív korrekciókat kiválasztani (DL)

Az additív korrekciók a megmunkálás során programozható folyamat-korrekciók. Ezek egy vágóél geometriai adataira vonatkoznak és ezzel a szerszám-vágóél adatok részei.

Egy additív korrekció adatai egy DL-számmal lesznek felhívva (DL: Location dependent; a mindenkori használat helyére vonatkozó korrekciók) és a kezelőfelületen lesznek beadva.

Alkalmazás

Az additív korrekciókkal a használat helyétől függő mérési hiba kiegyenlíthető.

Szintaxis

```
DL=<szám>
```

Jelentés

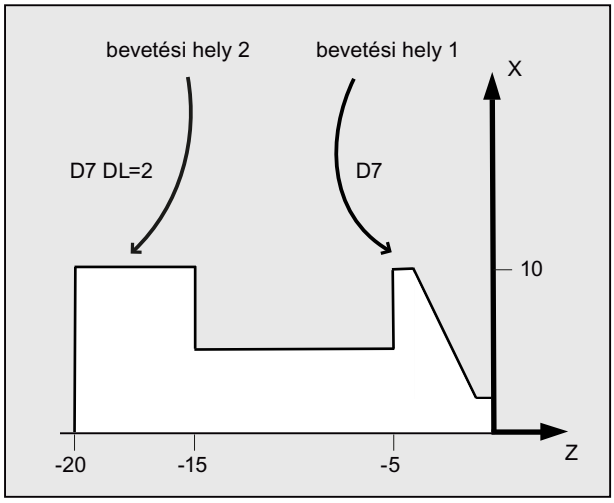
DL:	utasítás az additív korrekció aktiválására
<szám>:	A <szám> paraméterrel lesz megadva az aktiválandó hozzáadandó szerszámkorrekciós adatkészlet megadva..

Megjegyzés

Az additív korrekciók számának megadása és aktiválása gépadatokkal történik (→ gépgyártó tájékoztatásait figyelembe venni!)

Példa

Ugyanaz a vágóélet 2 tartóhelyre használják:



Programkód	Kommentár
N110 T7 D7	; A revolver a 7-es helyre lesz pozicionálva. D7 és DL=1 aktiválva lesz és a következő mondatban ki lesz egyenlítve
N120 G0 X10 Z1	
N130 G1 Z-6	
N140 G0 DL=2 Z-14	; A D7-hez additív a DL=2 aktiválva lesz és a következő mondatban ki lesz egyenlítve.
N150 G1 Z-21	
N160 G0 X200 Z200	; Szerszámcserepontra menni.
...	

11.2.2 Kopási és beállítási érték megadása ($TC_SCPxy[t,d], $TC_ECPxy[t,d])

A kopási és beállítási értékeket rendszerváltozókkal lehet olvasni és írni. Ennek a logikája a szerszámok és vágóélek megfelelő rendszerváltozóinak logikájához illeszkedik.

Rendszerváltozók

$TC_SCPxy[<t>,<d>]:	kopásértékek, amelyek az xy-nal a mindenkori geometria paraméterekhez vannak rendelve, ahol az x a kopásérték száma és az y a vonatkoztatás a geometria paraméterre
$TC_ECPxy[<t>,<d>]:	beállítási értékek, amelyek az xy-nal a mindenkori geometria paraméterekhez vannak rendelve, ahol az x a beállítási érték száma és az y a vonatkoztatás a geometria paraméterre
<t>: szerszám T-száma <d>: szerszám vágóél D-száma	

Megjegyzés

A megadott kopási és beállítási értékek hozzá lesznek adva a geometria paraméterekhez és az egyéb korrekció-paraméterekhez (D-szám)

Példa

A hossz 1 kopásértéke a <t> szerszám <d> vágóélére 1.0 értékre lesz megadva.

paraméter: $TC_DP**3** (hossz1, esztergaszerszámoknál)

kopásértékek: $TC_SCP1**3** ... $TC_SCP6**3**

beállítási értékek: $TC_ECP1**3** ... $TC_ECP6**3**

$TC_SCP4**3** [<t>,<d>] = 1.0

11.2.3 Additív korrekciókat törölni (DELDL)

A `DELDL` utasítással lehet egy szerszám vágóéléhez az additív korrekciókat törölni (tároló felszabadítása). Ennek során törölve lesznek a megadott kopásértékek és beállítási értékek is.

Szintaxis

```
DELDL[<t>,<d>]
DELDL[<t>]
DELDL
<állapot>=DELDL[<t>,<d>]
```

Jelentés

`DELDL`:	utasítás az additív korrekciók törléséhez	
`<t>`:	szerszám T-száma	
`<d>`:	szerszám vágóél D-száma	
`DELDL[<t>,<d>]`:	A`<t>` szerszám `<d>` vágóélének az összes additív korrekciója törölve lesz.	
`DELDL[<t>]`:	A `<t>` szerszám összes vágóélének az összes additív korrekciója törölve lesz.	
`DELDL`:	A TO egység összes szerszáma összes vágóélének az összes additív korrekciója törölve lesz.	
`<állapot>`:	törlés állapot	
	érték:	jelentés:
	0	A törlés sikeresen végre lett hajtva.
	-	A törlés nem lett végrehajtva (ha a paraméterezés pontosan egy vágóélet jelöl), vagy a törlés nem volt teljes (ha a paraméterezés több vágóélet jelöl).

Megjegyzés

Az aktív szerszám kopási és beállítási értékeit nem lehetett törölni (a `D` ill. a szerszámadatok törlési viselkedésével azonos viselkedés).

11.3 Szerszámkorrekció - különleges kezelés

Az SD42900 - SD42960 beállítási adatokkal a szerszámhossz és kopás előjelének kiértékelése vezérelhető.

Ez ugyanígy érvényes a kopás-összetevők viselkedésére a geometriatengelyek tükrözésénél vagy a megmunkálási sík változtatásánál és a hőmérséklet-kompenzációra is a szerszámirányban.

Kopásértékek

Ha a továbbiakban a kopásértékekre hivatkozunk, akkor ezalatt mindig a tulajdonképpeni kopásértékek ($TC_DP12 - $TC_DP20) összegét és a kopásértékekből ($SCPX3 - $SCPX11) valamint a beállítási értékekből ($ECPX3 - $ECPX11) képzett összegkorrekciót kell érteni.

Az összegkorrekciókhoz közelebbi található:
Irodalom:
Funktionshandbuch Werkzeugverwaltung

Beállítási-adatok

SD42900 $SC_MIRROR_TOOL_LENGTH	szerszámhossz-összetevők és az alapméretek összetevőinek tükrözése
SD42910 $SC_MIRROR_TOOL_WEAR	szerszámhossz-összetevők kopásértékeinek tükrözése
SD42920 $SC_WEAR_SIGN_CUTPOS	kopásösszetevők előjel-kiértékelése a vágóélhelyzettől függően
SD42930 $SC_WEAR_SIGN	kopásértékek előjeleit megfordítja
SD42935 $SC_WEAR_TRANSFORM	kopásértékek transzformációja
SD42940 $SC_TOOL_LENGTH_CONST	szerszámhossz-összetevők hozzárendelése a geometria-tengelyekhez
SD42950 $SC_TOOL_LENGTH_TYPE	szerszámhossz-összetevők hozzárendelése a szerszámtípustól függetlenül
SD42960 $SC_TOOL_TEMP_COMP	hőmérséklet-kompenzációs érték szerszámirányban szerszámtájolásnál is hatásos

Irodalom

Funktionshandbuch Grundfunktionen; Werkzeugkorrektur (W1)

További információk

A megváltoztatott beállítási adatok hatásossá válása

A szerszámösszetevők újraértékelése a leírt beállítási adatok változtatásakor csak akkor lesz hatásos, ha a következő alkalommal egy szerszám-vágóél kerül kiválasztásra. Ha egy szerszám már aktív, és e szerszám adataink az értékelése megváltozva kell hatásos legyen, akkor ezt a szerszámot újra ki kell választani.

Ugyanez érvényes arra az esetre, amikor a keletkező szerszámhossz változik, mivel egy tengely tükrözési távolsága előzőleg megváltozott. A tükrözési utasítás után a szerszámot újra ki kell választani, hogy a megváltozott szerszámhossz-összetevők hatásosak legyenek.

Tájolható szerszámtartó és új beállítási adatok

Az SD 42900 - SD 42940 beállítási adatok nem hatnak egy esetlegesen aktív orientálható szerszámtartó összetevőire. Egy szerszám azonban mindig a teljes eredő hosszával (szerszámhossz + kopás +alapméret) kerül beszámításra a tájolható szerszámtartónál. Az eredő össz-hossz kiszámolásánál minden változás figyelembe lesz véve, amelyet a beállítási adatok idéztek elő; vagyis a tájolható szerszámtartó vektorai függetlenek a megmunkálási síktól.

Megjegyzés

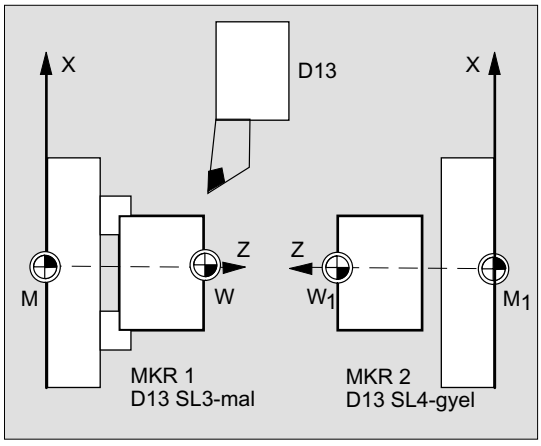
Tájolható szerszámtartó alkalmazásánál gyakran ésszerű az összes szerszámot egy nem tükrözött alaprendszerré definiálni, még azokat is, amelyek csak a tükrözési megmunkálásnál kerülnek felhasználásra. Tükrözött tengelyekkel történő megmunkálásnál a szerszámtartó úgy fog fordulni, hogy a szerszám valós helyzete helyesen legyen leírva. Minden szerszámhossz-összetevő ezután automatikusan a helyes irányban hat, úgy hogy az egyes összetevők kiértékelésének vezérlése a beállítási adatok által az egyes tengelyek tükrözési távolságától függően feleslegessé válik.

További alkalmazási lehetőségek

A tájolható szerszámtartó funkció alkalmazása akkor is ésszerű lehet, ha a gépen nincs fizikai lehetőség a szerszámok elforgatására, a szerszámok viszont különböző tájolásokkal vannak szilárdan szerelve. A szerszám-beméretezést ezután egységesen végre lehet hajtani egy alaptájolásban, és a megmunkálás számára releváns méretek egy virtuális szerszámtartó forgásai által adódnak.

11.3.1 Szerszámhosszak tükrözése

Az SD 42900 $SC_MIRROR_TOOL_LENGTH és SD 42910 $SC_MIRROR_TOOL_WEAR nem egyenlő nulla értékeinél szerszámhossz-összetevők és alapméretek összetevői kopásértékekkel a hozzájuk tartozó tengelyekre tükrözhetők.



SD42900 $SC_MIRROR_TOOL_LENGTH

Beállítási adat **nem egyenlő** nullával:

A szerszámhossz-összetevők ($TC_DP3, $TC_DP4 und $TC_DP5) és az alapméretek összetevői ($TC_DP21, $TC_DP22 und $TC_DP23), amelyeknek az ide tartozó tengelyei tükrözésre kerülnek, az előjel megfordítással szintén tükrözve lesznek.

A kopásértékek ezzel együtt **nem** kerülnek tükrözésre. Ha ezeket szintén tükrözni kell, akkor az SD42910 $SC_MIRROR_TOOL_WEAR beállítási adatot kell alkalmazni.

SD42910 $SC_MIRROR_TOOL_WEAR

Beállítási adat **nem egyenlő** nullával:

Azoknak a szerszámhossz-összetevőknek a kopásértékei, amelyeknek az ide tartozó tengelyei már tükrözve vannak, az előjel megfordítással által szintén tükrözésre kerülnek.

11.3.2 Kopás előjel értékelés

A SD 42920 $SC_WEAR_SIGN_CUTPOS és SD $SC_42930 WEAR_SIGN beállítási adatok nem egyenlő nulla értékeivel lehetséges a kopási komponensek előjel-kiértékelésének invertálása.

SD42920 $SC_WEAR_SIGN_CUTPOS

Beállítási adat **nem egyenlő** nullával:

Releváns vágóélhelyzetű szerszámoknál (eszterga- és köszörűszerszámok, szerszámtípusok 400) a kopás-összetevők előjel-értékelése a megmunkálási síkban a vágóélhelyzettől függ. A releváns vágóélhelyzet nélküli szerszámtípusoknál ennek a beállítási adatnak nincs jelentősége.

Az alábbi táblázatban a méreteket egy olyan X-szel jelöljük, amelynek az előjele SD 42920 (nem egyenlő 0-val) által megfordításra kerül:

vágóélhelyzet	hossz 1	hossz 2
1		
2		X
3	X	X
4	X	
5		
6		
7		X
8	X	
9		

Megjegyzés

Az SD42920 és SD42910 általi előjel-értékelések egymástól függetlenek. Ha pl. egy méretadat előjele mindkét beállítási adat által megváltoztatásra kerül, akkor az így kapott előjel változatlan marad.

SD42930 $SC_WEAR_SIGN

Beállítási adat **nem egyenlő** nullával:

Minden kopásméret előjele megfordításra kerül. Ez ugyanúgy hat a szerszámhosszra, mint az egyéb méretekre, pl. szerszámsugár, kerekítési sugár stb.

Ha egy pozitív kopásméret kerül beadásra, akkor a szerszám "rövidebb" és "keskenyebb" lesz, lásd a "Szerszámkorrekciók, Különleges eljárások, Megváltozott beállítási adatok hatásossá válása".

11.3.3 Aktív megmunkálás koordinátarendszere (TOWSTD, TOWMCS, TOWWCS, TOWBCS, TOWTCS, TOWKCS)

A gép kinematikájától vagy egy tájolható szerszámtartó meglététől függően az ezen koordinátarendszerek egyikében mért kopásértékek átvezetve ill. transzformálva lesznek egy alkalmas koordinátarendszerbe.

Az aktív megmunkálás koordinátarendszerei

A következő koordinátarendszerekből származhatnak a szerszámhossz offsetjei, amelyek a kopás szerszámhossz-összetevőként az 56-os csoport megfelelő G utasításával az aktív szerszámnál beszámításra kerülnek.

- Gép-koordinátarendszer (GKR)
- Alap-koordinátarendszer (AKR)
- Munkadarab-koordinátarendszer (MKR)
- Szerszám-koordinátarendszer (SKR)
- Kinematikai transzformáció szerszám-koordinátarendszere (KKR)

Szintaxis

```
TOWSTD
TOWMCS
TOWWCS
TOWBCS
TOWTCS
TOWKCS
```

Jelentés

TOWSTD:	szerszámhossz kopásérték korrekció alaphelyzet-érték
TOWMCS:	szerszámhossz korrekció GKR-ben
TOWWCS:	szerszámhossz korrekció MKR-ben
TOWBCS:	szerszámhossz korrekció AKR-ben
TOWTCS:	szerszámhossz korrekció szerszámtartó vonatkoztatási ponton (tájolható szerszám-tartó)
TOWKCS:	szerszámfej szerszámhossz korrekció (kinematikai transzformáció)

További információk

Megkülönböztető jellemzők

A következő táblázatban találhatók a legfontosabb megkülönböztető jellemzők:

G utasítás	kopásérték	aktív tájolható szerszámtartó
TOWSTD:	alaphelyzet-érték, szerszámhossz	kopásértékekre hat a forgatás
TOWMCS:	kopásérték GKR-ben TOWMCS azonos TOWSTD-vel, ha nincs aktív tájolható szerszámtartó	csak az eredő szerszámhossz vektora forog a kopás figyelembe vétele nélkül
TOWWCS:	kopásérték átszámítása MKR-ből GKR-be	szerszámvektor számítása a kopás figyelembe vétele nélkül, mint TOWMCS-nél
TOWBCS:	kopásérték átszámítása AKR-ből GKR-be	szerszámvektor számítása a kopás figyelembe vétele nélkül, mint TOWMCS-nél
TOWTCS:	kopásérték átszámítása szerszám-koordinátarendszerből GKR-be	szerszámvektor számítása a kopás figyelembe vétele nélkül, mint TOWMCS-nél

TOWWCS, TOWBCS, TOWTCS: A kopásvektor hozzá lesz adva a szerszámvektorhoz.

Lineáris transzformáció

A szerszámhossz GKR-ben csak akkor definiálható értelmesen, ha a GKR az AKR-ből egy lineáris transzformációval származik.

Nem lineáris transzformáció

Ha pl. TRANSMIT-tal egy nem lineáris transzformáció aktív, akkor a GKR megadásánál kívánt koordináta-rendszerként automatikusan az AKR lesz alkalmazva

Nincs kinematikai transzformáció és nincs tájolható szerszámtartó

Ha nincs kinematikai transzformáció és nincs tájolható szerszámtartó, akkor a MKR-ig mind a négy további koordinátarendszer egybe esik. Ezzel csak az MKR különbözik a többitől. Mivel kizárólag a szerszámhosszakat kell kiértékelni, a koordinátarendszerek közötti eltolásoknak nincs jelentősége.

Irodalom:
További információk találhatók a szerszámkorrekcióhoz:
Funktionshandbuch Grundfunktionen; Werkzeugkorrektur (W1)

Kopásértékek beszámítása

A **SD42935 $SC_WEAR_TRANSFORM** beállítási adat megadja, hogy a következő három kopási komponense:

- kopás
- összegkorrekciók finom
- összegkorrekciók durva

egy forgatásnak egy adapter-transzformáció vagy egy tájolható szerszámtartó által legyen alávetve, ha a következő G utasítások egyike aktív:

- `TOWSTD`
  alaphelyzet szerszámhossz korrekciókhoz
- `TOWMCS`
  kopásértékek gép-koordinátarendszerben (GKR)
- `TOWWCS`
  kopásértékek munkadarab-koordinátarendszerben (MKR)
- `TOWBCS`
  kopásértékek alap-koordinátarendszerben (AKR)
- `TOWTCS`
  kopásértékek szerszám koordinátarendszerben a szerszámtartó felvevőn (T szerszámtartó vonatkoztatás)
- `TOWKCS`
  kopásértékek szerszámfej koordinátarendszerében kinetikus transzformációnál

Megjegyzés

Az egyes kopás-összetevők értékelését (hozzárendelés a geometria-tengelyekhez, előjel-kiértékelés) a következő tényezők befolyásolják:

- aktív sík
- adapter transzformáció
- beállítási-adatok:
  - 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 Szerszámhossz és síkcsere

Az SD42940 $SC_TOOL_LENGTH_CONST beállítási adat nem egyenlő nullával a szerszámhossz-összetevőket, mint hossz, kopás és alapméret, lehet a geometria-tengelyekhez az eszterga- és köszörűszerszámokra egy síkcserénél hozzárendelni.

SD42940 $SC_TOOL_LENGTH_CONST

Beállítási adat **nem egyenlő** nullával:

A szerszámhossz-összetevők (hossz, kopás és alapméret) hozzárendelése a geometria-tengelyekhez nem változik a megmunkálási sík (`G17` - `G19`) változtatásakor.

Az alábbi táblázat a szerszámhossz-összetevők geometriatengelyekhez történő hozzárendelését mutatja eszterga- és köszörűszerszámoknál (szerszámtípus 400 ... 599):

tartalom	hossz 1	hossz 2	hossz 3
17	Y	X	Z

*)	X	Z	Y
19	Z	Y	X
-17	X	Y	Z
-18	Z	X	Y
-19	Y	Z	X

*) Minden 0-val nem egyenlő érték, amelyik nem azonos a hat felsorolt érték valamelyikével, mint az érték 18 kerül kiértékelésre.

Az alábbi táblázat a szerszámhossz-összetevők geometriatengelyekhez történő hozzárendelését mutatja minden más szerszámnál (szerszámtípus <400 ill. >599):

megmunkálási sík	hossz 1	hossz 2	hossz 3
*)	Z	Y	X
18	Y	X	Z
19	X	Z	Y
-17	Z	X	Y
-18	Y	Z	X
-19	X	Y	Z

*) Minden 0-val nem egyenlő érték, amelyik nem azonos a hat felsorolt érték valamelyikével, mint az érték 17 kerül kiértékelésre.

Megjegyzés

A táblázatokban történő ábrázolásnál abból kell kiindulni, hogy a geometriatengelyeket 3-ig X, Y, Z-vel jelöljük. Egy korrekció egy tengelyhez történő hozzárendeléséhez nem a tengelyjelölő, hanem a tengelyek sorrendje mértékadó.

11.4 Online szerszámkorrekció

11.4.1 Polinom függvény definiálása (FCTDEF)

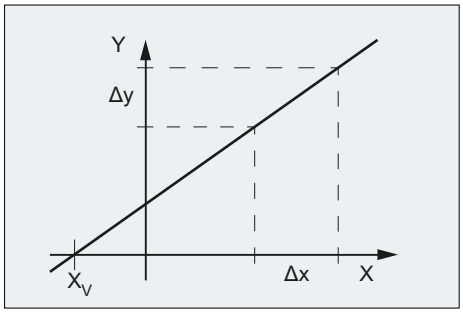
Bizonyos lehúzási stratégiákat (pl. lehúzó henger) jellemez, hogy a köszörűtárcsa folyamatosan (lineárisan) csökken sugárban a lehúzó henger fogásvételével. Ehhez szükséges egy lineáris függvény a lehúzó henger fogásvétele és a mindenkori hossz kopás-értékének írása között. A lineáris függvény definíciója az előre definiált FCTDEF(...) eljárással történik a maximum harmad fokú polinom függvényre.

Egyenes egyenlet

$y = f(x) = a_0 + a_1 * x_1$

a_1: egyenes meredeksége, $a_1 = \Delta x / \Delta y$

a_0: egyenes eltolása az X tengely mentén, $a_0 = -a1 * X_V$



Szintaxis

```
FCTDEF(<Func>,<LLimit>,<ULimit>,<a0>,<a1>,<a2>,<a3>)
```

Jelentés

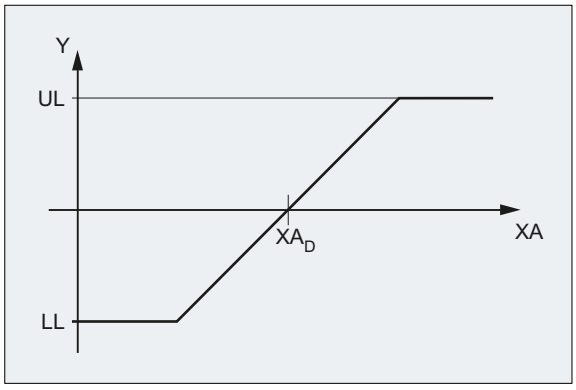
`FCTDEF(...)`:	egy polinom függvény definíciója PUTFTOCF(...)-ra: $y = f(x) = a_0 + a_1*x + a_2*x^2 + a_3*x^3$	
<`Func`**>**:	függvény-szám	
	adattípus:	INT
	értéktartomány:	1, 2, 3
`<LLimit>`:	alsó határérték	
	adattípus:	REAL
`<ULimit>`:	felső határérték	
	adattípus:	REAL
`<a0>,<a1>,<a2>,<a3>`:	polinom függvény együtthatói	
	adattípus:	REAL

Példa

Megadások

- függvény-szám 1
- alsó vagy felső határérték: -100, 100
- jelleggörbe emelkedése: a_1 = 1
- A munkapont a jelleggörbe közepén kell legyen. A jelleggörbe ehhez az XA tengely parancs-pozíciója alapján az MKR-ben a függvény definíciójának időpontjában az NC programban a negatív Y irányban el kell legyen tolva. $a_0 = -a_1 * XA_D = -1 * \$AA_IW$
- $a_2 = a_3 = 0$

Jelleggörbe



UL	felső határérték
LL	alsó határérték
XA_D	XA tengely parancsértéke a függvény definíciójának időpontjában az NC programban

Programozás

Programkód	Kommentár
`FCTDEF(1,-100,100,-$AA_IW[XA],1)`	`; függvény definíció`

11.4.2 Online szerszámkorrekció írás, folytonosan (PUTFTOCF)

Az előre definiált PUTFTOCF(...) eljárással az online szerszámkorrekció egy előtte FCTDEF(...) (Oldal 417)-fel definiált polinom függvénnyel történik.

Megjegyzés

Az online szerszámkorrekció történhet egy szinkronakcióval is.

További információkat lásd Szinkronakciók Működési kézikönyv.

Szintaxis

```
PUTFTOCF(<Func>,<RefVal>,<ToolPar>,<Chan>,<Sp>)
```

Jelentés

PUTFTOCF(...):	online-szerszámkorrekció írása, folyamatosan mondatonként a FCTDEF(...)-vel definiált polinom függvény alapján	
<Func>:	függvény-szám, megadva a függvény definíciónál FCTDEF(...)-val	
	adattípus:	INT
	értéktartomány:	1, 2, 3
<RefVal>:	vonatkozási érték, amiből a korrekció levezetésre kerül (pl. egy tengely valósértéke)	
	adattípus:	VAR REAL
<ToolPar>:	kopási paraméter száma (hossz 1, 2 vagy 3), amibe a korrekcióérték beszámításra kerül	
	adattípus:	INT
<Chan>:	csatorna száma, amelyben az online-szerszámkorrekció hatásos Utalás: Csak akkor szükséges, ha a korrekció nem az aktív csatornában kell hatásos legyen.	
	adattípus:	INT
<Sp>:	orsó száma, amelyre az online-szerszámkorrekció hat Utalás: Csak akkor szükséges, ha az aktív, használatban levő szerszám helyett egy nem aktív köszörű-tárcsát kell korrigálni.	
	adattípus:	INT

11.4.3 Online szerszámkorrekció írás, diszkrét (PUTFTOC)

Funkció

Az előre definiált PUTFTOC(...) eljárással az online szerszámkorrekció egy fix korrekcióértékkel történik.

Szintaxis

```
PUTFTOC(<CorrVal>,<ToolPar>,<Chan>,<Sp>)
```

Jelentés

PUTFTOC(...):	online-szerszámkorrekciót írni	
<CorrVal>:	korrekcióérték, ami a kopás paraméterhez lesz hozzáadva	
	adattípus:	REAL

<ToolPar>:	kopási paraméter száma (hossz 1, 2 vagy 3), amibe a korrekcióérték beszámításra kerül	
	adattípus:	INT
<Chan>:	csatorna száma, amelyben az online-szerszámkorrekció hatásos **Utalás:** Csak akkor szükséges, ha a korrekció nem az aktív csatornában kell hatásos legyen.	
	adattípus:	INT
<Sp>:	orsó száma, amelyre az online-szerszámkorrekció hat **Utalás:** Csak akkor szükséges, ha az aktív, használatban levő szerszám helyett egy nem aktív köszörű-tárcsát kell korrigálni.	
	adattípus:	INT

11.4.4 Online-szerszámkorrekciót be-/kikapcsolni (FTOCON/FTOCOF)

A FTOCON és FTOCOF G utasításokkal lesz az online-szerszámkorrekció be- ill. kikapcsolva.

Szintaxis

```
FTOCON
...
FTOCOF
```

Jelentés

FTOCON:	online-szerszámkorrekció bekapcsolás Az utasítást abban a csatornában kell programozni, amelyben az online-szerszámkorrekciót be kell kapcsolni.
FTOCOF:	online-szerszámkorrekció kikapcsolás Az utasítást abban a csatornában kell programozni, amelyben az online-szerszámkorrekciót ki kell kapcsolni. **Utalás:** Az FTOCOF-fal a szerszámkorrekció nem lesz tovább megtéve. Azonban a vágóél-specifikus korrekció-adatokban a PUTFTOC/PUTFTOCF-fal beszámított érték megmarad. Az online-szerszámkorrekció végleges deaktiválásához a FTOCOF után még a szerszám ki-/leválasztása (T...) szükséges.

11.5 3D-s szerszámkorrekció

11.5.1 3D-s szerszámsugár-korrekció aktiválása a 3D-s kerületmarásra (CUT3DC, CUT3DCD, ISD)

A 3D-s szerszámkorrekció (3D-s SZSK) a 3D-s kerületi marásra a határoló felületek figyelembe vétele nélkül a modálisan hatásos CUT3DC ill. CUT3DCD G utasításokkal lesz kiválasztva.

A tényleges aktiválás G41-gyel ill. G42-vel történik. Szerszámsugár-korrekció G40-nel lesz kikapcsolva.

Szintaxis

```
G41/G42 ORIC/ORID ISD=... CUT3DC/CUT3DCD CDOF2 X... Y... Z...
...
G40 X... Y... Z...
```

Jelentés

CUT3DC:	3D-s SZSK kerületi marásra (csak aktív 5-tengelyes transzformációnál)	
CUT3DCD:	egy eltérés-szerszámra vonatkoztatott 3D-s SZSK kerületi marásra (csak aktív 5-tengelyes transzformációnál) A sugár eltérés a $TC_DP15 szerszám-paraméterrel lesz megadva.	
G41/G42 X... Y... Z...:	szerszámsugár-korrekciót bekapcsolni	
	G41:	szerszámsugár-korrekció a kontúrtól balra
	G42:	szerszámsugár-korrekció a kontúrtól jobbra
	Utalás: A bekapcsolás egy lineáris mondatban (G0/G1) kell történjen.	
CDOF2:	Ütközés-felügyelet kikapcsolása 3D-s kerületi marásra	
ORIC/ORID:	A ORIC és ORID G utasításokkal a viselkedés lesz megadva a tájolás változásoknál a külső sarkokon.	
	ORIC:	Tájolás-változások a külső sarkokon a betoldandó körmondattal átlapolódnak.
	ORID:	Tájolás-változások a külső sarkokon a betoldandó körmondat előtt végre lesznek hajtva.
ISD=<Value>:	Az ISD címmel kerületi marásnál és **aktív** 3D-s szerszámsugár-korrekciónál a szerszám bemerülési mélységét meg lehet változtatni.	
	<Value>:	bemerülési mélység hossza
G40 X... Y... Z...:	szerszámsugár-korrekciót kikapcsolni **Utalás:** A kikapcsolás egy lineáris mondatban (G0/G1) geometria-tengely mozgásokkal kell történjen.	

Megjegyzés

A G utasítások a 3D-s SZSK kiválasztásához a rámeneti mondatban ki lesznek értékelve, azaz tipikusan a mondatban, amelyik G41-et vagy G42-t tartalmaz.

A G41 ill. G42 lehet a korrekcióban releváns geometria-tengelyek elmozdulása nélküli mondatokban is programozva. Ebben az esetben a rámenet mondat az ilyen mondatra következő első elmozdulási mondat.

A 3D-s SZSK változat váltása aktív szerszámsugár-korrekciónál vészjelzés nélkül figyelmen kívül lesz hagyva.

Példa

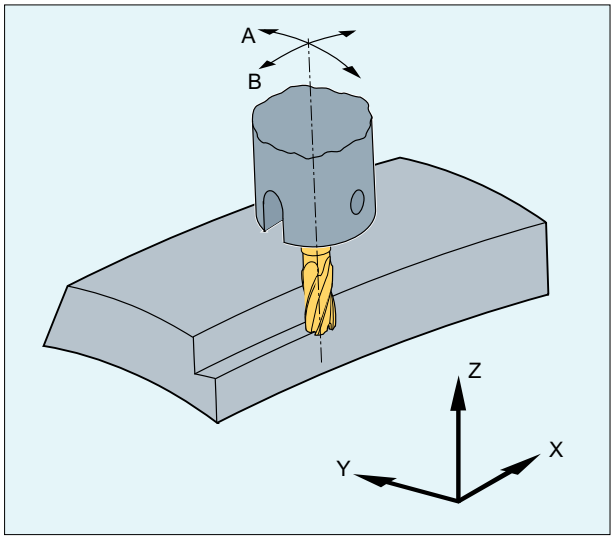
```
Programkód                          Kommentár
                                    ; D1 szerszám definíciója:
$TC_DP1[1,1]=120                    ; típus (származó)
$TC_DP3[1,1]=20                     ; hosszkorrekció-vektor
$TC_DP6[1,1]=8                      ; sugár

N10 X0 Y0 Z0 T1 D1 F12000           ; Szerszám kiválasztása.
N20 TRAORI(1)                       ; Transzformáció bekapcsolása.
N30 G42 ORIC ISD=10 CUT3DC G64 X30  ; 3D-s kerületi marást aktiválni.
                                    ; Tájolás-változások a külső sarkokon folytonosan,
                                    ; Bemerülési mélység: 10mm
N40 ORIWKS A30 B15                  ; Tájolás változás egy sarokon tengely-pozíciók megadásával.
N50 Y20 A3=1 C3=1                   ; Elmozdulási mondat tájolás változással.
                                    ; Tájolás megadása irányvektorral.
N60 X50 Y30                         ; Elmozdulási mondat állandó tájolással.
N70 Y50 A3=0.5 B3=1 C3=5            ; Elmozdulási mondat tájolás változással.
N80 M63                             ; Mondat mozgás információ nélkül.
N90 X0 ISD=20                       ; Elmozdulási mondat bemerülési mélység változással.
N100 G40 Y0                         ; Szerszámsugár-korrekció kikapcsolása.
N110 M30
```

További információk

Pálya és tájolás

A kerületmarásnak az itt használt változata egy pálya megadásával (vezető vonal) és a hozzátartozó tájolással van megvalósítva. Ilyen megmunkálásnál a pályára a szerszámformának nincs hatása. Döntő csak a szerszám-beavatkozási sugara.



Kép 11-1 Kerületi marás

Rámeneti viselkedés

A rámeneteli viselkedés a szerszámsugár-korrekció 3D-s változatánál mindig a NORM.

Viselkedés a külső sarkokon

Külső sarkokon a kerületi marásnál 3D-s SZSK-val a G csoport 18 G utasítások (sarok-viselkedés szerszámkorrekciónál) kiértékelése analóg a 2½D-s SZSK viszonyaival:

- G450: átmenet-kör (a szerszám megkerüli a munkadarabot egy körpályán)
  A 2½D-s SZSK megoldásával ellentétben a beszúrt kontúr-elem egy külső sarkon mindig egy kör 0 sugárral, amelyre a szerszámsugár-korrekció úgy hat, mint az összes többi programozott pályára is. Nem lehetséges a kör helyett egy kúpmetszetet beszúrni. Ezért a DISC címnek ebben az esetben nincs jelentés és nem lesz kiértékelve.
- G451: ekvidisztánsok metszéspontja (a szerszám a munkadarabban szabadra vág)
  A metszéspont úgy kerül meghatározásra, hogy a két érintett mondat offset-görbéi meghosszabbításra kerülnek és ezek metszéspontja a szerszámtájolásra merőleges síkban a sarkon lesz meghatározva.

A metszéspont eljárás(G451) nem lesz használva, ha az érintett mozgási mondatok közé legalább egy mondat be van szúrva, amelyik szerszámtájolás változást tartalmaz. Ilyen esetben a sarkon mindig egy kör lesz beszúrva.

Viselkedés tájolás-változásnál a külső sarkokon

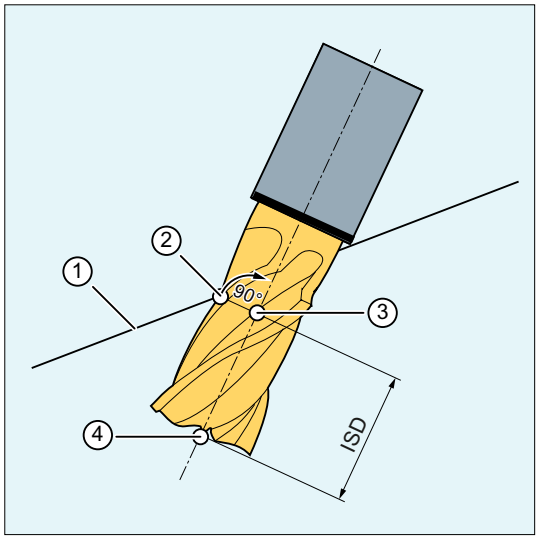
Az ORIC ill. ORID program utasításokkal lehet megadni, hogy a tájolás változások, amelyek a két sarkot képező mozgási mondat között lettek programozva, a beszúrt kör-mondat kezdete előtt (ORID) vagy azzal egyidőben (ORIC) legyenek végrehajtva.

Bemerülési mélység

A maró bemerülési mélysége a maró segédpont távolsága a szerszámcsúcstól.

A maró-segédpont a maró megmunkálási pontnak egy függőleges vetülete a programozott pályára a szerszámhossz tengelyen.

A bemerülési mélységgel a megmunkálási pont helyzete a szerszám palástfelületén lesz beállítva.



① programozott pálya
② maró megmunkálási pont
③ maró-segédpont
④ marócsúcs
ISD bemerülés-mélység (InSertion Depth)

Kép 11-2 Bemerülési mélység

Szerszámsugár-korrekció egy eltérés-szerszámra vonatkoztatva

Egy eltérés-szerszámra vonatkoztatott 3D-s SZSK a CUT3DCD utasítással lesz aktiválva. Ezt akkor kell használni, ha a programozott kontúr egy szabvány-szerszám középpont-pályára vonatkozik és a megmunkálás egy attól eltérő szerszámmal történik. A 3D-s szerszámsugár-korrekció kiszámításánál csak az aktív szerszám sugarának kopás értékei ($TC_DP15) és az esetleg programozott OFFN és TOFFR szerszámsugár-korrekció offsetek lesznek beszámítva. Az aktív szerszám alap-sugara ($TC_DP6) **nem** lesz figyelembe véve.

Zsebmarás ferde oldalfalakkal kerületi marásnál CUT3DC-vel

Ennél a 3D-s szerszámsugár-korrekciónál a marósugár eltérése kompenzálva lesz, amennyiben a ráállás a megmunkálandó felületre annak normálvektora irányából történik. Ennek során a maró homlokoldalának síkja változatlan marad, ha az ISD bemerülési mélység nem változik. Ekkor egy pl. a normál-szerszámnál kisebb sugarú maró a zseb-alját, ami a határoló felület is, nem érné el. A szerszám automatikus ráállításához a vezérlésnek ismerni kell ezt a határoló felületet, lásd a "3D-s kerületi marás egy határoló felület figyelembe vételével (CUT3DCC, CUT3DCCD) (Oldal 431)" fejezetet.

Advanced Surface / Top Surface

Megjegyzés

A CUT3DCD szerszámsugár-korrekció alkalmazásánál a "Advanced Surface" vagy "Top Surface" liszensz köteles funkcióval kapcsolatban a "Advanced Surface" / "Top Surface" beállítási ajánlásokat figyelembe kell venni!

A beállított adatok vizsgálatára a SIOS portálon speciális vizsgáló programok állnak rendelkezésre:

- vizsgáló programok Advanced Surface (https://support.industry.siemens.com/cs/ww/en/view/78956392)-hez
- vizsgáló programok Top Surface (https://support.industry.siemens.com/cs/ww/en/view/78956392)-hez

Lásd még

Vizsgáló programok Top Surface-hez (https://support.industry.siemens.com/cs/ww/en/view/109738423)

11.5.2 3D-s szerszámkorrekciót a 3D-s homlokmarásra kiválasztani (CUT3DF, CUT3DFS, CUT3DFF, CUT3DFD)

A 3D-s szerszámkorrekció (3D-s SZSK) a 3D-s homlokmarásra a modálisan hatásos CUT3DF, CUT3DFS, CUT3DFF ill. CUT3DFD G utasításokkal lesz kiválasztva.

A tényleges aktiválás G41-gyel ill. G42-vel történik.

A szerszámsugár-korrekció kiszámításához a 3D-s homlokmarás szükséges a felület normál definíciója a megmunkálandó síkra. Ez a G41 ill. G42 mondatában az A4, B4, C4 és A5, B5, C5 címekkel történik.

Szerszámsugár-korrekció G40-nel lesz kikapcsolva.

Szintaxis

```
G41/G42 ORIC/ORID CUT3DF/CUT3DFS/CUT3DFF/CUT3DFD X... Y... Z... A4=... B4=...
C4=... A5=... B5=... C5=...
...
G40 X... Y... Z...
```

Jelentés

CUT3DFS:	3D-s SZSK homlokmarásra állandó tájolással. A szerszám-tájolást a G17 - G19 határozza meg és ezt framek nem befolyásolják.	
CUT3DFF:	3D-s SZSK homlokmarásra állandó tájolással. A szerszám-tájolás a G17 - G19 által meghatározott és adott esetben egy frame által elforgatott irány.	
CUT3DF:	3D-s SZSK homlokmarásra tájolás-változással (csak aktív 5-tengelyes-transzformációval)	
CUT3DFD:	egy eltérés szerszámra vonatkoztatott 3D-s SZSK homlokmarásra tájolás-változással (csak aktív 5-tengelyes-transzformációval) A sugár eltérés a $TC_DP15 szerszám-paraméterrel lesz megadva. **Utalás**: CUT3DFD csak a "Felület normálok simítása 3D-s homlokmarásnál"-lal kombinációban lehetséges. Ez a "Top Surface" liszensz köteles funkció CYCLE832(...) (Oldal 775) felhívásával lesz aktiválva.	
G41/G42 X... Y... Z...:	szerszámkorrekció bekapcsolás A G41 és G42 viselkedése 3D-s homlokmarásnál azonos. **Utalás:** A bekapcsolás egy lineáris mondatban (G0/G1) kell történjen.	
A4/5=... B4/5=... C4/5=...:	a felület normál definíciója a megmunkálandó síkra	
	A4=... B4=... C4=...:	definíció a mondat elején
	A5=... B5=... C5=...:	definíció a mondat végén
ORIC/ORID:	A ORIC és ORID G utasításokkal a viselkedés lesz megadva a tájolás változásoknál a külső sarkokon.	
	ORIC:	Tájolás-változások a külső sarkokon a betoldandó körmondattal átlapolódnak.
	ORID:	Tájolás-változások a körmondat előtt végrehajtódnak.
G40 X... Y... Z...:	szerszámsugár-korrekciót kikapcsolni **Utalás:** A kikapcsolás egy lineáris mondatban (G0/G1) geometria-tengely mozgásokkal kell történjen.	

Megjegyzés

A G41 ill. G42 lehet a korrekcióban releváns geometria-tengelyek elmozdulása nélküli mondatokban is programozva. Ebben az esetben a rámenet mondat az ilyen mondatra következő első elmozdulási mondat.

A 3D-s SZSK változat váltása aktív szerszámsugár-korrekciónál vészjelzés nélkül figyelmen kívül lesz hagyva.

Példák

Példa 1: 3D-s homlokmarás CUT3DF-fel:

Programkód	Kommentár
N10	; D1 szerszám definíciója:
N20 $TC_DP1[1,1]=121	; szerszámtípus (kúpos maró)
N30 $TC_DP3[1,1]=20	; hosszkorrekció
N40 $TC_DP6[1,1]=5	; sugár
N50 $TC_DP7[1,1]=3	; lekerekítési sugár
N60	
N70	
N80 X0 Y0 Z0 A0 B0 C0 G17 T1 D1 F12000	; Szerszám kiválasztása.
N90 TRAORI(1)	; Tájolási transzformáció kiválasztani.
N100 B4=-1 C4=1	; A sík definíciója.
N110 **G41** ORID **CUT3DF** G64 X10 Y0 Z0	; Szerszámkorrekció bekapcsolása.
N120 X30	
N130 Y20 A4=1 C4=1	; Külső sarok, a sík új definíciója.
N140 B3=1 C3=5	; Tájolás változás ORID-dal.
N150 B3=1 C3=1	; Tájolás változás ORID-dal.
N160 X-10 A5=1 C5=2 ORIC	
N170 A3=-2 C3=1	; Tájolás változás ORIC-cal.
N180 A3=-1 C3=1	; Tájolás változás ORIC-cal.
N190 Y-10 A4=-1 C4=3	; A sík új definíciója.
N200 X-20 Y-20 Z10	; Belső sarok előző mondattal.
N210 X-30 Y10 A4=1 C4=1	; Belső sarok, a sík új definíciója.
N220 A3=1 B3=0.5 C3=1.7	; Tájolás változás ORIC-cal.
N230 X-20 Y30 A4=1 B4=-2 C4=3 ORID	
N240 A3 = 0.5 B3=-0.5 C3=1	; Tájolás változás.
N250 X0 Y30 C4=1	; Pályamozgás, új sík.
	; Tájolás relatív programozással.
N260 BSPLINE X20 Z15	; Spline kezdet, tájolás relatív programozása
N270 X30 Y25 Z18	; Spline alatt aktív marad.
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, tájolás tengely programozással.
N320 G1 X0 A3=-0.123 B3=0.456 C3=2.789 B4=-1 C4=5 B5=-1 C5=2	; Pálya-mozgás, tájolás, nem állandó sík.
N330 X-20 G40	; Szerszámsugár-korrekció kikapcsolása.
N340 M30	

Példa 2: Egy CAD rendszerből létrehozott NC program (részlet) CUT3DFD-vel

Programkód	Kommentár
N01 G710	
N03 T="12"	

Programkód	Kommentár
N06 S5305 M03	
N07 G642	
; Kezdőpozícióra menet GKR-ben a szerszámhossz figyelembe vételével.	
G00 G90 X-250.62787 Y-38.37944 A=DC(253.12719) B-12.49543	
G00 G90 Z251.80052	
; Pozícionálás vége GKR-ben.	
;	
TRAORI(1)	; Tájolási transzformáció kiválasztani.
G500	
D1	
CYCLE832(0.01, _TOP_SURFACE_SMOOTH_ON + _ORI_FINISH, 1)	; felhívás CYCLE832-vel: ; kontúr tűrés = 0,01 mm, ; megmunkálási mód: Top Surface simítással, ; simítás egy tájolási tűrés beadásával, ; tájolási tűrés = 1 fok
CUT3DFD	
N08 G90 G94	
N09 G00 X-269.21195 Y128.32027 Z1.18577 A3=-.216361688 B3=.934284397 C3=-.283373051	; Az N09 ... N10 mondatok a rámeneti mozgás gyors részei a munkadarabra állandó tájolással.
N10 G00 X-251.90301 Y53.57752 Z23.85561	
N11 G01 X-247.57578 Y34.89183 Z29.52308 F50000.00000	; Az N11 ... N21 mondatokban lesz megvalósítva a rámeneti mozgás lassú része. A szerszám már közel van a munkadarabhoz, ezen kívül most a forma-készítés (pl. COMPSURF) beállítások aktívak a CYCLE832-ből (az aktív G01 miatt). Ennek az úgynevezett forma-készítés viselkedésre belendülési fázisnak a pályaútja a kontúr tűrésnek mintegy 1000-szerese kell legyen (ebben a példában tehát 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	

Programkód	Kommentár
N22 **G41** X-248.06237 Y30.32400 Z9.36695 A5=-.060431854 B5=.973045457 C5=-.222554556 A3=-.214974689 B3=.929398552 C3=-.300007025 F1061.03295	; Az N22-től a felület normál először van teljesen a össz mondattal definiálva (azaz, a felület normál az előző N21-es mondat végén van, ezzel a felület normál az N22 mondat elején és a felület normál az N22 mondat végén). Ezzel teljesül a szerszámkorrekció G41/G42-vel bekapcsolásának előfeltétele teljesül.
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	
...	

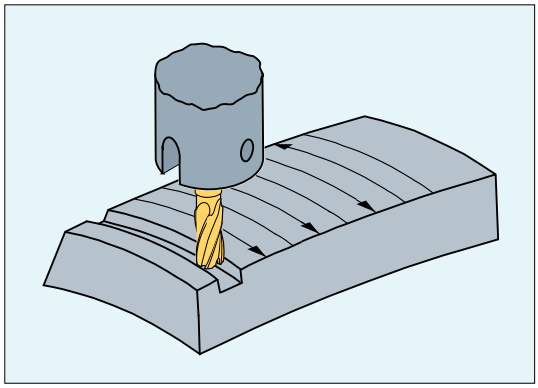
Megjegyzés

A 3D-s homlokmarás CUT3DFD-vel esetén a szerszámkorrekció G41/G42-vel bekapcsolásához szükséges a felület normálok definíciója. A G41/G42 programozása a felület normálok definíciója nélkül egy vészjelzés kiadását okozza.

További információk

3D-s homlokmarás

A 3D-marásnak ehhez a fajtájához szükséges a munkadarab-felületi 3D-pályák soronkénti leírása. A számítások szokás szerint CAM-ban kerülnek végrehajtásra a szerszámforma és a szerszámméretek figyelembevételével. A posztprocesszor a munkadarabprogramba írja - az NC-mondatok mellett - a szerszám-tájolásokat (aktív 5-tengelyes transzformációnál) és a G utasítást a kívánt 3D-szerszámkorrekcióhoz. Ezáltal a gépkezelőnek lehetősége van - az NC pályák számításához alkalmazott szerszámtól eltérően - valamivel kisebb szerszámot használni.



Kép 11-3 Homlokmarás

Rámeneti viselkedés

A rámeneteli viselkedés a szerszámsugár-korrekció 3D-s változatánál mindig a NORM.

Viselkedés a külső sarkokon

A külső sarkok a homlokmarásnál 0 sugarú körként lesznek kezelve, aminél a kör síkja az első mondat vég-érintőjére és a második mondat kezdő-érintőjére lesz felfeszítve. Ezzel lehetséges a tájolás változás a mondat átmenetnél is. Egy külső sarkon ezzel mindig egy kör lesz beszúrva kontúr-elemként. A metszéspont módszer homlokmarásnál nem áll rendelkezésre.

Viselkedés tájolás-változásnál a külső sarkokon

Az ORIC ill. ORID program utasításokkal lehet megadni, hogy a tájolás változások, amelyek a két sarkot képező mozgási mondat között lettek programozva, a beszúrt kör-mondat kezdete előtt (ORID) vagy azzal egyidőben (ORIC) legyenek végrehajtva.

Szerszámsugár-korrekció egy eltérés-szerszámra vonatkoztatva

Egy eltérés-szerszámra vonatkoztatott 3D-s szerszámsugár-korrekció a CUT3DFD utasítással lesz kiválasztva. Ezt akkor kell használni, ha a programozott kontúr egy szabvány-szerszám középpont-pályára vonatkozik és a megmunkálás egy attól eltérő szerszámmal történik. A 3D-s szerszámsugár-korrekció kiszámításánál csak az aktív szerszám sugarának kopás értékei ($TC_DP15) és az esetleg programozott OFFN és TOFFR szerszámsugár-korrekció offsetek lesznek beszámítva. Az aktív szerszám alap-sugara ($TC_DP6) **nem** lesz figyelembe véve.

A 3D-s homlokmarás CUT3DFD-vel csak a "Felület normálok simítása 3D-s homlokmarásnál"-lal kombinációban lehetséges. Ez a "Top Surface" liszensz köteles funkció CYCLE832(...) (Oldal 775) felhívásával lesz aktiválva. Az aktiválás a szerszámkorrekció G41/G42-vel bekapcsolása **előtt** kell történjen, és nem közvetlenül a szerszám használata előtt, hanem már egy akkora pályahosszra, ami megfelel a kontúr tűrés mintegy 1000-szeresének (pl. 1000 x 0,01 mm = 10 mm). A deaktiválásnak fordított sorrendben kell lefutnia: először a szerszámkorrekció kikapcsolása G40-nel, azután, a kontúr tűrés kb. 1000-szeresének megfelelő pályaút hossz után, a deaktiválás pl. CUT2D-vel (vagy mással).

A "Felület normálok simítása 3D-s homlokmarásnál" használatához szükséges még a "Felület normálok interpolációja polinomokkal" funkció engedélyezése:

MD28291 $MC_MM_SMOOTH_SURFACE_NORMALS = TRUE

Megjegyzés

A 3D-s homlokmaráshoz CUT3DFD-vel a "Top Surface"-szel kombinálva figyelembe kell venni a a "Top Surface"beállítási ajánlásokat!

A beállított adatok vizsgálatára a SIOS portálon speciális vizsgáló programok állnak rendelkezésre (https://support.industry.siemens.com/cs/ww/en/view/109738423):

11.5.3 3D-s kerületi marás egy határoló felület figyelembe vételével (CUT3DCC, CUT3DCCD)

A 3D-s kerületmarásnál a szerszámtájolás folyamatos vagy állandó változásával gyakran szerszámközéppont-pálya lesz programozva egy definiált szabványszerszámra. Mivel a gyakorlatban gyakran nem állnak rendelkezésre a megfelelő szabványszerszámok, használni lehet a szabványszerszámtól nem túlságosan eltérő (≤ 5 %) szerszámot.

A CUT3DCCD-vel egy valós különbség-szerszámra figyelembe lesz véve egy határoló felület, amelyet a programozott szabvány-szerszám írna le. Az NC program a szabványos szerszám középpontpályáját írja le.

A CUT3DCCD-vel a hengeres szerszámok alkalmazásánál figyelembe lesz véve egy határoló felület, amelyet a programozott szabvány-szerszám írna le. Az NC program a kontúrt írja le a megmunkálási felületen.

A határoló felület felület normál-vektora egy 3D-s homlokmarásnál az A4, B4, C4 és A5, B5, C5 segítségével lesz megadva.

Szintaxis

```
G41/G42 CUT3DCCD/CUT3DCC CDOF2 X... Y... Z... A4=... B4=... C4=... A5=... B5=...
C5=...
...
G40 X... Y... Z...
```

Jelentés

CUT3DCCD:	3D-s SZSK kerületi maráshoz egy határoló felület figyelembe vételével eltérés szerszámmal a szerszám középpont pályán: ráállás a határoló felületre	
CUT3DCC:	3D-s SZSK kerületi maráshoz egy határoló felület figyelembe vételével 3D-s sugárkorrekcióval: kontúr a megmunkálási felületen	
G41/G42 X... Y... Z...:	szerszámsugár-korrekciót bekapcsolni	
	G41:	szerszámsugár-korrekció a kontúrtól balra
	G42:	szerszámsugár-korrekció a kontúrtól jobbra
	Utalás: A bekapcsolás egy lineáris mondatban (G0/G1) kell történjen.	

CDOF2:	Ütközés-felügyelet kikapcsolása 3D-s kerületi marásra	
A4/5=... B4/5=... C4/5=...:	határoló sík felület normál definíciója	
	A4=... B4=... C4=...:	definíció a mondat elején
	A5=... B5=... C5=...:	definíció a mondat végén
G40 X... Y... Z...:	szerszámsugár-korrekciót kikapcsolni **Utalás:** A kikapcsolás egy lineáris mondatban (G0/G1) geometria-tengely mozgásokkal kell történjen.	

Megjegyzés

A G utasítások a 3D-s SZSK kiválasztásához a rámeneti mondatban ki lesznek értékelve, azaz tipikusan a mondatban, amelyik G41-et vagy G42-t tartalmaz.

A G41 ill. G42 lehet a korrekcióban releváns geometria-tengelyek elmozdulása nélküli mondatokban is programozva. Ebben az esetben a rámenet mondat az ilyen mondatra következő első elmozdulási mondat.

A 3D-s SZSK változat váltása aktív szerszámsugár-korrekciónál vészjelzés nélkül figyelmen kívül lesz hagyva.

Példa

Programkód	**Kommentár**
N10 $TC_DP1[1,1]=120	; hengeres maró
N20 $TC_DP6[1,1]=10	
N30 $TC_DP15[1,1]=-3	
...	
; Megmunkálás hengeres maróval és CUT3DCCD-vel	
N110 TRAORI	; Transzformáció bekapcsolása.
N120 A4=0 B4=0 C4=1	; Határoló sík felület normál definíciója a mondat elején.
N130 X0 Y0 Z0 A0 C0 T1 D1 F20000	
N140 X10 Y0 Z0 G41 **CUT3DCCD** CDOF2 G64	; 3D-s kerületi marást aktiválni a határoló felület figyelembe vételével + ütközés-felügyelet kikapcsolása.
N150 X20	
N160 X30 A45	; tompa szög ==> nincs fogás
N170 X40 A-45	; hegyes szög ==> fogás
N180 X55	
N190 Y10 Z10	; Mozgás szerszámirányban.
N200 Y20	
N210 C45	; Tiszta tájolás változás.
N220 Y30 C90	
N230 A5=-1 B5=0 C5=2 Y40	; Felület változása.
N240 Y50 G40	; Szerszámsugár-korrekció kikapcsolása.
...	

További információk

Szerszámtípus

A szerszámtípus ($TC_DP1 szerszám-paraméter) kiértékelésre kerül. Csak hengeres szárú marótípusok (henger- vagy szármaró, tórusz-maró és határesetben a hengeres süllyesztőmaró) megengedettek. Ez megfelel a 1 - 399 szerszámtípusoknak a 111 és 155...157 számok kivételével.

Szabványszerszám sarok-lekerekítéssel

A szabvány-szerszám sarok-lekerekítését a $TC_DP7 szerszámparaméter írja le. A $TC_DP16 szerszámparaméterből adódik a szabvány-szerszám és a valós szerszám sarok-lekerekítésének eltérése.

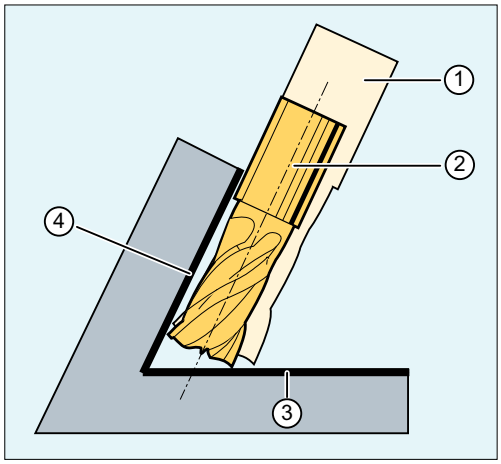
Példa: Szabvány-szerszámhoz képest csökkentett sugarú tórusz-maró

Szerszámtípus	Szár-sugár (R)	Sarok-sugár (r)
szabványszerszám sarok-lekerekítéssel	R = $TC_DP6	r = $TC_DP7
valós szerszám sarok-lekerekítéssel szerszámtípus 121 és 131 tórusz-maró (szármaró sarok lekerekítéssel)	R' = $TC_DP6 + $TC_DP15 + OFFN	r' = $TC_DP7 + $TC_DP16

Ebben a példában a $TC_DP15 + OFFN és a $TC_DP16 is negatív.

3D-s SZSK CUT3DCCD-vel: Szerszámközéppont-pálya ráállással a határoló felületig

Ha a megfelelő szabványszerszámhoz képest kisebb sugarú szerszám lesz használva, akkor a hosszirányban ráállított maró addig lesz tovább vezetve, amíg megérinti a zseb alját. Ezzel a megmunkálási felület és a határoló felület által képzett sarok annyira ki lesz forgácsolva, amennyire azt a szerszám engedi. Ez egy homlokmarás és kerületmarás kevert megmunkálási mód. A csökkentett sugarú szerszámhoz hasonlóan egy nagyobb sugarú szerszámnál az ellenkező irányban megfelelő visszaállítás történik.



① szabvány-szerszám
② kisebb sugarú szerszám fogásvétel a határoló felületig
③ határoló felület
④ megmunkálási felület

A G-csoport 22 összes más szerszámkorrekciójával ellentétben a CUT3DCCD-re megadott $TC_DP6 szerszámparaméternek nincs jelentősége a szerszámsugár számára és nem befolyásolja az eredő korrekciót. A korrekció-offset a szerszámsugár kopás-értékeinek összegéből ($TC_DP15 szerszám-paraméter) és a határoló felület függőleges offsetjének kiszámításához programozottOFFN szerszám-offsetből adódik.

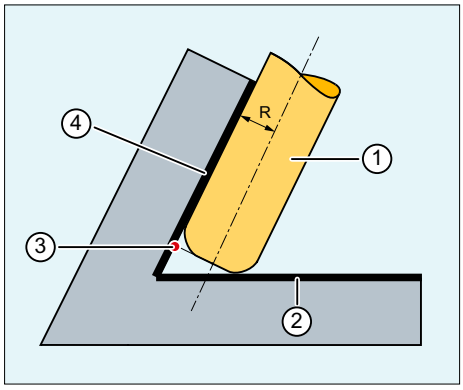
A létrehozott munkadarabprogramból nem lehet megállapítani, hogy a megmunkálandó felület jobbra vagy balra van-e a pályától. Ezért a kiindulás az eredeti szerszámra egy pozitív sugár és egy negatív kopásérték. Egy negatív kopásérték mindig egy csökkentett átmérőjű szerszámot ír le.

Hengeres szerszámok alkalmazása

A hengeres szerszámok alkalmazásánál a ráállítás csak akkor szükséges, ha a megmunkálási felület és a határoló felület egy hegyesszöget (90 foknál kisebb) képez. A tórusz-marók (származó sarok-lekerekítéssel) használatánál a hegyes- és a tompaszögnél is szükséges a szerszám ráállítása a szerszám hosszirányában.

3D-s SZSK CUT3DCC-vel: kontúr a megmunkálási felületen

Ha a CUT3DCC egy tórusz-maróval aktív, akkor a programozott pálya egy azonos átmérőjű fiktív hengeres maróra vonatkozik. Az ebből eredő pálya vonatkoztatási pont egy tórusz-maró alkalmazásánál a következő képen van ábrázolva.



① tórusz-maró
② határoló felület
③ pálya vonatkoztatási pont
④ megmunkálási felület
R szársugár (szerszámsugár)

Megengedett, hogy a megmunkálási felület és a határoló felület közötti szög egy mondaton belül hegyesszögről tompaszögre vagy fordítva változik.

A szabványszerszámmal ellentétben az alkalmazott valós szerszám lehet nagyobb vagy kisebb is. Ennél az eredő saroksugár nem lehet negatív és az eredő szerszámsugár előjel nem változhat.

A CUT3DCC-nél az NC munkadarabprogram a megmunkálási felület kontúrjára vonatkozik Ennél, akárcsak a szokásos szerszámsugár-korrekciónál, az össz-sugár lesz használva, amely a következők komponensekből tevődik össze:

- szerszámsugár ($TC_DP6 szerszámparaméter)
- kopásérték($TC_DP15 szerszámparaméter)
- a határoló felület függőleges offsetjének kiszámításához programozottOFFN szerszámoffset

A határoló felület helyzete a következő eltérésből lesz meghatározva:

szabvány-szerszám méretei - szerszámsugár ($TC_DP6 szerszámparaméter)

Advanced Surface / Top Surface

Megjegyzés

A CUT3DCC / CUT3DCCD szerszámsugár-korrekció alkalmazásánál a "Advanced Surface" vagy "Top Surface" liszensz köteles funkcióval kapcsolatban a "Advanced Surface" / "Top Surface" beállítási ajánlásokat figyelembe kell venni!

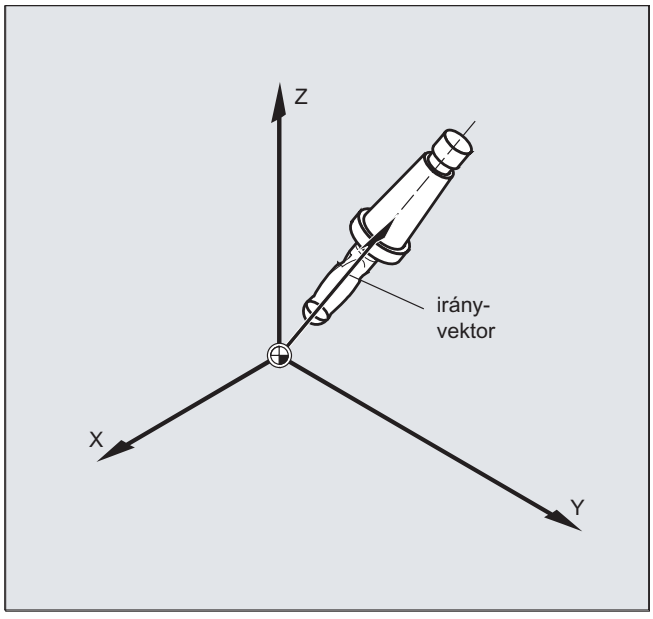
A beállított adatok vizsgálatára a SIOS portálon speciális vizsgáló programok állnak rendelkezésre:

- vizsgáló programok Advanced Surface (https://support.industry.siemens.com/cs/ww/en/view/78956392)-hez
- vizsgáló programok Top Surface (https://support.industry.siemens.com/cs/ww/en/view/109738423)-hez

---

11.6 Szerszám-tájolás (ORIC, ORID, OSOF, OSC, OSS, OSSE, ORIS, OSD, OST)

A szerszám-tájolás a szerszám geometriai helyzetét adja meg a térben. Egy 5-tengelyes megmunkálógépnél a szerszám tájolása program utasításokkal beállítható



A tájolásnak az `OSD` és `OST` által aktivált átmenet-simító mozgásai az interpolációs módtól függően a szerszám-tájoláshoz eltérően lesznek képezve.

Aktív vektor-interpolációnál a simított tájolás lefutás is vektor-interpolációval lesz interpolálva. Ezzel szemben aktív körtengely-interpolációnál a tájolás közvetlenül körtengely-mozgásokkal lesz simítva.

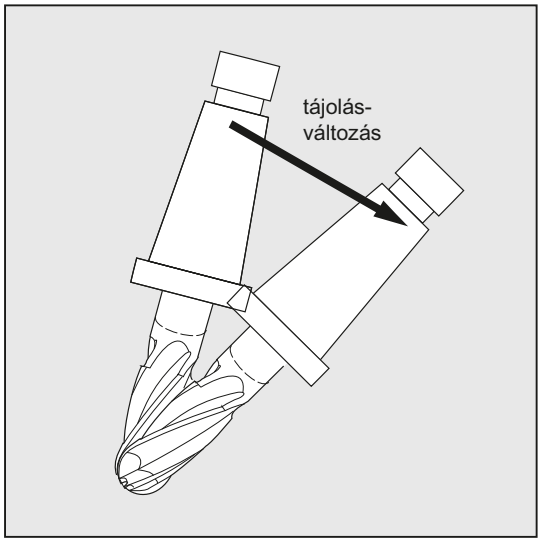
Programozás

Tájolás változások programozása:

A szerszám-tájolás változásait a következőkkel lehet programozni:

- `A`, `B`, `C` körtengelyek közvetlenprogramozása (körtengely-interpoláció)
- Euler- ill. RPY-szöggel
- irányvektor (vektor-interpoláció `A3` vagy `B3` vagy `C3`megadásával)
- `LEAD/TILT` (homlokmarás)

A vonatkozási koordinátarendszer vagy a gép-koordinátarendszer (`ORIMKS`) vagy az aktuális munkadarab-koordinátarendszer (`ORIWKS`).



Szerszámtájolás programozása:

ORIC:	tájolás és pályamozgás párhuzamosan
ORID:	tájolás és pályamozgás egymásután
OSOF:	nincs tájolás simítás
OSC:	tájolás állandó
OSS:	tájolás simítás csak a mondat elején
OSSE:	tájolás simítás a mondat elején és végén
ORIS:	tájolás változás sebessége bekapcsolt tájolás simításnál fok per mm-ben (érvényes OSS és OSSE-re)
OSD:	tájolás átmenet simítása az átmenet simítás hosszának megadásával beállítási adattal: SD42674 $SC_ORI_SMOOTH_DIST
OST:	tájolás átmenet simítása a szögtűrés megadásával fokban vektor-interpolációval a beállítási adattal: SD42676 $SC_ORI_SMOOTH_TOL körtengely-interpolációnál a megadott tűrés a tájolótengelyek maximális eltéréseként értendő

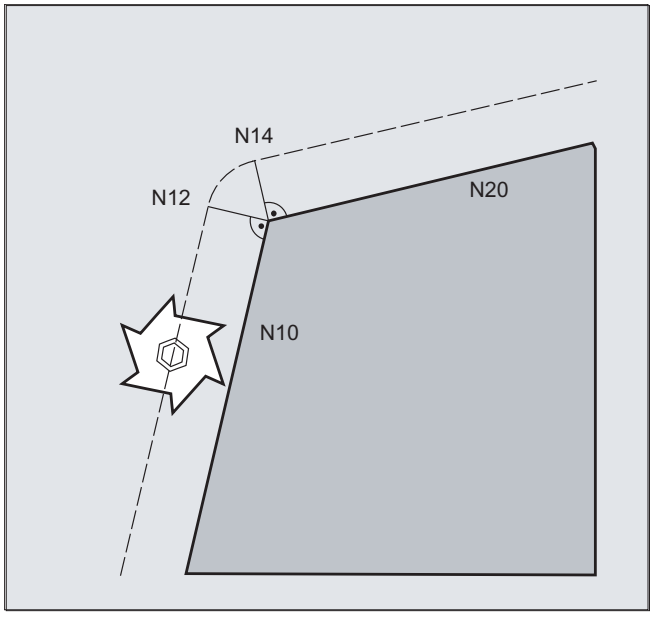
Megjegyzés

A szerszámtájolás átmenet simítás összes utasítása (OSOF, OSC, OSS, OSSE, OSD und OST) a G-csoport 34-ben van összefogva.. Ezek modálisan hatnak és ezen utasítások közül mindig csak egy lehet hatásos.

Példák

Példa 1: ORIC

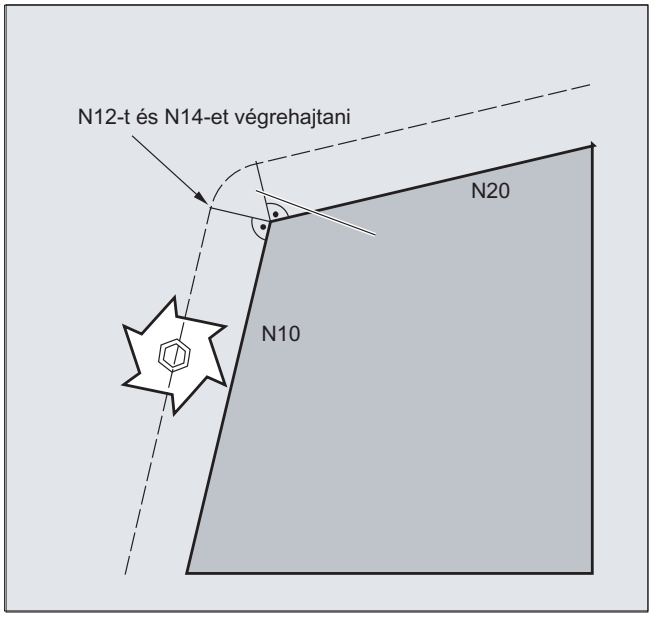
Ha az `N10` és `N20` mozgás-mondatok között két vagy több mondat van tájolás változással (pl. `A2=... B2=... C2=...`) programozva és `ORIC` aktív, akkor a beszúrt kör-mondat fel lesz osztva a közbenső mondatokra a szög változás.



Programkód	Kommentár
ORIC	
N8 A2=… B2=… C2=…	
N10 X… Y… Z…	
N12 C2=… B2=…	; A külső sarkon beszúrt kör-mondat a tájolás változásnak megfelelően felosztódik az N12 és N14 között. A kör-mozgás és a tájolás váltás itt párhuzamosan kerül végrehajtásra.
N14 C2=… B2=…	
N20 X =…Y=… Z=… G1 F200	

Példa 2: ORID

Ha az `ORID` aktív, akkor minden mondat a két mozgási mondat között az első mozgási mondat végén lesz végrehajtva. A kör-mondat az állandó tájolással közvetlenül a második mozgási mondat előtt lesz végrehajtva.



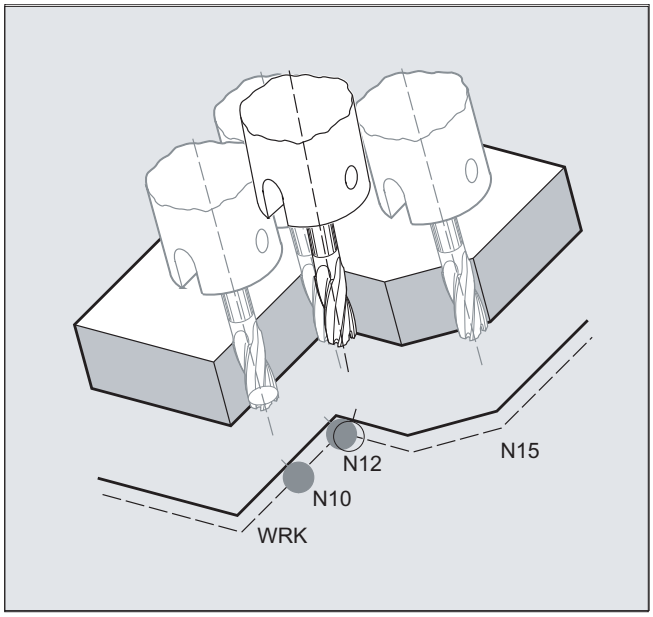
Programkód	Kommentár
ORID	
N8 A2=… B2=… C2=…	
N10 X… Y… Z…	
N12 A2=… B2=… C2=…	; Az N12 és N14 mondat az N10 végén lesz végrehajtva. Ezután lesz a kör-mondat az aktuális tájolással megtéve.
N14 M20	; segédfunkciók stb.
N20 X… Y… Z…	

Megjegyzés

A tájolás változás módjára egy külső sarkon az a program utasítás mérvadó, amelyik egy külső sarok első mozgás-mondatában aktív.

Tájolás változás nélkül: Ha a tájolás a mondathatáron nem változik, akkor a szerszám-keresztmetszet egy kör, ami mindkét kontúrt érinti.

Példa 3: A tájolás változása egy belső sarkon



Programkód

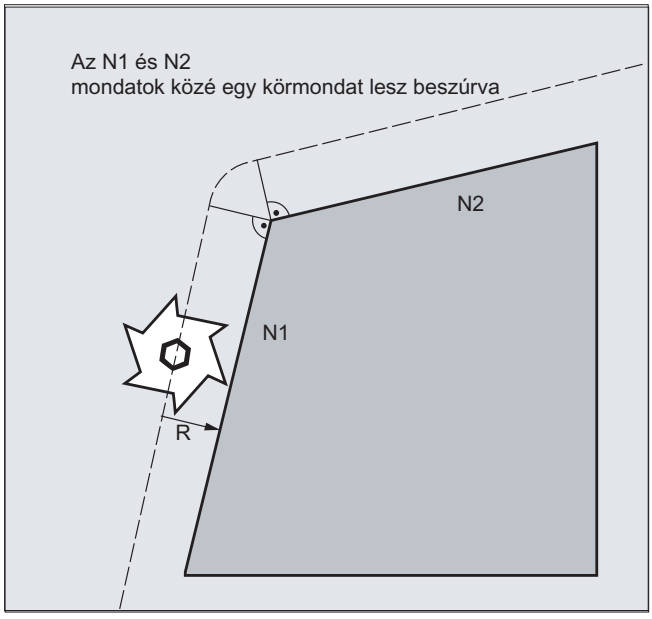
```
ORIC
N10 X …Y… Z… G1 F500
N12 X …Y… Z… A2=… B2=… C2=…
N15 X …Y… Z… A2=… B2=… C2=…
```

További információk

Viselkedés a külső sarkokon

Egy külső sarkon mindig egy kör-mondat lesz beszúrva a maró sugárral.

Az `ORIC` ill. `ORID` program utasításokkal lehet megadni, hogy az `N1` és `N2` mondatok között programozott tájolás változások a beszúrt kör-mondat előtt vagy azzal egyidőben legyenek végrehajtva.



Ha a külső sarkokon tájolás változás szükséges, ez választhatóan az interpolációval együtt vagy a pályamozgástól külön történhet.

Az `ORID`-nál először a beillesztett pályamozgás nélküli mondatok lesznek végrehajtva. A kör-mondat közvetlenül a sarkot képező két mozgási mondat előtt lesz beszúrva.

Ha egy külső sarkon több tájolás-mondat van beszúrva és az `ORIC` ki van választva, akkor a kör-mozgás el lesz osztva ezekre az egyes beszúrt mondatok tájolás változás értékeinek megfelelően.

Tájolás átmenet simítása OSD és OST által

Az átmenet simításánál `G642`-vel a maximális eltérés a kontúrtengelyekre és a tájolótengelyekre nem lehet nagyon különböző. A kettő közül a kisebb tűrés határozza meg az átmenet simító mozgás formáját ill. a szögtűrést, a tájolás lefutását viszonylag erősen simítani anélkül, hogy ennél nagyobb kontúreltéréseket kelljen elviselni.

Az `OSD` ill. `OST` i aktiválásával lehetséges egy előre megadott átmeneti hosszal és szögtűréssel a tájolás lefutásának nagyon csekély eltéréseit jelentős kontúreltérés nélkül "nagyvonalúan" simítani.

Megjegyzés

A `G642`-vel történő kontúr átmenet simítással (és tájolás lefutással) szemben a tájolás simításánál az `OSD` ill. `OST` által nem lesz külön mondat képezve, hanem az átmenet simító mozgás közvetlenül a programozott eredeti mondatokba lesz beszúrva.

Az `OSD` ill. `OST` használatával nem simíthatók olyan mondat-átmenetek, amelyeknél a szerszám-tájolásra interpolációs mód váltása (vektor –> körtengely, körtengely–> vektor) történik. Ezeket a mondat-átmeneteket adott esetben `G641`, `G642` ill. `G643` átmenet simítási funkciókkal lehet simítani.

11.7 Szabad D-szám megadás, vágóélszám

11.7.1 Szabad D-szám megadás, vágóél-szám (CE cím)

D-szám

A D-számok korrekciószámként alkalmazhatók. Kiegészítőleg a CE címmel a vágóél számát lehet megcímezni. A $TC_DPCE rendszerváltozóval lehet írni a vágóél-számot.

elő-beállítás: korrekció sz. == vágóél sz.

Gépadatokkal lehet megadni a D-számok maximális számát (vágóél-számok) és a maximális vágóélszámot szerszámonként (→ gépgyártó). A következő utasításoknak csak akkor van értelme, ha a maximális vágóél-szám (MD 18105) nagyobb, mint az élek száma szerszámonként (MD 18106). Vegyük ehhez figyelembe a gépgyártó tájékoztatásait.

Megjegyzés

A relatív D-szám megadás mellett a D-számokat 'lapos' ill. 'abszolút' D-számokként (1-32000) egy T-számra hivatkozás nélkül is meg lehet adni (a 'lapos D-szám struktúra' funkcióban).

Irodalom

Funktionshandbuch Grundfunktionen; Werkzeugkorrektur (W1)

11.7.2 Szabad D-szám megadás: D-számot vizsgálni (CHKDNO)

A `CKKDNO` utasítással megvizsgáljuk, hogy a meglevő D-számok egyértelműen vannak-e megadva. Egy TO-egységen belül definiált összes szerszám D-számai csak egyszer fordulhatnak elő. A tartalék-szerszámok nincsenek figyelembe véve.

Szintaxis

```
state=CHKDNO(Tno1,Tno2,Dno)
```

Jelentés

`state`:	=TRUE:	A D-számok a vizsgált tartományra egyértelműen meg vannak adva.
	=FALSE:	D-szám ütközés van vagy a paraméterezés érvénytelen. A Tno1, Tno2 és Dno átadja a paramétereket, amelyek az ütközést okozták. Ezeket az adatokat a munkadarab-programban ki lehet értékelni.
`CHKDNO(Tno1,Tno2)`:	A megnevezett szerszámok minden D-száma megvizsgálásra kerül.	

CHKDNO(Tno1):	A Tno1 minden D-száma az összes szerszámra megvizsgálásra kerül.
CHKDNO:	Minden szerszám minden D-száma az összes szerszámra megvizsgálásra kerül.

11.7.3 Szabad D-szám megadás: D-számot átnevezni (GETDNO, SETDNO)

A D-számokat egyértelműen kell megadni. Egy szerszám két különböző vágóélének nem lehet azonos D-száma.

GETDNO

Ez az utasítás meghatározza egy t T-számú szerszám egy meghatározott (ce) élének D-számát. Ha a beadott paraméterekhez nincs D-szám, d=0 lesz beállítva. Ha a D-szám érvénytelen, egy 32000-nél nagyobb érték lesz visszaadva.

SETDNO

Ezzel az utasítással rendeljük hozzá egy t szerszám ce vágóéle D-számához a d értéket. A state visszaadja ennek az utasításnak az eredményét (TRUE vagy FALSE). Ha a beadott paraméterekhez nem tartozik adatkészlet, FALSE lesz visszaadva A szintaxis-hiba vészjelzést eredményez. A D-számot nem lehet közvetlenül 0-ra állítani.

Szintaxis

```
d = GETDNO (t,ce)
state = SETDNO (t,ce,d)
```

Jelentés

d:	szerszám vágóél D-száma
t:	szerszám T-száma
ce:	szerszám vágóél-száma (CE-szám)
state:	megadja, hogy az utasítás hibátlanul végre lett-e hajtva (TRUE vagy FALSE)

Példa egy D-szám átnevezésére

Programozás	Kommentár
$TC_DP2[1,2] = 120	
$TC_DP3[1,2] = 5.5	
$TC_DPCE[1,2] = 3	; CE vágóélszám
...	
N10 def int DNrAlt, DNrNeu = 17	
N20 DNrAlt = GETDNO(1,3)	
N30 SETDNO(1,3,DNrNeu)	

Ezzel a CE=3 vágóélhez a 17 új D-érték lesz hozzárendelve. Mostantól ennek a vágóélnek adatai a 17-es D-számmal lesznek felhívva; úgy a rendszer-paramétereknél, mint a programozásnál NC-címekkel.

11.7.4 Szabad D-szám megadás: T-számot a megadott D-számhoz megállapítani (GETACTTD)

A `GETACTTD` utasítással megállapítjuk egy abszolút D-számhoz a hozzátartozó T-számot. Nincs egyértelműség vizsgálat. Ha egy TO-egységen belül több azonos D-szám van, az első megtalált szerszám T-száma lesz visszaadva. A "lapos" D-szám esetén az utasítás alkalmazásának nincs értelme, mivel itt mindig az 1-es érték lesz visszaadva (nincs T-szám az adatokban).

Szintaxis

```
status=GETACTTD(Tnr,Dnr)
```

Jelentés

`Dnr`:	D-szám, amelyhez a T-számot kell keresni	
`Tnr`:	megtalált T-szám	
`status`:	érték:	jelentés:
	0	A T-számot megtalálta. A Tnr tartalmazza a T-szám értékét.
	-1	A megadott D-számhoz nincs T-szám; Tnr=0.
	-2	A D-szám nem abszolút. A Tnr tartalmazza az első megtalált szerszám számát, amely D-száma Dnr értékű.
	-5	A funkciót más okból nem lehetett végrehajtani.

11.7.5 Szabad D-szám megadás: D-számot érvénytelenné tenni (DZERO)

A `DZERO` utasítás az átszerelés alatt ad támogatást. Az így megjelölt korrekcióadatokat a `CHKDNO` utasítás nem vizsgálja. Hogy ezek ismét elérhetők legyenek, a D-számokat ismét be kell állítani a `SETDNO`-val.

Szintaxis

```
DZERO
```

Jelentés

`DZERO`:	A TO-egység valamennyi D-számát érvénytelenné teszi

11.8 Szerszámtartó kinematika

Előfeltételek

Egy szerszámtartó egy szerszámot csak akkor tud a tér minden lehetséges irányában tájolni, ha

- van két V_1 és V_2 forgástengelye
- a forgástengelyek egymásra merőlegesek
- a szerszám hossztengelye a második V_2 forgástengelyre merőleges.

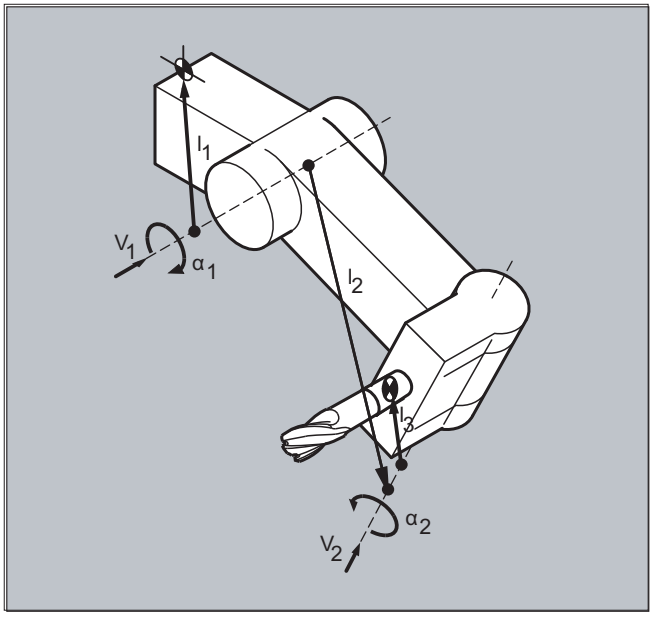
Kiegészítőleg érvényes s gépeknél, amelyeknél mindkét tengely forgatja az asztalt, hogy :

- a szerszám tájolásnak az első V_1 forgástengelyre merőlegesen kell állni.

Funkció

A szerszámtartó kinematika maximum két forgótengellyel v_1 vagy v_2 a $TC_CARR1[m] ... $TC_CARR17[m] 17 rendszerváltozóval lesz írva. A szerszámtartó leírása a következőkből áll:

- a szerszámtartó vonatkoztatási pontjának I_1vektoriális távolsága az első forgástengelytől, az első és második forgástengely I_2vektoriális távolsága, a szerszámtartó vonatkoztatási pontjának I_3 vektoriális távolsága a második forgástengelytől.
- a két V_1 és V_2 forgástengely irányvektorai
- α1, α2 forgásszögek a két tengely körül. A forgásszög a forgástengely vektor irányába tekintve az óramutató járása irányában pozitív.



A **felbontott kinematikájú** gépek számára (a szerszám és a munkadarab is forgatható) a rendszerváltozók a $TC_CARR18[m] ... $TC_CARR23[m] bevitelekkel lettek kibővítve.

Paraméter

Rendszerváltozók funkciói tájolható szerszámtartókhoz			
jelölés	**x komponens**	**y komponens**	**z komponens**
l_1 offsetvektor	$TC_CARR1[m]	$TC_CARR2[m]	$TC_CARR3[m]
l_2 offsetvektor	$TC_CARR4[m]	$TC_CARR5[m]	$TC_CARR6[m]
v_1 forgástengely	$TC_CARR7[m]	$TC_CARR8[m]	$TC_CARR9[m]
v_2 forgástengely	$TC_CARR10[m]	$TC_CARR11[m]	$TC_CARR12[m]
α_1 forgásszög α_2 forgásszög	$TC_CARR13[m] $TC_CARR14[m]		
l_3 offsetvektor	$TC_CARR15[m]	$TC_CARR16[m]	$TC_CARR17[m]

Rendszerváltozók bővítései tájolható szerszámtartókhoz			
jelölés	**x komponens**	**y komponens**	**z komponens**
l_4 offsetvektor	$TC_CARR18[m]	$TC_CARR19[m]	$TC_CARR20[m]
tengelyjelölő forgástengely v_1 forgástengely v_2	$v_1$és v_2 forgástengelyek jelölői (alapesetben nulla) $TC_CARR21[m] $TC_CARR22[m]		
kinematika-típus	$TC_CARR23[m]		
Tool **P**art **M**ixed mode	kinematika-típus-T ->	kinematika-típus-P ->	kinematika-típus-M ->
	csak a szerszám forgatható (alapeset)	csak a munkadarab forgatható	munkadarab & szerszám forgathatóak
offset forgástengely v_1 forgástengely v_2	$v_1$és v_2 forgástengelyek szöge fokban az alaphelyzet felvételénél $TC_CARR24[m] $TC_CARR25[m]		
szög-offset forgástengely v_1 forgástengely v_2	$v_1$és v_2 forgástengelyek Hirth-fogazás offset fokban $TC_CARR26[m] $TC_CARR27[m]		
szög-inkremens v_1 forgástengely v_2 forgástengely	$v_1$und $v_2$forgástengelyek Hirth-fogazás inkremens fokban $TC_CARR28[m] $TC_CARR29[m]		
min.-pozíció forgástengely v_1 forgástengely v_2	$v_1$és $v_2$forgástengelyek szoftver-határ minimum-pozícióra $TC_CARR30[m] $TC_CARR31[m]		
max.-pozíció forgástengely v_1 forgástengely v_2	$v_1$és $v_2$forgástengelyek szoftver-határ maximum-pozícióra $TC_CARR32[m] $TC_CARR33[m]		
szerszámtartó neve	Egy szerszámtartó egy szám helyett kaphat egy nevet. $TC_CARR34[m]		
felhasználó: tengelynév 1 tengelynév 2 jelölő **pozíció**	Szándékolt alkalmazás a felhasználói mérőciklusokban. $TC_CARR35[m] $TC_CARR36[m] $TC_CARR37[m]		
	$TC_CARR38[m]	$TC_CARR39[m]	$TC_CARR40[m]
finom-eltolás	paraméter, amit **az értékekhez az alapparaméterekben hozzáadni** lehet.		
l_1 offsetvektor	$TC_CARR41[m]	$TC_CARR42[m]	$TC_CARR43[m]
l_2 offsetvektor	$TC_CARR44[m]	$TC_CARR45[m]	$TC_CARR46[m]

Rendszerváltozók bővítései tájolható szerszámtartókhoz			
l_3 offsetvektor	$TC_CARR55[m]	$TC_CARR56[m]	$TC_CARR57[m]
l_4 offsetvektor	$TC_CARR58[m]	$TC_CARR59[m]	$TC_CARR60[m]
v_1 forgástengely	$TC_CARR64[m]		
v_2 forgástengely	$TC_CARR65[m]		

Megjegyzés

Magyarázat a paraméterekhez

Az "m" a leírandó szerszámtartó számát adja meg.

$TC_CARR47 ... $TC_CARR54 ill. $TC_CARR61 ... $TC_CARR63 nincsenek definiálva és ezekhez olvasása vagy írás hozzáférés kísérlete vészjelzést okoz.

A távolság vektorok kezdő- és végpontjai a tengelyeken szabadon választhatók. Mindkét tengely α_1, $\alpha_2$forgásszöge a szerszámtartók alapállapotában 0°-nak definiált. Egy szerszámtartó kinematikát így tetszőleges sok lehetőségre le lehet írni.

A szerszámtartó csak egy forgástengellyel vagy anélkül az egyik vagy mindkét irányvektor nullára állításával írható le.
Egy forgástengely nélküli szerszámtartónál a távolságvektorok kiegészítő szerszámkorrekcióként hatnak, amelyek komponenseit a megmunkálási síkok átkapcsolása (G17 ... G19)° nem befolyásolja.

Paraméterek bővítése

Forgástengelyek paraméterei

A rendszerváltozók ki lettek bővítve $TC_CARR24[m] ... $TC_CARR33[m]-mel amelyek leírása a következő:

offset v_1, v_2 forgástengelyekre	A v_1 vagy v_2 forgástengelyek pozíciójának változása a tájolható szerszámtartó alapállásánál.
szögoffset/ szöginkremens v_1, v_2 forgástengelyek	A v_1 és v_2 forgástengelyek offsetje vagy Hirth-fogazásának szöginkremense. A programozott vagy számított szög a legközelebbi értékre lesz kerekítve, amelyik egészszámú n-nél a phi = s + n * d-ből adódik.
minimum- és maximum-pozíció v_1, v_2 forgástengelyek	A v1 és v2 forgástengelyek minimum-/maximum pozíció határszöge (szoftverhatár).

Felhasználói paraméterek

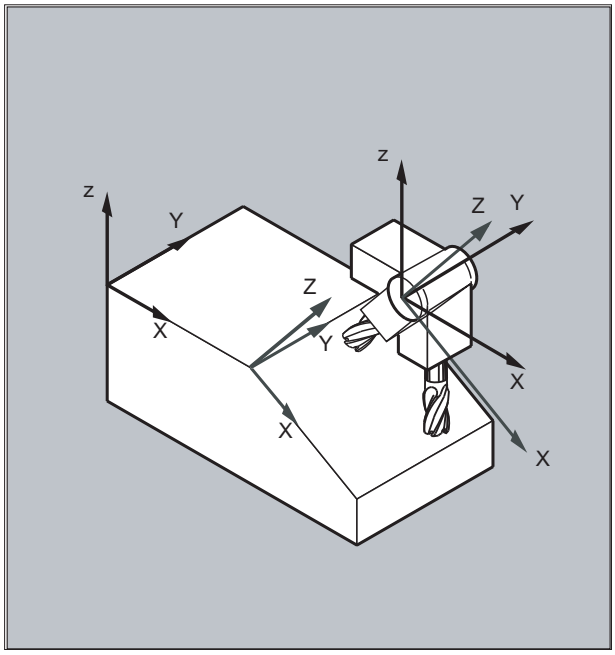
A $TC_CARR34 ... $TC_CARR40 a felhasználónak szabad rendelkezésére álló paraméterek és a SW 6.4-ig az NCK-ban nincsenek kiértékelve vagy nincs jelentésük.

Finom-eltolás paraméter

A $TC_CARR41 ... $TC_CARR65 finomeltolás paraméterek, amelyet a bázis-paraméterek értékeihez lehet hozzáadni. Egy alap-paraméterhez hozzárendelt finomeltolás értéke a paraméterhez 40 hozzáadásából adódik.

Példa

A következő példában alkalmazott szerszámtartót egy forgatással az Y-tengely körül tökéletesen le lehet írni.



Programkód	Kommentár
N10 $TC_CARR8[1]=1	; A szerszámtartó 1 első forgástengelye Y-komponensének definíciója.
N20 $TC_DP1[1,1]=120	; Egy származó definíciója.
N30 $TC_DP3[1,1]=20	; Egy származó definíciója 20 mm-es hosszal.
N40 $TC_DP6[1,1]=5	; Egy származó definíciója 5 mm-es sugárral.
N50 ROT Y37	; Frame definíció 37°-os forgatással az Y-tengely körül.
N60 X0 Y0 Z0 F10000	; Kiinduló pozícióra menni.
N70 G42 CUT2DF TCOFR TCARR=1 T1 D1 X10	; Sugárkorrekciót, szerszámhosszkorrekciót az elforgatott frameben beállítani, szerszámtartó 1-et, szerszám 1-et kiválasztani.
N80 X40	; A megmunkálást a 37°-os elforgatással elvégezni.
N90 Y40	
N100 X0	
N110 Y0	
N120 M30	

További információk

Felbontott kinematika

A felbontott kinematikájú gépek számára (a szerszám és a munkadarab is forgatható) a rendszerváltozók a `$TC_CARR18[m]` ... `$TC_CARR23[m]`-mel lettek bővítve és a leírásuk a következő:

A forgatható szerszámasztal a következőkből áll:

- a második V_2 forgástengely vektoriális távolsága a harmadik körtengely I_4 forgatható szerszámasztal vonatkoztatási pontjához

A körtengelyek a következőkből állnak:

- a V_1és V_2 forgástengelyek csatornajelölői, amelyek pozícióihoz adott esetben a tájolható szerszámtartó tájolásának meghatározásához hozzá kell férni.

A kinematika-típus T, P vagy M értéke:

- kinematika-típus T: csak a szerszám forgatható
- kinematika-típus P: csak a munkadarab forgatható
- kinematika-típus M: munkadarab és szerszám forgathatóak

Szerszámtartó adatok törlése

A `$TC_CARR1[0]=0`-val minden szerszámtartó adatot törölni lehet.

A kinematika típus `$TC_CARR23[T]=T` a három megengedett nagy- vagy kisbetű (T,P,M) egyikével kell legyen feltöltve és ezért nem szabad törölni.

Szerszámtartó adatok változtatása

Minden beírt értéket egy új érték hozzárendelésével a munkadarabprogramban változtatni lehet. A T, P és M karaktereken kívül minden más karakter esetén a tájolható szerszámtartó aktiválásának kísérlete vészjelzést okoz.

Szerszámtartó adatok olvasása

Minden beírt értéket egy változó hozzárendelésével a munkadarabprogramban olvasni lehet.

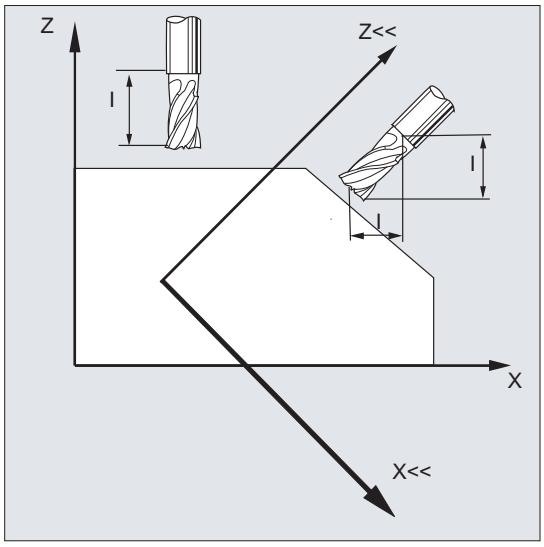
Finom-eltolások

Egy nem megengedett finom-eltolás érték csak akkor lesz felismerve, ha egy tájolható szerszámtartó lesz aktiválva, amelyik egy ilyen értéket tartalmaz és egyidejűleg az SD42974 $SC_TOCARR_FINE_CORRECTION = TRUE.

A megengedett finom-eltolások összege gépadattal egy maximálisan megengedett értékre lesz határolva.

11.9 Szerszámhossz-korrekció tájolható szerszámtartókra (TCARR, TCOABS, TCOFR, TCOFRX, TCOFRY, TCOFRZ)

A szerszám megváltoztatott térbeli tájolásával megváltoznak annak szerszámhossz-komponensei is.



Átszerelés után, pl. kézi beállítással vagy fix térbeli beállítású szerszámtartó cseréjével újra meg kell határozni a szerszámhossz-komponenseket. Ez a `TCOABS` és `TCOFR` utasításokkal történik.

Egy aktív frame tájolható szerszámtartójánál a szerszám-választásnál `TCOFRZ`, `TCOFRY` és `TCOFRX`-szel meghatározható az irány, amerre a szerszám mutasson.

Szintaxis

```
TCARR=[<m>]
TCOABS
TCOFR
TCOFRZ
TCOFRY
TCOFRX
```

Jelentés

`TCARR=[<m>]`:	"m" számú szerszámtartó lekérése
`TCOABS`:	szerszámhossz-komponensek kiszámítása az aktuális szerszámtartó-tájolásból
`TCOFR`:	szerszámhossz-komponensek meghatározása az aktív frame orientálásból
`TCOFRZ`:	tájolható szerszámtartó az aktív frame-ből, amelyiknek szerszáma a Z-irányba mutat
`TCOFRY`:	tájolható szerszámtartó az aktív frame-ből, amelyiknek szerszáma az Y-irányba mutat
`TCOFRX`:	tájolható szerszámtartó az aktív frame-ből, amelyiknek szerszáma az X-irányba mutat

További információk

Szerszámhossz-korrekció tartó-tájolásból (TCOABS)

A `TCOABS` kiszámítja a szerszámhossz-korrekciót a szerszámtartó aktuális tájolási szögéből; amely az $TC_CARR13 és $TC_CARR14 rendszerváltozóban került letételre.

A szerszámtartó-kinematika definíciójához rendszerváltozókkal lásd " Szerszámtartó-kinematika (Oldal 445) ".

A szerszámhossz-korrekció új kiszámításához frame-váltásnál a szerszámot ismét fel kell hívni.

Szerszámirány az aktív frame-ből

A tájolható szerszámtartót be lehet úgy állítani, hogy a szerszám a következő irányokba mutat:

- `TCOFR` ill. `TCOFRZ` esetén a Z-irányba
- `TCOFRY` esetén az Y-irányba
- `TCOFRX` esetén az X-irányba

Az átkapcsolás a `TCOFR` és `TCOABS` között a szerszámhossz-korrekciók újra számítását okozza.

Szerszámtartó lekérése (TCARR)

A `TCARR`R-ral az m számú szerszámtartó geometriai adatai lekérésre kerülnek (korrekció-tároló).

Az m=0-val az aktív szerszámtartó kiválasztása megszűnik.

A szerszámtartó geometriai adatai csak egy szerszám felhívása után lesznek aktívak. A kiválasztott szerszám egy szerszámtartó cseréjén túl is aktív marad.

A szerszámtartó aktuális geometriai adatait a munkadarabprogramban is meg lehet adni a megfelelő rendszerváltozón keresztül.

Szerszámhossz-korrekció újra-számítása (TCOABS) frame-váltásnál

A szerszámhossz-korrekció új kiszámításához frame-váltásnál a szerszámot ismét fel kell hívni.

Megjegyzés

A szerszám-tájolást az aktív frame-hez kézzel kell hozzáilleszteni.

A szerszámhossz-korrekció kiszámításánál egy közbenső lépésben a szerszámtartó forgásszöge is kiszámításra kerül. Mivel két forgótengelyes szerszámtartóknál általában két forgásszög-pár létezik, amelyekkel a szerszám-orientálás az aktív frame-re ráilleszthető, a

rendszerváltozókban letett forgásszög-értékeknek legalább megközelítőleg meg kell egyezniük a mechanikusan beállított forgásszögekkel.

Megjegyzés

Szerszámtájolás

A vezérlés nem tudja a frame-tájoláson keresztül kiszámított forgásszöget a gépen beállíthatóságára ellenőrizni.

Ha a szerszámtartó forgástengelyei szerkezetileg úgy vannak elrendezve, hogy a frame-tájolással kiszámított szerszám-tájolás nem érhető el, akkor egy vészjelzés kerül kiadásra.

A szerszám-finomkorrekció és a szerszámhossz-korrekció funkcionalitásainak kombinációja mozgó szerszámtartóknál nem megengedett. A két funkció egyidejű felhívásának megkísérlésénél hibakijelzés történik.

A `TOFRAME`-mel lehetőség van arra, hogy egy frame-t a kiválasztott szerszámtartó orientálási iránya alapján meghatározhassunk. Több információt erre a "Frame-k"fejezetben.

Aktív tájolás-transzformációnál (3-, 4-, 5-tengely-transzformáció) egy nullától eltérő tájolású szerszámtartót fel lehet hívni anélkül, hogy emellett egy vészjelzés kerülne kiadásra.

Átadási paraméterek szabvány- és mérő-ciklusokra

A szabvány- és mérő-ciklusok paramétereire definiált értéktartományok érvényesek.

Szögértékeknél az értéktartomány a következők szerint van meghatározva:

- forgatás az 1. geometriai tengely körül: -180 foktól +180 fokig
- forgatás a 2. geometriai tengely körül: -90 foktól +90 fokig
- forgatás az 3. geometriai tengely körül: -180 foktól +180 fokig

Lásd Frame-k fejezet, "Programozható forgatás (ROT, AROT, RPL)".

Megjegyzés

Szögértékek átadásánál egy szabvány- vagy mérő-ciklusnak figyelembe kell venni:

Az NC számítási felbontásánál kisebb értékeket nullára kell kerekíteni!

Az NC számítási felbontása szögpozíciókra a következő gépadatban van megadva:

MD10210 $MN_INT_INCR_PER_DEG

11.10 Online szerszámhossz-korrekció (TOFFON, TOFFOF)

A $AA_TOFF[ ] rendszerváltozóval a hatásos szerszámhosszak a három szerszámiránynak megfelelően háromdimenziósan valósidőben módosíthatóak.

Indexként <n> a három geometria-tengely jelölője van használva. Így az aktív korrekciós irányok száma az azonos időben aktív geometria-tengelyek által rögzítve van.

Az összes korrekció lehet egyidőben aktív.

Az online szerszámhossz-korrekció funkció használható a következőknél:

- tájolási transzformáció TRAORI
- tájolható szerszámtartó TCARR

Megjegyzés

Az online szerszámhossz-korrekció egy **opció,** amit előbb engedélyezni kell. Ennek a funkciónak csak az aktív tájolás-transzformációval vagy a tájolható szerszámtartóval kapcsolatban van értelme.

Szintaxis

```
TRAORI
TOFFON(<korrekció irány>[,<offset-érték>])
WHEN TRUE DO $AA_TOFF[<korrekció irány>]                ; szinkronakciókban
...
TOFFOF(<korrekció irány>)
```

Az online szerszámhossz-korrekció programozásához a mozgásszinkron akciókban további magyarázatok, lásd "Szinkron-akciók (Oldal 601)".

Jelentés

TOFFON:	online szerszámhossz-korrekciót **aktiválni**	
	<korrekció irány>:	szerszámirány (X, Y, Z), amelyben az online szerszámhossz-korrekció hatásos kell legyen
	<offset-érték>:	Az aktiválásnál a megfelelő korrekció-irányra meg elehet adni egy offset-értéket, ami rögtön elmozdulást okoz.
TOFFOF:	online szerszámhossz-korrekció **kikapcsolása** A megfelelő korrekciós-értékek törölve lesznek és egy előrefutás-állj lesz kiváltva.	

Példák

Példa 1: Szerszámhossz-korrekció kiválasztása

Programkód	Kommentár
MD21190 $MC_TOFF_MODE =1	; Abszolút értékekre menet.
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)	; Transzformáció be.
N20 TOFFON(Z)	; Online szerszámhossz-korrekció aktiválása Z szerszámiránya.
N30 WHEN TRUE DO $AA_TOFF[Z]=10 G4 F5	; Szerszámhossz-korrekció 10 interpolálása Z szerszámiránya.
...	
N100 XOFFSET=$AA_TOFF_VAL[X]	; Aktuális korrekció hozzárendelés X-irányba.
N120 TOFFON(X,-XOFFSET) G4 F5	; A szerszámhossz-korrekció X szerszámirányban ismét 0-ra lesz visszavéve.

Példa 2: Szerszámhossz-korrekció kikapcsolása

Programkód	Kommentár
N10 TRAORI(1)	; Transzformáció be.
N20 TOFFON(X)	; Online szerszámhossz-korrekció aktiválása X szerszámiránya.
N30 WHEN TRUE DO $AA_TOFF[X]=10 G4 F5	; Szerszámhossz-korrekció 10 interpolálása X szerszámiránya.
...	
N80 TOFFOF(X)	; X szerszámirány pozíció-offset törlése:
	...$AA_TOFF[X]=0
	tengely nem mozdul el
	az aktuális pozícióhoz MKR-ben a pozíció-offset hozzá lesz számítva az aktuális tájolásnak megfelelően

További információk

Mondat-feldolgozás

A mondat-feldolgozásnál az előrefutásban a főfutásban hatásos aktuális szerszámhossz-offset figyelembe lesz véve. A maximálisan megengedett tengelysebességek messzemenő

kihasználásához szükséges a mondatfeldolgozás megállítása `STOPRE` előrefutás-állj-jal, amíg a szerszám-offset fel lesz véve.

A szerszám-offset az előrefutás időpontjában akkor ismert, ha a szerszámhossz-korrekció a program-start után már nincs változtatva, vagy ha a szerszámhossz-korrekciók változtatása után több mondat lett feldolgozva, mint amennyit az IPO-puffer az előrefutás és főfutás között fel tud venni.

$AA_TOFF_PREP_DIFF változó

Az interpolátorban aktuálisan hatásos korrekció és a mondat-feldolgozás időpontjában hatásos korrekció különbségének értékét a $AA_TOFF_PREP_DIFF[<n>] általzóban lekérdezhető.

Gépadatok és beállítási adatok beállítása

Az online szerszámhossz-korrekció a következő gépadatok állnak rendelkezésre:

- MD20610 $MC_ADD_MOVE_ACCEL_RESERVE (gyorsulás-tartalék átlapolt mozgásra)
- MD21190 $MC_TOFF_MODE
  IA $AA_TOFF[<n>] rendszerváltozó tartalma abszolút értékként lesz megtéve vagy kiintegrálva.
- MD21194 $MC_TOFF_VELO (online szerszámhossz-korrekció sebessége)
- MD21196 $MC_TOFF_ACCEL (online szerszámhossz-korrekció gyorsulása)
- beállítási adatok a határértékek megadásához:
  SD42970 $SC_TOFF_LIMIT (online szerszámhossz-korrekció értékek felső határa)

Irodalom:
Funktionshandbuch Sonderfunktionen; F2: Többtengelyes transzformációk

11.11 Korrekcióadatok módosítása forgatható szerszámoknál

11.11.1 Tájolásokat kiszámítani (ORISOLH)

Az előre definiált ORISOLH funkció támogatja a felhasználót abban, hogy egy gép körtengely pozícióit úgy beállítsa, hogy egy esztergaszerszám egy definiált, kinematika független helyzetbe legyen hozva a munkadarabhoz viszonyítva. Előfeltétel, hogy egy 6 tengelyes transzformáció aktív, amely kinematikai láncokkal van paraméterezve.

Két alapfunkció áll rendelkezésre.

- szerszámot beállítani
  A β és γ szögek előre meg lesznek adva. A funkció kiszámítja a három tájolási tengely ehhez szükséges szögeit.
- szerszám közvetlen beállítása
  A második és harmadik tájolási tengelyek szögei előre meg lesznek adva. A funkció kiszámítja ebből a hozzá tartozó β és γ szöget, továbbá a még hiányzó első tájolási tengely szögét.

Megjegyzés

tájolási tengelyek sorrendje

Ha végig futunk a kinematikai láncon, amelyik leírja a gép felépítését, a munkadarabtól a szerszámig, akkor egy 6-tengelyes transzformáció három tájolási tengelyének a sorrendjére a következő előírások érvényesek:

- A tájolási tengely, amelyik a **munkadarabhoz** legközelebb van, az **első** tájolási tengely.
- A tájolási tengely, amelyik a **szerszámhoz** legközelebb van, a **harmadik** tájolási tengely.

Az első tájolási tengely általában egy orsó, és a megfelelő forgatás ezért ezekben az esetekben egy forgatható frame-mel lesz megvalósítva.

Szintaxis

```
<RetVal> = ORISOLH(<Cntrl>,<W1>,<W2>)
```

Jelentés

ORISOLH:	funkció felhívás		
<RetVal>:	funkció visszaadási érték		
	adattípus:	INT	
	értéktartomány:	0, -2, -3, ..., -17	
	értékek:	0	A funkció hiba nélkül végre lett befejezve.
		-2	Nincs érvényes aktív transzformáció (6-tengelyes tájolási transzformáció).
		-3	Az első paraméter (<Cntrl>) negatív.
		-4	Az első paraméter (<Cntrl>) 1-es helye érvénytelen. Csak a 0 és 1 értékek megengedettek.
		-5	Az első paraméter (<Cntrl>) 10-es helye érvénytelen. Csak a 0 ... 3 értékek megengedettek.
		-6	Az első paraméter (<Cntrl>) 100-as helye érvénytelen. Csak a 0 és 1 értékek megengedettek.
		-7	Az első paraméter (<Cntrl>) 1000-es helye érvénytelen. Csak a 0 ... 3 értékek megengedettek.
		-8	A "Szerszám közvetlen beállítása" funkciónál a γ szög túl nagy.
		-9	A "Szerszám közvetlen beállítása" funkciónál a megadott tengely-pozíciók legalább egyike megsérti a tengely határt.
		-10	Nincs aktív szerszám.
		-11	A kívánt tájolás nem állítható be.
		-12	A szabad tengelyszög illesztése Hirth fogazásnál az első vagy egyetlen megoldásra nem lehetséges.
		-13	A szabad tengelyszög illesztése Hirth fogazásnál a második megoldásra nem lehetséges.
		-14	A szabad tengelyszög illesztése Hirth fogazásnál a két megoldás egyikére sem lehetséges.
		-15	Az első tájolási tengely Hirth tengelyként van paraméterezve.
		-16	Az első és a harmadik körtengely is Hirth tengelyként van paraméterezve. A két tengelyből max. csak egyik lehet Hirth tengely.
		-17	A "Közvetlen billentés" funkciónál a megadott tengely-pozíciók legalább egyike nem kompatibilis a hozzá tartozó Hirth fogazással.

<Cntrl>:	vezérli a funkció viselkedését		
	adattípus:	INT	
	A <Cntrl> paraméter decimálisan kódolt (1-estől 1000-es helyig):		
	1-es hely:	Az 1-es hely vezérli a viselkedést hibáknál.	
		xxx**0**	Hiba esetén (visszaadási érték < 0) a program végrehajtása megszakad és a 14106 vészjelzés lesz kiadva. **Utalás:** A vészjelzés az 1-es hely értékétől függetlenül akkor is ki lesz adva, a <Cntrl> paraméter negatív.
		xxx**1**	Hiba esetén (visszaadási érték < 0) **nem** lesz vészjelzés kiadva. A felhasználónak magának lehetősége van a programban megfelelően reagálni.
	10-es hely:	Vezérli a viselkedést egy Hirth fogazású tájolási tengely megléte esetén. **Utalás:** Ez a paraméter csak a "Szerszám beállítása" funkciónál lesz kiértékelve (azaz, ha a 100-as hely értéke "0").	
		xx**0**x	A tengelypozíció a következő pozícióra lett kerekítve.
		xx**1**x	A tengelypozíciók úgy lesznek kerekítve, hogy a β szög eltérése a programozott értékétől minimális legyen.
		xx**2**x	A tengelypozíciók úgy lesznek kerekítve, hogy a β szög a legnagyobb lehetséges értékkel egyenlő, ami kisebb a programozott értéknél (β lekerekítve lesz).
		xx**3**x	A tengelypozíciók úgy lesznek kerekítve, hogy a β szög a legkisebb lehetséges értékkel egyenlő, ami nagyobb a programozott értéknél (β felkerekítve lesz).
	100-as hely:	Megadja, hogy milyen funkciót kell végrehajtani, ill. a következő <W1> és <W2> paramétereknek mi a jelentése.	
		x**0**xx	"Szerszám beállítása" funkció <W1> és <W2> paraméterek jelentése a következő: • <W1> = β • <W2> = γ A tájolási tengelyek hozzá tartozó szögei ki lesznek számítva.
		x**1**xx	"Szerszám közvetlen beállítása" funkció <W1> a pozíció megadás a második tájolási tengelyre, <W2> a pozíció megadás a harmadik tájolási tengelyre egy 6-tengelyes transzformációnál. Az első tájolási tengely pozíciója és a β és γ szögek meghatározásra kerülnek, amelyek mindkét pozíció megadással kompatibilisek. Ha nem lép fel hiba, a $P_ORI_POS[<n>, <m>] rendszerváltozók mindig két megoldást adnak ki. Ennél az első <n> index (0 vagy 1) a megoldásra és a második <m> index (0 ... 2) a tájolási tengelyre mutat: • $P_ORI_POS[0/1, **0**]: első tájolási tengely pozíciója • $P_ORI_POS[0/1, **1**]: β szög • $P_ORI_POS[0/1, **2**]: γ szög Meg lesz vizsgálva, hogy a <W1> és <W2> pozíció adatok az esetleg aktív Hirth fogazással vagy az aktív szoftver ha-

			tárokkal kompatibilisek-e. Ha ez nem így van, egy megfelelő hibaszám lesz visszaadva (lásd <RetVal> paraméter). A <W1> és <W2> szögek tetszőleges választásánál a szerszám vágóéle általában nincs a megmunkálási síkban. A γ szög, amivel a vágóél a megmunkálási síkból ki van forgatva, nem lehet nagyobb a határértéknél, amelyet a SD42999 $SC_ORISOLH_INCLINE_TOL beállítási adat határoz meg.
	1000-es hely:	Megadja, hogy a megoldások pozíciói adott esetben hogyan lesznek módosítva, ha a 100-as hely értéke "0", azaz a "Szerszám beállítása" funkciónál.	
		0xxx	A kiszámított tengelypozíciók lehetőleg közel legyenek az aktuális géptengely pozíciókhoz.
		1xxx	A modulo-tengelyeknél a kiszámított tengelypozíciók lehetőleg a modulo-tartomány közepéhez közel, más tengelyeknél lehetőleg 0-hoz közel legyenek. A nem modulo tengelyekre ez azt jelenti, hogy a tengelypozíciók a -180° … +180° tartományra lesznek korlátozva.
		2xxx	A kiszámított tengelypozíciók a tengely-típustól függetlenül a -180° … +180° tartományra legyenek korlátozva.
<W1>:	első szög A jelentés a <Cntrl> paraméter 100-as helyéből adódik.		
	adattípus:	REAL	
<W1>:	második szög A jelentés a <Cntrl> paraméter 100-as helyéből adódik.		
	adattípus:	REAL	

Megjegyzés

Nem programozott paraméterek alapértéke "0".

További információk

A megtalált megoldások számát a további állapot információkkal együtt az ORISOLH funkció végrehajtásánál a következő rendszerváltozókkal lesz olvasva.

Rendszerváltozó	Jelentés		
$P_ORI_POS [<n>, <m>]	Megadja a tájolási tengelyek szögeit, amelyek a tájolás programozásnál adódnak.		
	<n>:	megoldás indexe	
		értéktartomány:	0, 1
	<m>:	tájolási tengely indexe	
		értéktartomány:	0 ... 2
			A tájolási tengelyek (1 ... 3) sorrendje a tengelyek definíciójára vonatkozik az $NT_ROT_AX_NAME-ben.
	Az ORISOLH funkció felhívásánál a "Szerszám közvetlen beállítása" módusban a a $P_ORI_POS[0/1, 1] és P_ORI_POS[0/1, 2] változók a β és γ szögeknek a mindkét megoldáshoz tartozó értékeit tartalmazzák. A $P_ORI_POS[<n>, <m>]-be bevitt első megoldás, azaz az <n> = 0 indexű, mindig az, amelyiket a vezérlés választ, amikor a kért tájolás közvetlenül fel lesz véve. A második <m> index a tájolási tengelyre vonatkozik, azaz az $NT_ROT_AX_NAME-re. Az $P_ORI_POS[<n>, <m>]-be bevitt tengelypozíciók figyelembe veszik az offseteket, amelyek az $NK_OFF-ba és az $NK_OFF_FINE-ba vannak beadva, azaz ezeket a tengelyszögeket a következő mondatokban módosítás nélkül lehet használni a kívánt tájolás beállításához. Ha a körtengely egy Hirth tengely, akkor a megoldás pozíciók a Hirth fogazás legközelebbi raszter helyére lesznek kerekítve. A $P_ORI_DIFF rendszerváltozókban ki lehet olvasni a Hirth fogazású körtengelyekre a pontos megoldások és a Hirth raszterre illesztett megoldások tengely-pozícióinak eltérését.		
$P_ORI_DIFF [<n>, <m>]	Megadja az eltérést a tájolási tengelyeknek a pontos és a $P_ORI_POS-ban rendelkezésre bocsájtott pozíciói között, amelyek a tájolás programozásnál adódnak.		
	<n>:	megoldás indexe	
		értéktartomány:	0, 1
	<m>:	tájolási tengely indexe	
		értéktartomány:	0 ... 2
			A tájolási tengelyek (1 ... 3) sorrendje a tengelyek definíciójára vonatkozik az $NT_ROT_AX_NAME-ben.
	A tartalom csak akkor lehet nullától eltérő, ha a pozíciók raszterben vannak (Hirth fogazás), azaz ha az érintett tengely $NT_HIRTH_INCR rendszeradata nem nulla és ha ez a tengely egy kézi körtengely.		

Rendszerváltozó	Jelentés		
$P_ORI_SOL	Ha egy tájolási transzformáció egynél több tájolási tengelyre lesz a tengelyszög kiszámítva, amelyek adott tájoláshoz vezetnek, általában nincs megoldás. A $P_ORI_SOL rendszerváltozókban az érvényes megoldások száma további állapot információkkal együtt szerepel. A $P_ORI_SOL tartalma következőképpen van kódolva:		
	érték < 0	Általános hiba állapotok	
		-1	Az aktív transzformációra még nem lett kiszámítva megoldás (hiányzó ORISOLH hívás).
		-2	Nincs aktív transzformáció, vagy az aktív transzformáció nem tájolási transzformáció (6-tengelyes transzformáció), amelyik pozíciókat tud adni az megadott tájolás programozáshoz.
		-4	A kívánt tájolás az adott kinematikával nem állítható be.
		-5	A ORISOLH funkció felhívásánál "Szerszám közvetlen beállítása" módusban nincs megoldás.
		-6	A ORISOLH funkció felhívásánál "Szerszám közvetlen beállítása" módusban a γ szög túl nagy.
		-7	A ORISOLH funkció felhívásánál "Szerszám közvetlen beállítása" egy olyan szög lett megadva, amelyik a Hirth fogazás alapján nem állítható be.
		-8	Az első tájolási tengelyt (frame tengely) nem lehet Hirth tengelyként paraméterezve.
		-9	Az első és a harmadik körtengely is Hirth tengelyként van paraméterezve. A két tengelyből max. csak egyik lehet Hirth tengely.
		-10	Nincs lehetőség a megoldás(ok) illesztésére a Hirth tengely fogazáshoz.
	érték > 0 1-es hely	A matematikailag lehetséges megoldások száma a tengely határok és az esetleges hiba feltételek figyelembe vétele nélkül.	
		0	Nincs megoldás, azaz a kívánt tájolás nem állítható be. Ennek az esetnek három különböző oka lehet: • A kívánt tájolás a gép kinematikája alapján (nem derékszögűen elrendezett tájolási szögek) elvileg tetszőleges tájolási tengely mozgástartományoknál sem érhető el. Ebben az esetben a $P_ORI_SOL 10-es helye és 100-as helye is nulla, a tájolási tengelyekhez hozzárendelt $P_ORI_STAT állapot változó a "-4" értéket tartalmazza. • A kiszámított megoldások nem elérhetők, mert megsértenék a tengely határolásokat. A tájolási tengelyek pozíciói, amelyek a tengely határolások nélkül adódnának, az $P_ORI_POS-ban nem olvashatóak. • A ORISOLH funkció felhívásánál "Szerszám közvetlen beállítása" a tengely pozíciók úgy lettek megadva, hogy a szerszám tájolási vektora vagy tájolási normál vektora párhuzamosan van beállítva az első tájolási tengellyel, amelynek a pozícióját ki kell számítani. Ezen tengely pozíciója ezekben az esetekben nem definiált.
		1	Egy megoldás van.

Rendszerváltozó	Jelentés		
			Ennek az esetnek három különböző oka lehet: • Az előre megadott tájolás és a gép kinematikája alapján a tengely-határok figyelembe vétele nélkül is csak egy megoldás van (matematikailag két egybeeső megoldás van). Ez az eset a nem derékszögű kinematikáknál a tájolási tartomány szélén lép fel. A $P_ORI_POS mindkét (azonos) megoldást tartalmazza. • Csak egy megoldás van, mert egy második megoldás a tengely-határok megsértése miatt érvénytelen. Az érvényes megoldás mindig az első megoldás a $P_ORI_POS-ban. A második megoldás, amelyik a tengely-határok figyelembe vétele nélkül adódna, az $P_ORI_POS-ban szintén olvasható. • A ORISOLH funkció felhívásánál "Szerszám közvetlen beállítása" módusban ez a normál eset. Két tájolási tengelynek a megadott tengely-pozícióihoz a kiszámítandó hiányzó tájolási tengelyre csak egy érvényes pozíció van.
		2	Két megoldás van.
		8	Végtelenül sok megoldás van, azaz egy tájolási tengely (pólus tengely) pozíciója tetszőleges. A másik tengely(ek) lehetséges két állásából az egyik a tengely-határok megsértése miatt kizárva.
		9	Végtelenül sok megoldás van, azaz egy tájolási tengely (pólus tengely) pozíciója határozatlan. A nem meghatározott tengely a 100-as helyből vagy a a $P_ORI_STAT rendszerváltozókból lehet megállapítani.
	érték > 0 10-es hely	Bit kódolású kijelzés megsértett tengely-határokra. A pontos hibaokot a $P_ORI_STAT rendszerváltozókból lehet megállapítani.	
		bit 0 (érték 10):	Legalább egy megoldásra az 1. tájolási tengely legalább egy tengely-határa meg van sértve.
		bit 1 (érték 20):	Legalább egy megoldásra az 2. tájolási tengely legalább egy tengely-határa meg van sértve.
		bit 2 (érték 40):	Legalább egy megoldásra az 3. tájolási tengely legalább egy tengely-határa meg van sértve.
	érték > 0 100-as hely	Bit kódolású kijelzés nem definiált tengely-pozíciókra (csak akkor tud fellépni, ha végtelen sok megoldás van, azaz ha az 1-es hely egyenlő "9"-cel).	
		bit 0 (érték 100):	Az 1. tájolási tengely pozíciója nem definiált.
		bit 1 (érték 200):	Az 2. tájolási tengely pozíciója nem definiált.
		bit 2 (érték 400):	Az 3. tájolási tengely pozíciója nem definiált.
	Az 1., 2. és 3. jelölések A tájolási tengelyek a tengelyek definíciójára vonatkoznak az $NT_ROT_AX_NAME-ben.		

Rendszerváltozó	Jelentés			
$P_ORI_STAT [<n>]	A maximum 3 tájolási tengelyre az állapotot adja az ORISOLH felhívása után.			
	<n>:	tájolási tengely indexe (megfelel az érintett tájolási tengely indexének a $NT_ROT_AX_NAME-ben)		
		értéktartomány:	0 ... 2	
			A tájolási tengelyek (1 ... 3) sorrendje a tengelyek definíciójára vonatkozik az $NT_ROT_AX_NAME-ben.	
	A $P_ORI_STAT tartalma következőképpen van kódolva:			
	érték < 0	Általános hiba állapotok		
		-1	Az állapot nem definiált (hiányzó ORISOLH hívás).	
		-2	Nincs aktív transzformáció, vagy az aktív transzformáció nem tájolási transzformáció (6-tengelyes transzformáció), amelyik pozíciókat tud adni az megadott tájolás programozáshoz.	
		-3	A tengely nem része az aktív transzformációnak.	
		-4	A tengely pozícióját nem lehet kiszámítani, mert az adott kinematikával a kívánt tájolás a tengely tetszőleges feltételezett mozgástartományban sem érhető el.	
		-5	A ORISOLH funkció felhívásánál "Szerszám közvetlen beállítása" a tengely pozíciók úgy lettek megadva, hogy a szerszám tájolási vektora vagy tájolási normál vektora párhuzamosan van beállítva az első tájolási tengellyel, amelynek a pozícióját ki kell számítani. Ezen tengely pozíciója ezekben az esetekben nem definiált.	
		-6	A ORISOLH funkció felhívásánál "Szerszám közvetlen beállítása" módusban a γ szög túl nagy.	
		-7	A ORISOLH funkció felhívásánál "Szerszám közvetlen beállítása" egy olyan szög lett megadva, amelyik a Hirth fogazás alapján nem állítható be.	
		-8	Az első tájolási tengelyt (frame tengely) nem lehet Hirth tengelyként paraméterezve.	
		-9	Az első és a harmadik körtengely is Hirth tengelyként van paraméterezve. A két tengelyből max. csak egyik lehet Hirth tengely.	
		-10	Nincs lehetőség a megoldás(ok) illesztésére a Hirth tengely fogazáshoz.	
	érték > 0 1-es hely	Bit kódolású kijelzés megsértett tengely-határokra az első megoldásnál.		
		bit 0 (érték 1):	Az első megoldás megsérti az alsó tengelyhatárt.	
		bit 1 (érték 2):	Az első megoldás megsérti a felső tengelyhatárt.	
	érték > 0 10-es hely	Bit kódolású kijelzés megsértett tengely-határokra a második megoldásnál.		
		bit 0 (érték 10):	A második megoldás megsérti az alsó tengelyhatárt.	
		bit 1 (érték 20):	A második megoldás megsérti a felső tengelyhatárt.	
	érték > 0 100-as hely	Egy nem definiált tengelypozíció kijelzése.		
		bit 0 (érték 100):	A tájolási tengely pozíciója nem definiált, azaz a kívánt tájolás a körtengely összes tetszőleges	

Rendszerváltozó	Jelentés		
			beállításával el lesz érve (pólus helyzet). Ez az információ a $P_ORI_SOL rendszerváltozókban is megvan.
	A tengely-határok megsértését jelző hiba-számokból egyidőben több is felléphet. Egy tengely-határ megsértésénél meg lesz kísérelve a 360° többszörösének hozzáadásával vagy levonásával egy, a megengedett tengely-határokon belüli pozíció elérése, ha ez nem lehetséges, nincs egyértelműen definiálva, hogy az alsó vagy a felső tengely-határ lett megsértve. Ha a kívánt tájolásra nincs megoldás ($P_ORI_SOL = 0), a transzformációban levő tájolótengelyek állapota "0".		

Megjegyzés

$NT_ROT_AX_NAME

Ezekkel a rendszerváltozókkal maximum 3 tengelyre lesz utalva, amelyek a tájolás beállítását szolgálják. Tartalmazza a lánc-elemek neveit ($NK_NAME), amelyek azokat a géptengelyeket (körtengelyek) definiálják, amelyek a kinematikus transzformációkból adódó tájolási mozgásokat végre kell hajtsák. A sorrend, amiben a maximum három körtengely ebben a rendszerváltozóban szerepel, a gép kinematikája szempontjából jelentés nélküli, mivel ez a kinematikai lánc szerkezetéből lesz levezetve. De mivel a sorrend definiálja, amivel a más változók a kör-tengelyekhez hozzáférnek, a tájolási tengelyek sorrendjének a $NT_ROT_AX_NAME-ben egyezni kell a kinematika leírással.

Megjegyzés

Állapot információk

Az állapot információk, amelyek pl. kijelzik, hogy egy tájolás nem elérhető vagy csak a releváns tengely-határok megsértésével lenne elérhető, nem okoznak NC vészjelzést. A felhasználó felelőssége a megnevezett feltételekre megfelelően reagálni.

11.11.2 Korrekcióadatok módosítását forgatható szerszámoknál aktiválni (CUTMOD, CUTMODK)

Korrekcióadatok módosítása forgatható szerszámoknál az NC programban a CUTMOD (tájolható szerszámtartókkal kapcsolatban) ill. CUTMODK (tájolás transzformációkra, amelyek kinematikai láncként lettek definiálva) nyelvi utasítással kell aktiválni.

Megjegyzés

Mivel a tájolható szerszámtartó és a tájolási transzformációk, amelyek kinematikai láncokkal lettek definiálva, nem lehetnek egyidőben aktívak, nincs konfliktus a két változat között.

Szintaxis

```
CUTMOD = <Value>
```

ill.:

```
CUTMODK = <Command>
```

Jelentés

CUTMOD:	funkció hívás tájolható szerszámtartókkal kapcsolatban		
<Value>:	hozzárendelt érték		
	adattípus:	INT	
	érték:	0	A funkció nem aktív. A $P_AD... rendszerváltozók által adott értékek azonosak a megfelelő szerszám-paraméterekkel.
		> 0	A funkció aktiválva lesz, ha egy tájolható szerszámtartó a megadott számmal aktív, vagyis az aktiválás egy megadott tájolható szerszámtartóhoz kötött. A $P_AD... rendszerváltozók által adott értékek a megfelelő szerszám-paraméterekhez képest az aktív forgatástól függően módosítva lettek. A megadott tájolható szerszámtartó deaktiválása a funkciót átmenetileg deaktiválja, egy másik tájolható szerszámtartó aktiválása viszont tartósan deaktiválja. Az első esetben ezért a funkciót ugyanannak a tájolható szerszámtartónak az újbóli kiválasztása ismét aktiválja, a második esetben egy újbóli kiválasztás szükséges akkor is, ha egy későbbi időpontban a tájolható szerszámtartó a megadott számmal újra aktiválva lesz. A funkciót a Reset nem befolyásolja.
		-1	A funkció mindig aktiválva lesz, ha egy tájolható szerszámtartó aktív. A szerszámtartó cseréjénél vagy kikapcsolásánál és későbbi újra kiválasztásánál a CUTMOD-t nem kell újra beállítani.
		-2	A funkció mindig aktiválva lesz, he egy tájolható szerszámtartó aktív, amelynek száma azonos az aktuálisan aktív tájolható szerszámtartóval. Ha nincs aktív tájolható szerszámtartó, az azonos jelentésű, mint a CUTMOD=0. Ha egy tájolható szerszámtartó aktív, az azonos jelentésű az aktuális szerszámtartó-szám közvetlen megadásával..
		< -2	A 2-nél kisebb értékek nem lesznek figyelembe véve, vagyis ez az eset úgy van kezelve, mintha a CUTMOD nem lenne programozva. **Utalás:** Ez az értéktartomány lehetőleg ne legyen megváltoztatva, mert ez későbbi bővítésekre van tartalékolva..
CUTMODK:	funkció hívás tájolás transzformációkkal kapcsolatban, amelyek kinematikai láncokkal lettek definiálva		

<Command>:	hozzárendelt utasítás		
	adattípus:	STRING	
	érték:	"NEW"	Egy aktív, kinematikai láncokkal definiált transzformációnak "Korrekció adatok módosítása" szempontjából releváns adatai, a transzformáció és az aktuális kontúr frame neve tárolva lesz. **Utalás:** Ez az utasítás csak akkor megengedett, ha a megfelelő transzformáció (TRAORI_DYN, TRAORI_STAT vagy TRAANG_K) aktív.
		"OFF"	Kikapcsolja a "Korrekció adatok módosítása"-t. Az előzőleg "NEW"-val létrehozott adatok megmaradnak. **Utalás:** Ez az utasítás akkor is megengedett, ha a CUTMODK nem aktív. Ez ekkor hatástalan marad. Az esetleg meglevő adatkészlet a "Korrekció adatok módosítására" megmarad.
		"ON"	Ezzel az utasítással a "Korrekció adatok módosítása" egy előzőleg a "NEW"-val eltárolt adatkészlettel újra aktiválva lesz. Ha ezen utasítás végrehajtásánál egy transzformáció a tárolt adatkészlet nevével aktív, a "Korrekció adatok módosítása" azonnal hatásos lesz. Egyébként az aktiválás késleltetve lesz egy aktív transzformáció aktiválásáig.
		"CLEAR"	Mint az "OFF" utasítás, kikapcsolja a "Korrekció adatok módosítása"-t és azon kívül törli a tárolt adatkészletet. **Utalás:** Ez az utasítás akkor is megengedett, ha a CUTMODK nem aktív.

Megjegyzés

SD42984 $SC_CUTDIRMOD

A CUTMOD ill. CUTMODK utasítással aktiválható funkció helyettesíti az SD42984 $SC_CUTDIRMOD beállítási adattal aktiválható funkciót. Ez a funkció azonban továbbra is rendelkezésre áll. De mivel értelmetlen a két funkciót párhuzamosan használni, csak akkor lehet aktiválni, ha a CUTMOD nullával és a CUTMODK nulla karakterlánccal egyenlő.

További információk

Módosított korrekció adatokat olvasni

A módosított korrekció adatok a következő rendszerváltozókban és BTSS változókban lesznek rendelkezésre bocsájtva:

Jelentés	Rendszerváltozó	BTSS változó
vágóélhelyzet	$P_AD[2]	cuttEdgeParam2
tartószög	$P_AD[10]	cuttEdgeParam10
vágásirány	$P_AD[11]	cuttEdgeParam11
szabadszög	$P_AD[24]	cuttEdgeParam24

Az adatok a megfelelő szerszámparaméterekhez ($TC_DP2[..., ...] stb.) viszonyítva mindig akkor vannak módosítva, ha a "Korrekció adatok módosítása forgatható szerszámoknál" funkció a `CUTMOD`-dal ill. `CUTMODK`-val aktiválva lett és a szerszám egy tájolható szerszámtartó ill. egy megfelelő tájolási transzformáció által forgatva lett.

További, a funkció szempontjából releváns rendszerváltozók

Rendszerváltozó	Jelentés
$P_CUTMOD_ANG / $AC_CUTMOD_ANG	A szöget adja, amivel egy szerszám az aktív megmunkálási síkban el lett forgatva és amely a CUTMOD ill. CUTMODK funkcióknál a módosított vágóél adatok megállapításának alapjául szolgál.
$P_CUTMOD / $AC_CUTMOD	Olvassa az érvényes értéket, ami utoljára a CUTMOD utasítással programozva lett (szerszámtartó száma, amelyre a korrekció adatok módosítása aktiválva kell legyen). Ha az utoljára programozott `CUTMOD=-2` értékű volt (aktiválás az aktuálisan aktív tájolható szerszámtartóval), akkor a rendszerváltozóban nem a "-2" érték, hanem a programozás időpontjában aktív tájolható szerszámtartó száma lesz visszaadva.
$P_CUTMODK / $AC_CUTMODK	Olvassa a transzformáció nevét, amely alatt a "Korrekció adatok módosításához" az aktuális érvényes adatkészlet létre lett hozva.
$P_CUT_INV / $AC_CUT_INV	A TRUE értéket adja, ha a szerszám úgy van elforgatva, hogy az orsóirányt invertálni kell. Ehhez abban a mondatban, amelyikre az olvasási művelet vonatkozik, a következő négy feltételnek kell teljesülnie: 1. Egy eszterga- vagy köszörűszerszám aktív (szerszámtípus 400 ... 599 és / vagy SD42950 $SC_TOOL_LENGTH_TYPE = 2). 2. A korrekció adatok módosítása a `CUTMOD` ill. `CUTMODK` utasítással lesz aktiválva. 3. Egy tájolható szerszámtartó ill. egy kinematikai láncokkal definiált tájolási transzformáció aktív, amelyik a CUTMOD ill. CUTMODK utasítással lett aktiválva. 4. A szerszám a tájolható szerszámtartó által ill. a kinematikai tájolási transzformáció által úgy lesz elforgatva, hogy a szerszám-vágóél eredő normálja a kiindulási helyzethez képest 90°-nál többel (tipikusan 180°) van elforgatva. Ha a megadott négy feltételből legalább egy nem teljesül, a változó a FALSE értéket adja. Azokra a szerszámokra, amelyek vágóélhelyzete nem definiált, a változó értéke mindig FALSE.

Rendszerváltozó	Jelentés	
$P_CUTMOD_ERR	hiba állapot a CUTMOD funkció utolsó felhívása után A CUTMOD funkciót fel lehet hívni közvetve is a szerszámcserénél. A változó Reset-nél nullára lesz visszaállítva. Ez a szerszámcserénél először törölve lesz és adott esetben újra írva. A változó bit kódolású. A bitek jelentése a következő:	
	bit 0:	Az aktív szerszámra nincs érvényes vágásirány definiálva.
	bit 1:	Az aktív szerszám mindkét vágóél-szöge (szabadszög és tartószög) nulla.
	bit 2:	Az aktív szerszám szabad-szöge nem megengedett értékű (< 0° vagy > 180°).
	bit 3:	Az aktív szerszám tartó-szöge nem megengedett értékű (< 0° vagy > 90°).
	bit 4:	Az aktív szerszám lapkaszöge nem megengedett értékű (< 0° vagy > 90°).
	bit 5:	Az aktív szerszám vágóélhelyzet - tartószög kombinációja nem megengedett (az 1 ... 4 vágóélhelyzeteknél a tartószög ≤ 90° kell legyen, az 5 ... 8 vágóélhelyzeteknél ≥ 90° kell legyen).
	bit 6:	Aktív szerszám nem megengedett forgatása. A szerszám ± 90°-kal (kb. 1° tűréssel) ki lett forgatva az aktív megmunkálási síkból. Ezzel a vágóélhelyzet a megmunkálási síkban többé nem definiált.
	bit 7:	A vágólapka nincs a megmunkálási síkban, és a vágólapka és megmunkálási sík közötti szög túllépi a SD42998 $SC_CUTMOD_PLANE_TOL rendszeradatban megadott felső határt.
	bit 8:	A vágólapka nincs a megmunkálási síkban. Az α szög értéke nagyobb 1°-nál. Az α szög a forgásszög a koordináta-tengely körül, amely a β szög forgástengelyére és a γ szög forgástengelyére is merőleges (G18-nál az X tengely).

$P_...: előrefutás-változók

$AC_...: főfutás-változók

Az összes főfutás változót lehet olvasni szinkronakciókban. Egy olvasás az előrefutásból előrefutás-álljt okoz.

Sík váltás

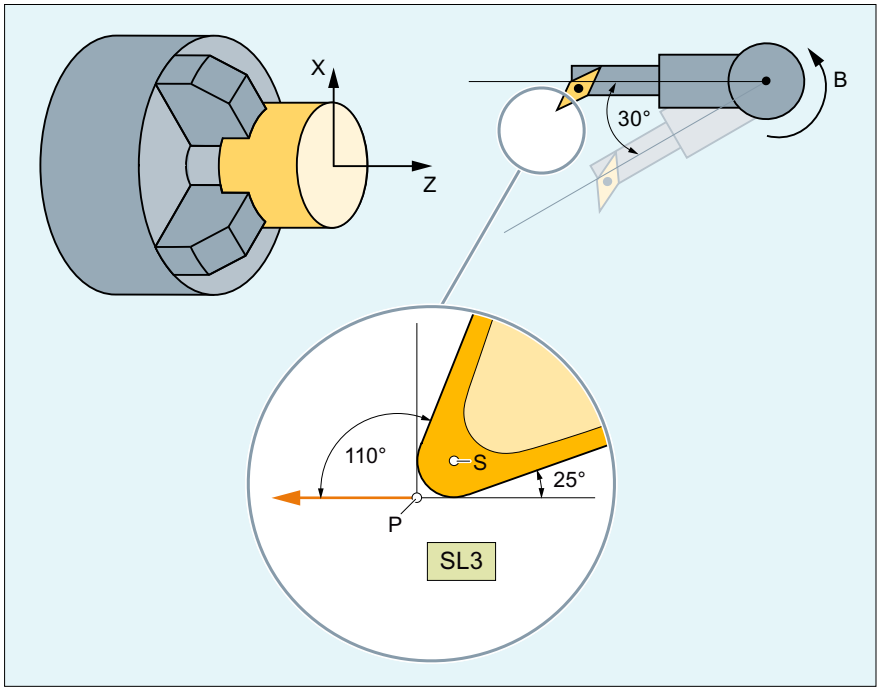
A módosított vágóélhelyzet, vágásirány és tartó- ill. szabadszög meghatározására a a vágóél nézete a mindenkori aktív síkban (G17 - G19) mérvadó.

Ha azonban a SD42940 $SC_TOOL_LENGTH_CONST beállítási adat (szerszámhossz-kompenzáció váltása síkváltásnál) egy nullától különböző érvényes értéket (plusz vagy mínusz 17, 18 vagy 19) tartalmaz, akkor ez határozza meg a síkot, amelyben a megfelelő értékeket meg kell nézni.

A beállítási-adatok elsőbbségi sorrendje a G kódok által az $MC_TOOL_PARAMETER_DEF_MASK beállítási-adat bit 18 beállításával hatályon kívül helyezhető. Azaz, ha ez a bit be van állítva, mindig a 6-os G csoport G utasításával definiált sík érvényes.

Módosított vágóélhelyzet hatásossága

A módosított vágóélhelyzet és a módosított vágóél vonatkoztatási pont a programozásnál a már aktív szerszámra azonnal hatásos. A szerszám kiválasztása ehhez nem szükséges.



S: vágóél középpont

P: vágóél vonatkoztatási pont

SL: vágóélhelyzet

Kép 11-4 Szerszámra vágóélhelyzet 3-mal és egy tájolható szerszámtartóval, amelyik a szerszámot a B tengely körül forgatni tudja.

Programkód	Kommentár
N10 $TC_DP1[1,1]=500	
N20 $TC_DP2[1,1]=3	; vágóélhelyzet
N30 $TC_DP3[1,1]=12	
N40 $TC_DP4[1,1]=1	
N50 $TC_DP6[1,1]=6	
N60 $TC_DP10[1,1]=110	; tartószög
N70 $TC_DP11[1,1]=3	; vágásirány
N80 $TC_DP24[1,1]=25	; szabadszög
N90 $TC_CARR7[2]=0 $TC_CARR8[2]=1 $TC_CARR9[2]=0	; B tengely
N100 $TC_CARR10[2]=0 $TC_CARR11[2]=0 $TC_CARR12[2]=1	; C tengely
N110 $TC_CARR13[2]=0	
N120 $TC_CARR14[2]=0	
N130 $TC_CARR21[2]=X	
N140 $TC_CARR22[2]=X	

```
Programkód                                    Kommentár
N150 $TC_CARR23[2]="M"

N160 TCOABS CUTMOD=0
N170 G18 T1 D1 TCARR=2                        ; X        Y         Z
N180 X0 Y0 Z0 F10000                          ; 12.000   0.000     1.000

N190 $TC_CARR13[2]=30
N200 TCARR=2
N210 X0 Y0 Z0                                 ; 10.892   0.000    -5.134
N220 G42 Z-10                                 ;  8.696   0.000   -17.330
N230 Z-20                                     ;  8.696   0.000   -21.330
N240 X10                                      ; 12.696   0.000   -21.330
N250 G40 X20 Z0                               ; 30.892   0.000    -5.134

N260 CUTMOD=2 X0 Y0 Z0                        ;  8.696   0.000    -7.330
N270 G42 Z-10                                 ;  8.696   0.000   -17.330
N280 Z-20                                     ;  8.696   0.000   -21.330
N290 X10                                      ; 12.696   0.000   -21.330
N300 G40 X20 Z0                               ; 28.696   0.000    -7.330

N310 M30
```

A számértékek a kommentárokban megadják a mindenkori mondatvégi pozíciókat gép-koordinátákban (GKR) az X → Y → Z sorrendben.

Magyarázat

Az `N180` mondatban a szerszám először `CUTMOD=0`-nál és nem elforgatott tájolható szerszámtartóval lesz kiválasztva. Mivel a tájolható szerszámtartó valamennyi offset-vektora 0, az a pozíció lesz felvéve, amelyik a `$TC_DP3[1,1]` és `$TC_DP4[1,1]`-ben megadott szerszámhossznak megfelel.

Az `N200` mondatban a tájolható szerszámtartó egy 30°-os forgatással a B tengely körül lesz aktiválva. Mivel a vágóélhelyzet `CUTMOD=0` miatt nem lesz módosítva, továbbra is a régi vágóél vonatkoztatási pont a mérvadó. Ezért az `N210` mondatban az a pozíció lesz felvéve, amelyik a régi vágóél vonatkoztatási pontot a nullapontban megtartja (vagyis az (1, 12) vektor a Z/X-síkban 30°-kal lesz elforgatva).

Az `N260` mondatban ellenben a `N200 CUTMOD=2` hatásos. A tájolható szerszámtartó forgatása miatt a módosított vágóélhelyzet 8. Ebből következnek az eltérő tengelypozíciók.

Az `N220` ill. `N270` mondatokban a szerszámsugár-korrekció (SZSK) aktiválva lesz.. Az eltérő vágóélhelyzetnek a két programrészben nincs befolyása a mondatok végpozícióira, amelyekben a szerszámsugár-korrekció aktív, a megfelelő pozíciók ezért azonosak. Csak az `N260` ill. `N300` kikapcsolási mondatokban hatnak ismét a különböző vágóélhelyzetek.

11.12 Szerszám környezetekkel dolgozni

Funkciók áttekintése

- Szerszám-környezetet tárolni (TOOLENV) (Oldal 471)
- Szerszám-környezetet törölni (DELTOOLENV) (Oldal 474)
- T-, D-, és DL-számokat olvasni (GETTENV) (Oldal 475)
- Szerszámhosszakat ill. szerszámhossz-komponenseket kiolvasni (GETTCOR) (Oldal 476)
- Szerszám-komponenseket változtatni (SETTCOR) (Oldal 483)

Rendszerváltozók áttekintése

- Információk olvasása a tárolt szerszám-környezethez ($P_TOOLENVN, ($P_TOOLENV) (Oldal 476)

11.12.1 Szerszám-környezetet tárolni (TOOLENV)

A TOOLENV funkció az összes aktuális állapot tárolását szolgálja, amelyeknek a tárolóban eltárolt szerszámadatok kiértékelésénél jelentésük van.

Ezek részletesen a következő adatok:

- G-csoport aktív G utasítása
  - 6 (G17, G18, G19)
  - 56 (TOWSTD, TOWMCS, TOWWCS, TOWBCS, TOWTCS, TOWKCS)
- Az aktív sík
- A gépadatok:
  - MD18112 $MN_MM_KIND_OF_SUMCORR (összeg-korrekciók tulajdonságai a TO-tartományban)
  - MD20360 $MC_TOOL_PARAMETER_DEF_MASK (szerszám paraméter definíció)

- A beállítási adatok:
  - SD42900 $SC_MIRROR_TOOL_LENGTH (előjelváltó szerszámhossz tükrözésnél)
  - SD42910 $SC_MIRROR_TOOL_WEAR (előjelváltó kopás tükrözésnél)
  - SD42920 $SC_WEAR_SIGN_CUTPOS (kopás előjele vágóél készülékes szerszámoknál)
  - SD42930 $SC_WEAR_SIGN (kopás előjele)
  - SD42935 $SC_WEAR_TRANSFORM (transzformációk szerszám-komponensekre)
  - SD42940 $SC_LENGTH_CONST (szerszámhossz-komponensek váltása síkváltásnál)
  - SD42942 $SC_TOOL_LENGTH_CONST_T (szerszámhossz-komponensek váltása síkváltásnál esztergaszerszámokra)
  - SD42950 $SC_TOOL_LENGTH_TYPE (szerszámhossz-összetevők hozzárendelése a szerszámtípustól függetlenül)
  - SD42954 $SC_TOOL_ORI_CONST_M (szerszámtájolás komponensek váltása síkváltásnál marószerszámokra)
  - SD42956 $SC_TOOL_ORI_CONST_M (szerszámtájolás komponensek váltása síkváltásnál esztergaszerszámokra)
- Az aktuális össz-frame (forgatások és tükrözések, nincsenek nullaponteltolások vagy skálázások) tájolási része
- Az aktív tájolható szerszámtartó tájolási része és eredő hossza
- Egy aktív transzformáció tájolási része és eredő hossza

A megnevezett adatokon kívül, amelyek a szerszám környezetét leírják, az aktív szerszám T-száma, D-száma és DL-száma is tárolva lesz, hogy később ezt szerszámot azonos környezetben, mint a TOOLENV felhívásánál, el lehessen érni, a szerszám újra leírása nélkül.

Szintaxis

```
<állapot> = TOOLENV(<név>)
```

Jelentés

TOOLENV(...):	Előre definiált funkció egy szerszám-környezet tárolására	
	egyedül a mondatban:	igen

<állapot>:	Funkció visszaadási érték. Negatív értékek hibaállapotot mutatnak.		
	adattípus:		INT
	érték:	0	Funkció OK
		-1	Nincs tárolóhely foglalva a szerszám-környezetek számára: MD18116 $MN_MM_NUM_TOOL_ENV = 0 Azaz a "Szerszám-környeztek" funkcionalitás nem létezik.
		-2	Nincs több tárolóhely a szerszám-környezetek számára.
		-3	A nulla-string nem megengedett a szerszám-környezet neveként.
		-4	Nincs (<név>) paraméter megadva.
Paraméter			
1 <név>:	név, amelyen az aktuális adatkészletet tárolni kell Ha már van egy azonos nevű adatkészlet, az át lesz írva. Az állapot ebben az esetben "0".		
	adattípus:		STRING

További információk

Alapméret/adapter-méret - szerszámhossz-korrekció

Azt, hogy a szerszámhossz kiszámításában az adapter-hossz vagy az alap-méret ($TC_DP21, $TC_DP22 und $TC_DP23 vágóél-specifikus paraméter) lesz figyelembe véve, aktív szerszámtár-kezelésnél (csak a "Szerszámkezelés" opcióval áll rendelkezésre) a következő gépadat értéke dönti el:

MD18104 $MN_MM_NUM_TOOL_ADAPTER (szerszám-adapter a TO-tartományban)

Mivel ennek a gépadatnak a változása csak a vezérlés felfutásánál válhat hatásossá, ez nem lesz tárolva a szerszám-környezetben.

Tájolható szerszámtartók és transzformációk eredő hossza

Megjegyzés

Úgy a tájolható szerszámtartóknál, mint a transzformációknál is vannak rendszerváltozók ill. gépadatok, amelyek további szerszámhossz-komponensenként hatnak, és teljesen vagy részben ezeknek a végrehajtható forgatásoknak lesznek alárendelve. A további szerszámhossz részeket, amelyek ezáltal adódnak, a TOOLENV felhívásánál szintén meg kell adni, mert ezek a környezet részét képezik, amelyben a szerszám használva lesz.

Adapter transzformáció

Az adapter-transzformáció a szerszámtartó és ezzel a teljes szerszám egy tulajdonsága. Ezért ez nem része a szerszám-környezetnek, amelyet egy másik szerszámra használni lehet.

A szerszám össz-hosszak meghatározáshoz szükséges összes adat eltárolása által lehetséges egy későbbi időpontban egy szerszám hatásos hosszát kiszámítani, akkor is, ha az abban az időpontban már nem aktív ill. ha a környezeti feltételek (pl. G utasítások vagy beállítási-adatok) megváltoztak. Egy másik szerszám hatásos hosszát szintén ki lehet számítani annak a feltételezésével, hogy ugyanolyan körülmények között lesz használva, mint az a szerszám, amelyikre az állapot tárolva lett.

Adatkészletek maximális száma szerszám-környezetek számára

A MD18116 $MN_MM_NUM_TOOL_ENV gépadatban van megadva, hogy a szerszám-környezetek leíráshoz maximum hány adatkészletet lehet létrehozni. Az adatok a TOA tartományban vannak. Ezek a vezérlés kikapcsolása után is megmaradnak.

Az adatmentés nem lehetséges. Ez azt jelenti, hogy ezek az adatok a különböző vezérlések között nem vihetők át.

11.12.2 Szerszám-környezetet törölni (DELTOOLENV)

A DELTOOLENV funkcióval a szerszám-környezeteket leíró adatkészleteket lehet törölni. A törlés azt jelenti, hogy az adott néven tárolt adatkészlethez többé nem lehet hozzáférni (a hozzáférési kísérlet vészjelzést okoz).

Megjegyzés

Az adatkészleteket csak a DELTOOLENV funkcióval, egy INITIAL.INI letöltéssel vagy egy hideg indítással (NC felfutás szabvány gépadatokkal) lehet törölni. Nincsenek további automatikus törlési eljárások.

Szintaxis

```
<állapot> = DELTOOLENV(<név>)
<állapot> = DELTOOLENV()
```

Jelentés

`DELTOOLENV(...)`:	Előre definiált funkció egy szerszám-környezet törlésére		
	egyedül a mondatban:		igen
`<állapot>`:	Funkció visszaadási érték. Negatív értékek hibaállapotot mutatnak.		
	adattípus:		INT
	érték:	0	Funkció OK
		-1	Nincs tárolóhely foglalva a szerszám-környezetek számára: MD18116 $MN_MM_NUM_TOOL_ENV = 0 Azaz a "Szerszám-környeztek" funkcionalitás nem létezik.
		-2	A szerszám-környezet a megadott névvel nem létezik.
Paraméter			
1 `<név>`:	törlendő adatkészlet neve		
	adattípus:		STRING
`DELTOOLENV()`:	`DELTOOLENV()` a név megadása nélkül törli a szerszám-környezeteket leíró összes adatkészletet		

11.12.3 T-, D-, és DL-számokat olvasni (GETTENV)

A GETTENV funkció egy szerszám környezetben tárolt T-, D- és DL-számok olvasására szolgál.

Szintaxis

```
<állapot>= GETTENV(<név>, <TDDL>)
```

Jelentés

GETTENV(...):	Előre definiált funkció a T-, D- és DL-számok olvasására egy szerszám-környezet leírását szolgáló adatkészletben		
	egyedül a mondatban:		igen
<állapot>:	Funkció visszaadási érték. Negatív értékek hibaállapotot mutatnak.		
	adattípus:		INT
	érték:	0	Funkció OK
		-1	Nincs tárolóhely foglalva a szerszám-környezetek számára: MD18116 $MN_MM_NUM_TOOL_ENV = 0 Azaz a "Szerszám-környeztek" funkcionalitás nem létezik.
		-2	A szerszám-környezet a megadott névvel nem létezik.
Paraméter			
1 <név>:	Adatkészlet neve, amelyből a T-, D-, és DL-számokat olvasni kell		
	adattípus:		STRING
2 <TDDL>:	Ezen eredmény-paraméter mező tartalmazza a szerszám T-, D- és DL-számát, amelynek a szerszám-környezete a megadott adatkészletben tárolva van: • <TDDL> [0]: T-szám • <TDDL> [1]: D-szám • <TDDL> [2]: DL-szám		
	adattípus:		INT[3]
GETTENV(,<TDDL>), GETTENV("",<TDDL>):	A GETTENV funkció felhívásánál megengedett az első paraméter elhagyása vagy első paraméterként nulla-string megadása. Ebben a különleges esetben a <TDDL>-ben az **aktív** szerszám T-, D- és DL-számai lesznek visszaadva.		

11.12.4 Információk olvasása a tárolt szerszám-környezethez ($P_TOOLENVN, ($P_TOOLENV)

Információk a tárolt szerszám-környezethez a következő rendszerváltozókkal olvashatók:

$P_TOOLENVN:	A TOOLENV-vel definiált (és még nem törölt) adatkészletek számát adja a szerszám-környezetek leírásához			
	szintaxis:	<n> = $P_TOOLENVN		
	jelentés:	<n>:	definiált adatkészletek száma	
			adattípus:	INT
			értéktartomány:	0 ... MD18116 $MN_MM_NUM_TOOL_ENV
	A hozzáférés ehhez a rendszerváltozóhoz akkor is megengedett, ha nem lehetséges szerszám-környezet (MD18116 = 0). Ebben az esetben a visszaadási érték "0".			
$P_TOOLENV:	Az <i>-dik adatkészletnek a nevét adja egy szerszám-környezet leírásához			
	szintaxis:	<név> = $P_TOOLENV[<i>]		
	jelentés:	<név>:	<i> számú adatkészlet neve	
			adattípus:	STRING
		<i>:	adatkészlet száma	
			adattípus:	INT
			értéktartomány:	1 ... $P_TOOLENVN
	A számok hozzárendelése az adatkészletekhez nem állandó, hanem az adatkészletek törlésével és újra létrehozásával változhat. Az adatkészletek számozása belül lesz végrehajtva. Ha az <i> egy nem definiált adatkészletre utal, a nulla-string lesz visszaadva. Ha az <i> index érvénytelen, azaz <i> kisebb 1-nél vagy nagyobb a szerszám-környezetek adatkészleteinek maximális számánál (MD18116 $MN_MM_NUM_TOOLENV), a következő vészjelzés lesz kiadva: 17020 "nem megengedett array index1" vészjelzés			

11.12.5 Szerszámhosszakat ill. szerszámhossz-komponenseket kiolvasni (GETTCOR)

A GETTCOR funkció szerszámhosszak ill. szerszámhossz komponensek olvasására szolgál.

Ennél paraméterezéssel meg lehet adni, hogy melyik komponensek legyenek figyelembe véve, és a szerszámot milyen alkalmazási feltételeknél kell figyelembe venni.

Szintaxis

```
<állapot> = GETTCOR(<Len>[, <Comp>, <Stat>, <T>, <D>, <DL>])
```

Jelentés

GETTCOR(...):	Előre definiált funkció szerszámhosszak ill. szerszámhossz-komponensek olvasására		
	egyedül a mondatban:		igen
<állapot>:	Funkció visszaadási érték. Negatív értékek hibaállapotot mutatnak.		
	adattípus:		INT
	érték:	0	Funkció OK
		-1	Nincs tárolóhely foglalva a szerszám-környezetek számára: MD18116 $MN_MM_NUM_TOOL_ENV = 0 Azaz a "Szerszám-környeztek" funkcionalitás nem létezik.
		-2	A szerszám-környezet a <Stat>-tal megadott névvel nem létezik.
		-3	Érvénytelen string a <Comp> paraméterben. Ezen hiba oka lehet érvénytelen vagy duplán programozott paraméter.
		-4	Érvénytelen T-szám
		-5	Érvénytelen D-szám
		-6	Érvénytelen DL-szám
		-7	Hozzáférési kísérlet egy nem meglevő tároló-modulhoz.
		-8	Hozzáférési kísérlet egy nem meglevő opcióhoz (pl. programozható szerszámtájolás, szerszámkezelés).
		-9	A <Comp> string tartalmaz egy kettőspontot (egy koordinátarendszer specifikációjának jelölése), de nem követi egy érvényes karakter az igényelt koordinátarendszer jelölésére.
Paraméter			

1	<Len>:	eredmény-vektor	
		adattípus:	REAL[11]
		A vektor komponensek a következő sorrendben vannak elrendezve: • <Len> [0]: szerszámtípus • <Len> [1]: vágóélhelyzet • <Len> [2]: abszcissza • <Len> [3]: ordináta • <Len> [4]: aplikáta • <Len> [5]: szerszámsugár A hossz komponensek vonatkoztatási koordinátarendszere a <Comp>-ban és <Stat>-ban definiált koordinátarendszer. Ha a <Comp>-ban nincs koordinátarendszer definiálva, a szerszámhosszak gép koordinátarendszerben lesznek ábrázolva. Az abszcissza, ordináta és aplikáta hozzárendelése a geometria-tengelyekhez függ a használt szerszám-környezet aktív síkjától. Azaz G17-nél az abszcissza párhuzamos X-szel, G18-nál Z-vel stb. A <Len>[6] ... <Len>[10] komponensek tartalmazzák a további paramétereket, amelyek meg lehetnek adva egy szerszám geometriájának leírásához (pl. $TC_DP7 ... $TC_DP11 a geometriára ill. megfelelő komponensek a kopásra ill. össz- és beállítás korrekciók). Ez az 5 további elem és a szerszámsugár csak az E, G, S és W komponensekre van definiálva. Ennek értékelése nem függ a <Stat>-tól. A megfelelő értékek a <Len>[6] ... <Len>[10]-ben ezért csak akkor lehetnek nullától eltérőek, ha a szerszámhossz kiszámításában legalább egy a négy megadott komponensből részt vesz. Az egyéb komponenseknek nincs hatásuk az eredményre. Az adatok a vezérlés alap-rendszerére vonatkoznak (hüvelyk vagy metrikus).	

2	<Comp>:	szerszámhossz-komponensek (opcionális)			
		adattípus:	STRING		
		A karakterlánc két rész-stringből áll, amelyek egymástól egy kettősponttal vannak elválasztva. ált. formátum: "<SubStr_1> [: <SubStr_2]"			
		<SubStr_1>:	Az **első rész-string** jelöli a szerszámhossz komponenseket, amelyeket a szerszámhossz kiszámításánál figyelembe kell venni. A karakterek sorrendje a rész-stringben ill. azok írásmódja (kicsi vagy nagy) tetszőleges. A karakterek közé tetszőleges számú üres jel vagy tabulátor jel (white spaces) lehet beszúrva. **Utalás:** A karaktereket a rész-stringben **duplán nem** szabad programozni!		
			karakter:	–	mínusz jel (csak első karakterként megengedett!) Ki lesz számítva az össz-szerszámhossz a komponensek kivonásával, amelyek a következő stringben specifikálva vannak.
				C	Adapter méret vagy alap méret (a két alternatív létező komponens közül annak, amelyik az alkalmazott szerszámra aktív)
				E	beállítás korrekciók
				G	geometria
				K	kinematikai transzformációk (csak általános 3-, 4-, és 5-tengelyes transzformációknál lesz kiértékelve!)
				S	összeg-korrekciók
				T	a tájolható szerszámtartó
				W	kopás
			Ha az első rész-string üres (kivétel white space-ek), azt jelenti, hogy a teljes szerszámhosszat az összes komponens figyelembe vételével kell kiszámítani. Ugyanez érvényes, ha a <Comp> paraméter nincs megadva.		
		<Substr_2>:	Az opcionális **második rész-string** jelöli a koordinátarendszert, amelyikben a szerszámhosszat ki kell adni. A második rész-string csak egyetlen releváns karakterből áll.		
			karakter:	A	beállítható nullapont koordinátarendszer (ACS)
				B	Alap-koordinátarendszer (AKR)
				K	Kinematikai transzformáció szerszám-koordinátarendszere (KKR)
				M	Gép-koordinátarendszer (GKR)
				T	Szerszám-koordinátarendszer (SKR)
				W	Munkadarab-koordinátarendszer (MKR)
			Ha nincs megadva koordinátarendszer, a kiértékelés a gép-koordinátarendszerben (GKR) történik. Az adott esetben figyelembe veendő forgatásokat a <Stat>-ban definiált szerszámkörnyezet adja meg.		

3	<_Stat>:	adatkészlet neve egy szerszám-környezet leírásához (opcionális)	
		adattípus:	STRING
		Ha ezen paraméter értéke a nulla-string ("") vagy nincs megadva, az aktuális állapot lesz használva. Ha nincs szerszám specifikálva, az aktuális szerszám lesz használva.	
4	<T>:	szerszám belső T-száma (opcionális)	
		adattípus:	INT
		Ha ezen paraméter nincs megadva vagy az értéke "0", a <Stat>-ban megadott érték lesz használva. Ha ezen paraméter értéke "-1", az aktív szerszám T-száma lesz használva. Megengedett az aktív szerszám számának közvetlen megadása is. **Utalás:** Ha a <Stat> nincs megadva, szerszám-környezetként az aktuális állapot lesz használva. Mivel utalás történik a <T> = 0-val a szerszám-környezetben eltárolt T-számra, ebben az aktív szerszám lesz használva, azaz a <T> = 0 és <T> = -1 megadások ebben a külön esetben azonos jelentésűek.	
5	<D>:	szerszám vágóél-száma (opcionális)	
		adattípus:	INT
		Ha ezen paraméter nincs megadva vagy az értéke "0", a használt D-szám a forrás T-számához igazodik. Ha a T-szám a szerszám-környezetből használva lesz, a szerszám-környezet D-száma is olvasva lesz, egyébként az aktuálisan aktív szerszámé.	
6	<DL>:	helyfüggő korrekció száma (opcionális)	
		adattípus:	INT
		Ha ezen paraméter nincs megadva, a használt DL-szám a forrás T-számához igazodik. Ha a T-szám a szerszám-környezetből használva lesz, a szerszám-környezet D-száma is olvasva lesz, egyébként az aktuálisan aktív szerszámé.	

Példák

`GETTCOR(_LEN)`	Kiszámítja az aktuálisan aktív szerszám szerszámhosszát a gép koordináta-rendszerben az összes komponens figyelembe vételével.
`GETTCOR(_LEN,"CGW:W")`	Az aktív szerszámra ki lesz számítva az adapter- ill. alap-méretből, geometriából és kopásból összeálló szerszámhossz. További részek, mint pl. tájolható szerszámtartó vagy kinematikai transzformációk nem lesznek figyelembe véve. A kiadás munkadarab koordináta-rendszerben történik.
`GETTCOR (_LEN,"-K:B")`	Az aktív szerszám össz-szerszámhossza az esetleg aktív kinematikai transzformáció hossz-komponenseinek figyelembe vétele nélkül lesz kiszámítva. Kiadás az alap-koordinátarendszerben
`GETTCOR (_LEN,":M","Testenv1",,3)`	A szerszám-környezetben a "Testenv1" néven eltárolt szerszám össz-szerszámhossza a gép-koordinátarendszerben lesz kiszámítva. A számítás azonban a megadott vágóél-számtól függetlenül a D3 vágóél-számra történik.

További információk

Adapter transzformáció / tájolható szerszámtartó / kinematikai transzformációk

A forgatások és a komponensek felcserélése, amelyeket adott estben az adapter transzformáció, tájolható szerszámtartó és egy kinematikai transzformáció hajt végre, a szerszám-környezet részei. Ezek ezért mindig végre lesznek hajtva, akkor is, ha a megfelelő hossz-komponenseket nem kell figyelembe venni. Ha ez nem kívánatos, olyan szerszám-környezeteket kell definiálni, amelyeknél a megfelelő transzformációk nem aktívak. Sok esetben (akkor mindig, ha egy gépen nincsenek transzformációk ill. tájolható szerszámtartó használva), a szerszám-környezet eltárolt adatkészletei automatikusan teljesítik ezeket a feltételeket, így ezeket a felhasználónak nem kell külön figyelembe venni.

Eszterga- és köszörűszerszámok szerszámhosszak kiszámítása az MD20360 $MC_TOOL_PARAMETER_DEF_MASK-tól függően

Azt, hogy hogyan kell eszterga- és köszörűszerszámoknál a kopást és a szerszámhosszakat egy esetlegesen meglevő átmérő-tengelyen kiértékelni, gépadatban van megadva.

MD20360 $MC_TOOL_PARAMETER_DEF_MASK (szerszám paraméter definíció)

Bit	Érték	
0	Az eszterga- és köszörűszerszámoknál a síktengely **kopás paramétere** ármérő értékként lesz beszámítva:	
	= 0 (alapérték)	nem
	= 1	igen
1	Az eszterga- és köszörűszerszámoknál a síktengely **szerszámhossz komponense** ármérő értékként lesz beszámítva:	
	= 0 (alapérték)	nem
	= 1	igen

Ha az érintett bitek be vannak állítva, a hozzájuk tartozó megadás 0,5-ös tényezővel lesz kiértékelve. Ez a kiértékelés megjelenik a GETTCOR által adott szerszámhosszakban is.

Példa:

MD20360 $MC_TOOL_PARAMETER_DEF_MASK = 3

MD20100 $MC_DIAMETER_AX_DEF (geometria-tengely sík-tengely funkcióval) = "X"

X az átmérő-tengely (szabványos esztergagép konfiguráció)

```
Programkód                          Kommentár
N30 $TC_DP1[1,1]=500
N40 $TC_DP2[1,1]=2
N50 $TC_DP3[1,1]=3.0                ; L1 geometria
N60 $TC_DP4[1,1]=4.0
N70 $TC_DP5[1,1]=5.0
N80 $TC_DP12[1,1]=12.0              ; L1 kopás
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)
```

Programkód	Kommentár
N140 R4=_LEN[3]	; 7.5 (= 0.5 * 3.0 + 0.5 * 12.0)
N150 R5=_LEN[4]	; 19.0 (= 5.0 + 14.0)
N160 M30	

Kinematikai transzformációk és tájolható szerszámtartó hossz-komponensei

Ha egy szerszámhossz számításnál figyelembe lesz véve egy **tájolható szerszámtartó**, a következő vektorok kerülnek be a szerszámhossz számításba:

Típus	Vektorok
M	l1 és l2
T	l1, l2 és l3
P	Szerszámhossz a tájolható szerszámtartó által nem lesz befolyásolva.

Az általános **5-tengelyes transzformációnál** a trafó-típus 24 és 56 esetén a következő gépadatok lesznek figyelembe véve a szerszámhossz számításnál:

Trafo-típus	Gépadatok
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

A trafo-típus 56 (mozgatható szerszám és mozgatható mumkadarab) megfelel az M típusnak a tájolható szerszámtartónál.

Az 5-tengelyes transzformációnál az eddigi szoftver kiadásokban a MD24560/24660 $MC_TRAFO5_JOINT_OFFSET_1/2 vektor (kinematikus eltolás vektor az 1./2. 5-tengelyes transzformációk a csatornában) megfelel az l_1 és l_3 vektorok összegének az M típusú tájolható szerszámtartónál.

A transzformációhoz mindkét esetben csak az összeg számít. A két komponens összetételének a módja nem számít. A szerszámhosszak számításánál azonban számít, hogy melyik részt kell a szerszámhoz és melyiket a szerszám-asztalhoz hozzárendelni. Ezért lett bevezetve az MD24558/24658 $MC_TRAFO5_JOINT_OFFSET_PART_1/2 (kinematikai eltolás vektor az asztalon) gépadat. Ez megfelel az l3 vektornak. Az MD24560/24660 $MC_TRAFO5_JOINT_OFFSET_1/2 gépadat most már nem az l1 és l3 összege, hanem az l1 vektor. Ha az MD24558/24658 $MC_TRAFO5_JOINT_OFFSET_PART_1/2 gépadat nulla, akkor a viselkedés azonos az eddigi viselkedéssel.

Kompatibilitás

A GETTCOR funkció együtt a TOOLENV és SETTCOR funkciókkal többek között az eddig a mérőciklusokban megvalósított funkcionalitás részbeni pótlását szolgálják.

A mérőciklusokban csak a paraméterek egy része lett kiértékelve, amelyek végülis a hatásos szerszámhosszat meghatározzák. A megnevezett funkciókat lehet úgy paraméterezni, hogy a mérőciklusok viselkedése a szerszámhossz számításának vonatkozásában reprodukálható.

11.12.6 Szerszám-komponenseket változtatni (SETTCOR)

A SETTCOR funkció a szerszám-komponensek megváltoztatását szolgálja az összes peremfeltétel figyelembe vételével, amelyek az egyes komponensek kiértékelésében részt vesznek.

Megjegyzés

A terminológiához: Ha a következőkben a szerszámhosszal összefüggésben a szerszám-komponensekről beszélünk, akkor a vektorosan szemlélt komponenseket gondoljuk, amelyekből a teljes szerszámhossz összetevődik (pl. geometria és kopás). Egy ilyen komponens ezért három egyedi értékből (L1, L2, L3) tevődik össze, amelyeket a következőkben koordináta-értékeknek nevezünk.

A "geometria" szerszám-komponens ezzel például a három $TC_DP3 ... $TC_DP5 koordináta-értékből áll.

Szintaxis

```
<állapot>= SETTCOR(<CorVal>, <Comp>, [<CorComp>, <CorMode>, <GeoAx>,
<Stat>, <T>, <D>, <DL>])
```

Jelentés

`SETTCOR(...)`:	Előre definiált funkció a szerszám-komponensek változtatásához	
	egyedül a mondatban:	igen

<állapot>:	Funkció visszaadási érték. Negatív értékek hibaállapotot mutatnak.		
	adattípus:		INT
	érték:	0	Funkció OK
		-1	Nincs tárolóhely foglalva a szerszám-környezetek számára: MD18116 $MN_MM_NUM_TOOL_ENV = 0 Azaz a "Szerszám-környeztek" funkcionalitás nem létezik.
		-2	A szerszám-környezet a <Stat>-tal megadott névvel nem létezik.
		-3	Érvénytelen string a <Comp> paraméterben. Ezen hiba oka lehet érvénytelen vagy duplán programozott paraméter.
		-4	Érvénytelen T-szám.
		-5	Érvénytelen D-szám.
		-6	Érvénytelen DL-szám.
		-7	Hozzáférési kísérlet egy nem meglevő tároló-modulhoz.
		-8	Hozzáférési kísérlet egy nem meglevő opcióhoz (pl. programozható szerszámtájolás, szerszámkezelés).
		-9	Nem megengedett számérték a <CorComp> paraméterre.
		-10	Nem megengedett számérték a <CorMode> paraméterre.
		-11	A <Comp> és <CorComp> paraméterek tartalmai ellentmondásosak.
		-12	A <Comp> és <CorMode> paraméterek tartalmai ellentmondásosak.
		-13	A <GeoAx> paraméter tartalma nem geometria-tengely.
		-14	Írás kísérlet egy nem meglevő beállítás korrekcióhoz.
Paraméter			
1	<CorVal>:	korrekció-vektor A <Stat>-ban definiált munkadarab-koordinátarendszerben (MKR) a következő hozzárendelés érvényes: • <CorVal> [0]: abszcissza • <CorVal> [1]: ordináta • <CorVal> [2]: aplikáta Ha csak egy szerszám-komponenst kell korrigálni (vagyis nincs vektoros korrekció, lásd <CorMode> paraméter), a korrekcióérték mindig a <CorVal>[0]-ban van, függetlenül, hogy melyik tengelyre hat. A két további komponens tartalma ekkor nem lesz kiértékelve. Ha a <CorVal> ill. a <CorVal> egyik komponense a síktengelyre vonatkozik, akkor az adat **sugár méretként** lesz kiértékelve. Ez azt jelenti, hogy egy szerszám pl. megadott értékkel "hosszabb" lesz, az megfelelően a munkadarab átmérő duplán akkora változását okozza. A méret adatok a vezérlés **alap-rendszerére** vonatkoznak (hüvelyk vagy metrikus).		
		adattípus:	REAL[3]

2	<Comp>:	szerszám-komponens(ek)			
		adattípus:	STRING		
		A karakterlánc két rész-stringből áll, amelyek egymástól egy kettősponttal vannak elválasztva. ált. formátum: "<SubStr_1> [: <SubStr_2]"			
		<SubStr_1>:	Az **első rész-string** mindig meg kell legyen és egy vagy két karakterből állhat. Ekkor az első vagy egyetlen karakter az 1. komponens (Val_1) és a második karakter a 2. komponens (Val_2), amelyek a következő <CorComp> és <CorMode> paramétereknek megfelelően lesznek feldolgozva.		
			karakter:	C	Adapter méret vagy alap méret (a két alternatív létező komponens közül annak, amelyik az alkalmazott szerszámra aktív)
				E	beállítás korrekciók
				G	geometria
				S	összeg-korrekciók
				W	kopás
		<Substr_2>:	A **második rész-string** opcionális. Alternatív módon állhat az (egyetlen) "W" vagy "T" betűből.		
			karakter:	W	Ha a második rész-string üres vagy a "W" betűt tartalmazza, a korrekció-értékek úgy lesznek beszámítva, mintha a **munkadarab** koordinátarendszerben (MKR) lettek volna bemérve.
				T	Ha a második rész-string a "T" betűt tartalmazza, a korrekció-értékek úgy lesznek beszámítva, mintha a **szerszám** koordinátarendszerben (**T**ool Coordinate System, TCS) lettek volna bemérve.
		A karakterek írásmódja (kicsi vagy nagy) a stringben tetszőleges. Tetszőleges számú üres jel vagy tabulátor jel (white spaces) lehet beszúrva.			

3	<CorComp>:	specifikálja a szerszám adatkészlet komponenseit, amelyeket le kell írni (opcionális)		
		adattípus:		INT
		érték:	0	A <CorVal>[0] korrekcióérték a <GeoAx> paraméterben átadott geometria-tengelyre vonatkozik a munkadarab koordináta-rendszerben vagy a szerszám koordináta-rendszerben (lásd ehhez a <Comp> paraméter leírását). Azaz a korrekcióértéket úgy kell beállítani a megjelölt szerszám-komponensekben, hogy az összes paraméter figyelembe vételével, amelyek befolyásolhatják a szerszámhossz számítást, eredményként az össz-szerszámhossznak a megadott tengelyirányba az előre megadott értékű változása adódjon. Ezt a változást a <Comp>-ban megadott komponensek és a <CorMode>-ban megadott szimbolikus számítási előírás (lásd a következő paramétereket) korrekciójával kell elérni. Az eredő korrekció ezért mind a három tengely-komponensre hatással lehet.
			1	Mint "0", csak vektorosan. A <CorVal> vektor tartalma az abszcisszára, ordinátára és aplikátára vonatkozik a munkadarab koordináta-rendszerben vagy a szerszám koordináta-rendszerben (lásd ehhez a <Comp> paraméter leírását). A következő <GeoAx> paraméter nem lesz kiértékelve.
			2	Vektoros korrekciók, azaz L1, L2 és L3 szimultán változtathatóak. A "0" és "1" változatokkal ellentétben a <CorVal>-ban levő korrekció-értékek a szerszám Val_1-komponenseinek korrekció-értékeire vonatkoznak (lásd a következő <CorMode> paramétert). A szerszámnak esetlegesen meglevő ferde helyzete a munkadarab koordináta-rendszerhez viszonyítva nem befolyásolja a korrekciót.
			3 - 5	Az L1 ... L3 ($TC_DP3 ... $TC_DP5) szerszámhosszak ill. a kopás, beállítás vagy össz-korrekciók megfelelő értékeinek korrekciója. A korrekcióérték a <CorVal>[0]-ban van. A mérés a szerszám Val_1-komponenseinek (lásd a következő <CorMode> paramétert) koordinátáiban történik. A szerszámnak esetlegesen meglevő ferde helyzete a munkadarab koordináta-rendszerhez viszonyítva nem befolyásolja a korrekciót.
			6	A szerszámsugár ($TC_DP6) ill. a kopás, beállítás vagy össz-korrekciók megfelelő értékeinek korrekciója. Az MD20360 $MC_TOOL_PARAMETER_DEF_MASK gépadat bit 10 és 11 (átmérő ill átmérő-kopás értékelése választhatóan sugár vagy átmérő adatként) figyelembe lesznek véve.
			7 – 11	A $TC_DP7 ... $TC_DP11 ill. a kopás, beállítás vagy össz-korrekciók megfelelő értékeinek korrekciója. Ezek a paraméterek úgy lesznek kezelve, mint a szerszámsugár.
		Ha ezen paraméter nincs megadva, az értéke "0".		

4	<CorMode>:	A végrehajtandó írás művelet módját specifikálja (opcionális)		
		adattípus:		INT
		érték:	0	$Val_{1új}$ = <CorVal>
			1	$Val_{1új}$ = $Val_{1régi}$ + <CorVal>
			2	$Val_{1új}$ = <CorVal> $Val_{2új}$ = 0
			3	$Val_{1új}$ = $Val_{1régi}$ + $Val_{2régi}$ + <CorVal> $Val_{2új}$ = 0
		A $Val_{1régi}$ + $Val_{2régi}$ írásmód szimbolikusan értendő. Ha a két komponens (a <_Stat> állapota alapján) eltérően lesz értékelve, azaz a két komponens között egy forgatás hatásos, akkor $Val_{2régi}$ az összeadás előtt úgy lesz transzformálva, hogy az eredő szerszámhossz a $Val_{2új}$ törlése után és a <CorVal> hozzáadása előtt változatlan marad. A <CorVal> mindig a Val_1-re vonatkozik. A <CorVal> egy érték, amely a <Comp> paraméter második részétől függően munkadarab koordináta-rendszerben (MKR) vagy szerszám-koordinátarendszer (SKR) lesz mérve. Ezért szemben a szerszám-komponensekkel, amelyekbe be lesz számítva, adott esetben már transzformálva van. Ezért nem lehet közvetlenül az eltárolt értékkel számításba venni, hanem a Val_1 ill. Val_2-höz hozzáadás előtt vissza kell transzformálni. Ennek következménye lehet, hogy a korrekció a <CorComp>-ban definiálttól eltérő tengelyre hat, ill. hogy ez több tengelyre hat. A <CorComp> = 0 esetre, azaz ha a <CorVal> nem vektort, hanem csak egy értéket tartalmaz, a leírt műveletek azokban a koordinátákban lesznek végrehajtva, amelyekben a <CorVal> mérve lett (MKR / TCS). Ez különösen érvényes a $Val_{2új}$ nullázására is a 2 és 3 változatokban. Ez az eredmény azután vissza lesz transzformálva a szerszám koordinátákba. Ennek lehet az a következménye, hogy a nullázandó koordinátaértékek (L1, L2, L3) egyike sem lesz nulla, ill. hogy a koordinátaértékek, amelyek előtte nullák voltak, most nem nullák lesznek. Ha azonban a megfelelő műveletek egymás után mind a három geometria-tengelyre végre lesznek hajtva, garantált, hogy a törlendő komponensek mindhárom koordinátaértéke is nulla lesz. Ha a szerszám a munkadarab koordináta-rendszerrel szemben nincs elforgatva, vagy úgy van elforgatva, hogy az összes szerszám-komponens a koordinátatengelyekkel párhuzamos marad (tengely felcserélések), garantálva van, hogy mindig csak egy szerszám-koordináta változik. Ugyanannak a műveletnek (<CorMode>) egymás utáni végrehajtása <CorComp> = 0-val mind a három koordinátatengelyre tetszőleges sorrendben azonos ugyanannak a műveletnek egyszeri végrehajtásával <CorComp>=1-gyel. A <Comp> paraméter a "0" és "1" paraméter-értékekre egy és a "2" és "3" paraméter-értékekre két karaktert kell tartalmazzon. Példa: <Comp> tartalma az "ES" string, <CorMode>-é a "2" érték ⇒ beállítás-korrekció$_{új}$ = <CorVal>, össz-korrekció$_{új}$ = 0 Ha a <CorMode> paraméter nincs megadva, az értéke "0".		
5	<GeoAx>:	Megadja a geometria-tengely indexét, amelyben a <CorVal>[0] korrekcióérték mérve lett (opcionális)		
		adattípus:		INT
		értéktartomány:		0 ... 2
		A 0 ... 2 indexek az abszcisszára, ordinátára és aplikátára vonatkoznak az aktuális szerszám-környezet hatásos síkjában (G17/G18/G19). Ezen paraméterek értéke csak akkor lesz kiértékelve, ha a <CorComp> paraméter értéke "0".		

6	<Stat>:	adatkészlet neve egy szerszám-környezet leírásához (opcionális)	
		adattípus:	STRING
		Ha ezen paraméter értéke a nulla-string ("") vagy nincs megadva, az aktuális állapot lesz használva. Ha nincs szerszám specifikálva, az aktuális szerszám lesz használva.	
7	<T>:	szerszám belső T-száma (opcionális)	
		adattípus:	INT
		Ha ezen paraméter nincs megadva vagy az értéke "0", a <Stat>-ban megadott érték lesz használva. Ha ezen paraméter értéke "-1", az aktív szerszám T-száma lesz használva. Megengedett az aktív szerszám számának közvetlen megadása is. **Utalás:** Ha a <Stat> nincs megadva, szerszám-környezetként az aktuális állapot lesz használva. Mivel utalás történik a <T> = 0-val a szerszám-környezetben eltárolt T-számra, ebben az aktív szerszám lesz használva, azaz a <T> = 0 és <T> = -1 megadások ebben a külön esetben azonos jelentésűek.	
8	<D>:	szerszám vágóél-száma (opcionális)	
		adattípus:	INT
		Ha ezen paraméter nincs megadva vagy az értéke "0", a használt D-szám a forrás T-számához igazodik. Ha a T-szám a szerszám-környezetből használva lesz, a szerszám-környezet D-száma is olvasva lesz, egyébként az aktuálisan aktív szerszámé.	
9	<TL>:	helyfüggő korrekció száma (opcionális)	
		adattípus:	INT
		Ha ezen paraméter nincs megadva, a használt DL-szám a forrás T-számához igazodik. Ha a T-szám a szerszám-környezetből használva lesz, a szerszám-környezet D-száma is olvasva lesz, egyébként az aktuálisan aktív szerszámé. Ha a T, D és DL egy helyfüggő korrekciók nélküli számot definiál, a <Comp> paraméterben nem szabad össz vagy beállítási korrekciókat specifikálni (hibakód a <Status>-ban).	

Megjegyzés

A három <Comp>, <CorComp> és <CorMode> paraméternek nem az összes kombinációja értelmes. Például a számítási előírás 3 a <CorComp>-ban két karakter megadását igényli a <Comp>-ban Ha egy nem megengedett paraméter kombináció lesz megadva, a <Status>-ban egy megfelelő hibakód kerül visszaadásra.

Példák

Példa 1

Programkód	Kommentár
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=120	; marószerszám
N30 $TC_DP3[1,1]=10.0	; L1 geometria
N40 $TC_DP12[1,1]=1.0	; L1 kopás
N50 _CORVAL[0]=0.333	

```
Programkód                                  Kommentár
N60 T1 D1 G17 G0
N70 R1=SETTCOR(_CORVAL,"G",0,0,2)
N80 T1 D1 X0 Y0 Z0                          ; ==> GKR pozíció X0.000 Y0.000 Z1.333
N90 M30
```

<CorComp> "0", ezért a geometria-komponens Z irányban ható koordinátaértéke a 0,333 korrekcióértékkel helyettesítve.

Az eredő szerszám össz-hossz ezzel: L1 = 0,333 + 1,000 = 1,333

Példa 2

```
Programkód                                  Kommentár
N10 DEF REAL _CORVAL[3]
N20 $TC_DP1[1,1]=120                        ; marószerszám
N30 $TC_DP3[1,1]=10.0                       ; L1 geometria
N40 $TC_DP12[1,1]=1.0                       ; L1 kopás
N50 _CORVAL[0]=0.333
N60 T1 D1 G17 G0
N70 R1=SETTCOR(_CORVAL,"W",0,1,2)
N80 T1 D1 X0 Y0 Z0                          ; ==> GKR pozíció X0.000 Y0.000 Z11.333
N90 M30
```

<CorComp> "1", ezért a Z irányban ható 0,333 korrekcióérték az 1,0 korrekcióértékhez lesz hozzáadva.

Az eredő szerszám össz-hossz ezzel: L1 = 10,0 + 1,333 = 11,333

Példa 3

```
Programkód                                  Kommentár
N10 DEF REAL _CORVAL[3]
N20 $TC_DP1[1,1]=120                        ; marószerszám
N30 $TC_DP3[1,1]=10.0                       ; L1 geometria
N40 $TC_DP12[1,1]=1.0                       ; L1 kopás
N50 _CORVAL[0]=0.333
N60 T1 D1 G17 G0
N70 R1=SETTCOR(_CORVAL,"GW",0,2,2)
N80 T1 D1 X0 Y0 Z0                          ; ==> GKR pozíció X0.000 Y0.000 Z0.333
N90 M30
```

<CorComp> "2", ezért a Z irányban ható korrekció be lesz vive a geometria-komponensekbe (a régi érték felül lesz írva), és a kopásérték törölve lesz.

Az eredő szerszám össz-hossz ezzel: L1 = 0,333 + 0,0 = 0,333

Példa 4

```
Programkód                                  Kommentár
N10 DEF REAL _CORVAL[3]
N20 $TC_DP1[1,1]=120                        ; marószerszám
```

```
Programkód                                  Kommentár
N30 $TC_DP3[1,1]=10.0                       ; L1 geometria
N40 $TC_DP12[1,1]=1.0                       ; L1 kopás
N50 _CORVAL[0]=0.333
N60 T1 D1 G17 G0
N70 R1=SETTCOR(_CORVAL,"GW",0,3,2)
N80 T1 D1 X0 Y0 Z0                          ; ==> GKR pozíció X0.000 Y0.000 Z11.333
N90 M30
```

<CorComp> "3", ezért a kopásérték és a korrekcióérték a geometria-komponensekhez lesz hozzáadva és a kopás-komponensek törölve lesznek.

Az eredő szerszám össz-hossz ezzel: L1 = 11,333 + 0,0 = 11,333.

Példa 5

```
Programkód                                  Kommentár
N10 DEF REAL _CORVAL[3]
N20 $TC_DP1[1,1]=120                        ; marószerszám
N30 $TC_DP3[1,1]=10.0                       ; L1 geometria
N40 $TC_DP12[1,1]=1.0                       ; L1 kopás
N50 _CORVAL[0]=0.333
N60 T1 D1 G17 G0
N70 R1=SETTCOR(_CORVAL,"GW",0,3,0)
N80 T1 D1 X0 Y0 Z0                          ; ==> GKR pozíció X0.333 Y0.000 Z11.000
N90 M30
```

<CorComp> "3", mint az előző példában, a korrekció most az index "0"-jú geometria-tengelyre (X tengely) hat, amelyikhez a G17 miatt a marószerszámnál az L3 szerszám-komponens van hozzárendelve. A SETTCOR felhívása ezért nem befolyásolja a $TC_DP3 és $TC_DP12 szerszám-paramétereket. A korrekcióérték ehelyett a $TC_DP5-be lesz bevive.

Példa 6

```
Programkód                                  Kommentár
N10 DEF REAL _CORVAL[3]
N20 $TC_DP1[1,1]=500                        ; esztergaszerszám
N30 $TC_DP3[1,1]=10.0                       ; L1 geometria
N40 $TC_DP4[1,1]=15.0                       ; L2 geometria
N50 $TC_DP12[1,1]=10.0                      ; L1 kopás
N60 $TC_DP13[1,1]=0.0                       ; L2 kopás
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                         ; ==> GKR pozíció X24.330 Y0.000 Z17.500
N120 M30
```

A szerszám egy esztergaszerszám. Az N80 mondatban egy frame-forgatás lesz aktiválva úgy, hogy az alap-koordinátarendszer (AKR) a munkadarab-koordinátarendszerrel (MKR)

szemben el van forgatva. A korrekcióérték (N70) az MKR-ben az "1"-es indexű geometria-tengelyre hat, azaz mivel G18 aktív, az X tengelyre. Mivel <CorMode> = 3 érvényes, a szerszámkopás a MKR X tengelye irányában az N100 végrehajtása után nulla kell legyen.

A lényeges szerszám-paraméterek tartalma a programvégen ezért:

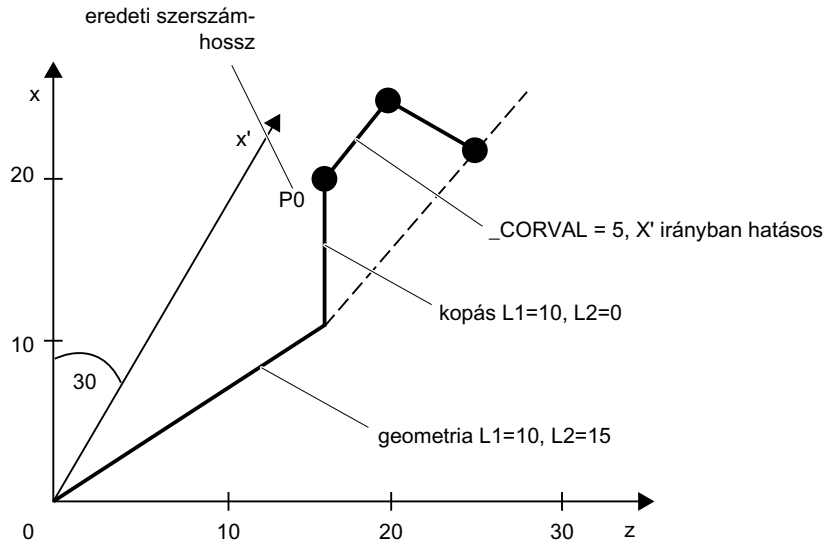
$TC_DP3[1,1] : 21.830 ; L1 geometria

$TC_DP4[1,1] : 21.830 ; L2 geometria

$TC_DP12[1,1] : 2 500 ; L1 kopás

$TC_DP13[1,1] : -4.330 ; kopás L2

A geometriai viszonyok a következő képen vannak ábrázolva. A teljes kopás, beleértve a _CORVAL-t, a MKR X' irányába lesz kivetítve. Ez a P2 pontot adja. Ezen pont koordinátái (X-Y koordinátákban mérve) a szerszám geometria-komponenseibe lesznek bevive. A kopásban a P_2 - P_1 eltérésvektor marad meg. Ezzel a kopásnak nincs többé komponense a _CORVAL irányában.



Ha a példa programot az N110 után a következő utasítással folytatjuk, a maradék kopás teljesen át lesz véve a geometriába, mivel a korrekció most a Z' tengelyen hat (paraméter <GeoAx> = 0):

```
N120 _CORVAL[0]=0.0
N130 R1=SETTCOR(_CORVAL,"GW",0,3,0)
N140 T1 D1 X0 Y0 Z0 ; ==> GKR pozíció X24.330 Y0.000 Z17.500
```

Mivel az új korrekcióérték "0", a szerszám össz-hossza és azzal az N140-ben felvett pozíció sem változhat. Ha a _CORVAL az N120-ban nem lenne "0", egy új szerszám össz-hossz és azzal egy új megváltozott pozíció is adódna az N140-ben, a szerszámhosszak kopás-része azonban ebben az esetben nulla lenne, azaz a teljes szerszámhossz ezután minden esetben benne a szerszám geometria-részében.

Ugyanazt az eredményt, mint a SETTCOR funkció kétszeri felhívásával a <CorComp> = 0 paraméterrel, el lehet érni az egyszeri felhívásával <CorComp> = 1-gyel (vektoros korrekció):

```
Programkód                                  Kommentár
N10 DEF REAL _CORVAL[3]
N20 $TC_DP1[1,1]=500                        ; esztergaszerszám
N30 $TC_DP3[1,1]=10.0                       ; L1 geometria
N40 $TC_DP4[1,1]=15.0                       ; L2 geometria
N50 $TC_DP12[1,1]=10.0                      ; L1 kopás
N60 $TC_DP13[1,1]=0.0                       ; L2 kopás
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                         ; ==> GKR pozíció X24.330 Y0.000 Z17.500
N120 M30
```

Ebben az estben a szerszám összes kopás-komponense rögtön a SETTCOR első felhívása után az N100-ban nulla.

Példa 7

```
Programkód                                  Kommentár
N10 DEF REAL _CORVAL[3]
N20 $TC_DP1[1,1]=500                        ; esztergaszerszám
N30 $TC_DP3[1,1]=10.0                       ; L1 geometria
N40 $TC_DP4[1,1]=15.0                       ; L2 geometria
N50 $TC_DP12[1,1]=10.0                      ; L1 kopás
N60 $TC_DP13[1,1]=0.0                       ; L2 kopás
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                         ; ==> GKR pozíció X25.000 Y0.000 Z15.000
N120 M30
```

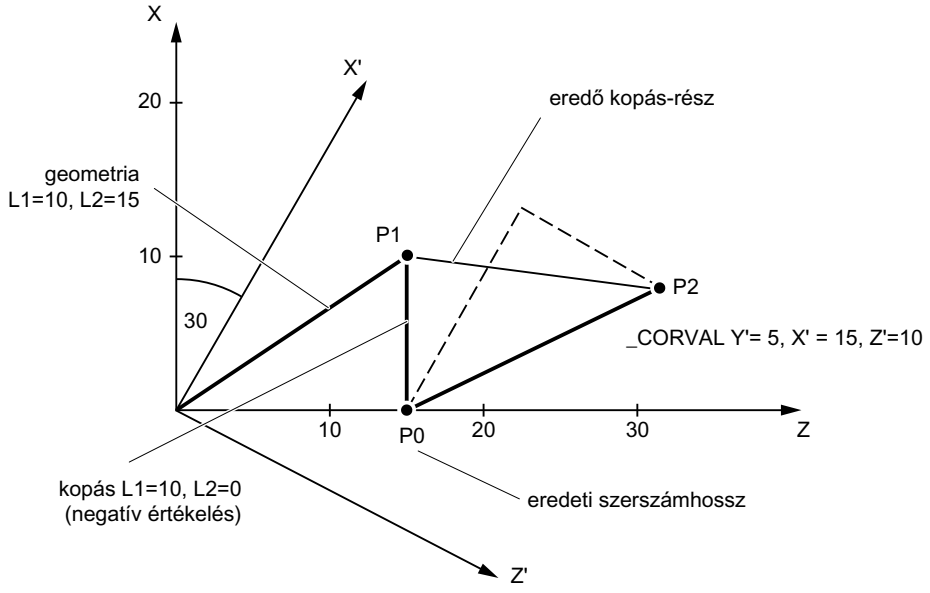
A 6-os példával szemben itt a <CorComp> paraméter = 3, ezért a <GeoAx> paraméter megadása hiányozhat. A _CORVAL[0]-ban levő érték most közvetlenül az L1 szerszám-komponensre hat, a forgatásnak az N80-ban nincs hatása az eredményre, a kopás-komponens a $TC_DP12-ben együtt a _CORVAL[0]-val a geometria részbe lesz átvéve, úgy hogy a $TC_DP13 miatt a teljes szerszámhossz már a SETTCOR első felhívása után az N100-ban a szerszám geometria részében található.

Példa 8

```
Programkód                                  Kommentár
N10 DEF REAL _CORVAL[3]
```

Programkód	Kommentár
N20 $TC_DP1[1,1]=500	; esztergaszerszám
N30 $TC_DP3[1,1]=10.0	; L1 geometria
N40 $TC_DP4[1,1]=15.0	; L2 geometria
N50 $TC_DP5[1,1]=20.0	; L3 geometria
N60 $TC_DP12[1,1]=10.0	; L1 kopás
N70 $TC_DP13[1,1]=0.0	; L2 kopás
N80 $TC_DP14[1,1]=0.0	; L3 kopás
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	; ==> GKR pozíció X7.990 Y25.000 Z31.160
N170 M30	

Az N90-ben az SD42930 $SC_WEAR_SIGN beállítási adat be lesz beállítva, azaz a kopást negatív előjellel kell kiértékelni. A korrekció vektoros (<CorComp> = 1), és a korrekció-vektort hozzá kell adni a kopáshoz (<CorMode> = 1). A geometriai viszonyok a Z-X síkban a következő képen vannak ábrázolva.



<CorMode> = 1 miatt a szerszám geometria-része változatlan. Az MKR-ben (forgatás Y tengely körül) definiált korrekció-vektort így át kell venni a kopás részbe, hogy a teljes szerszámhossz a 3-as képen a P_2 pontra utal. A szerszám eredő kopás részét ezért a P_1 és P_2 pontok távolsága adja meg.
De mivel a kopás az SD42930 $SC_WEAR_SIGN beállítási adat miatt negatívan lesz értékelve, az így megállapított korrekciót negatív előjellel kell a korrekció-tárolóba bevinni. A lényeges szerszám-paraméterek tartalma a programvégen ezért:

$TC_DP3[1,1] : 10.000 ; L1 geometria (változatlan)

$TC_DP4[1,1] : 15.000 ; L2 geometria (változatlan)

$TC_DP5[1,1] : 10.000 ; L3 geometria (változatlan)

$TC_DP12[1,1] : 2.010 ; L1 kopás (= 10 - 15 * cos(30) + 10 * sin(30))

$TC_DP13[1,1] : -16.160 ; kopás L2 (= -15 * sin(30) - 10 * cos(30))

$TC_DP14[1,1] : -5.000 ; kopás L3

Az L3 komponensen az Y irányban felismerhető a SD42930 $SC_WEAR_SIGN beállítási adat hatása a további komplikálás nélkül a frame forgatás által.

További információk

Eszterga- / köszörűszerszámok szerszámhosszak kiszámítása az MD20360 $MC_TOOL_PARAMETER_DEF_MASK-tól függően

Azt, hogy hogyan kell eszterga- / köszörűszerszámoknál a kopást és a szerszámhosszakat egy esetlegesen meglevő átmérő-tengelyen kiértékelni, gépadatban van megadva.

MD20360 $MC_TOOL_PARAMETER_DEF_MASK.<bit> = <érték>

<bit>	<érték>	Jelentés
0	0	Az eszterga- / köszörűszerszámoknál a síktengely **kopás paramétere sugár értékként** lesz beszámítva:
	1	Az eszterga- / köszörűszerszámoknál a síktengely **kopás paramétere ármérő értékként** lesz beszámítva:
1	0	Az eszterga- / köszörűszerszámoknál a síktengely **szerszámhossz komponense sugár értékként** lesz beszámítva:
	1	Az eszterga- / köszörűszerszámoknál a síktengely **szerszámhossz komponense ármérő értékként** lesz beszámítva:

Ha az érintett bitek be vannak állítva, a hozzájuk tartozó megadás 0,5-ös tényezővel lesz kiértékelve. A korrekció a SETTCOR-ral úgy lesz végrehajtva, hogy a teljes hatásos szerszámhossz változás egyenlő a <CorVal>-ban átadott értékkel. Ha egy hossz az MD20360 $MC_TOOL_PARAMETER_DEF_MASK gépadat alapján a hossz számításánál a 0,5-ös tényezővel lesz kiértékelve, ezen komponensek korrekciója ezért a dupla átadott értékkel kell történjen.

Példa

MD20360 $MC_TOOL_PARAMETER_DEF_MASK = 2 (szerszámhosszat az átmérő-tengelyen 0,5-ös tényezővel kell kiértékelni)

X tengely az átmérő tengely.

```
Programkód                                  Kommentár
N10 DEF REAL _LEN[11]
N20 DEF REAL _CORVAL[3]
N30 $TC_DP1[1,1]=500                        ; szerszámtípus:
N40 $TC_DP2[1,1]=2                          ; vágóélhelyzet
N50 $TC_DP3[1,1]=3.                         ; geometria - hossz 1
N60 $TC_DP4[1,1]=4.                         ; geometria - hossz 2
```

Programkód	Kommentár
N70 $TC_DP5[1,1]=5.	; geometria - hossz 3
N80 _CORVAL[0]=1.	
N90 _CORVAL[1]=1.	
N100 _CORVAL[2]=1.	
N110 T1 D1 G18 G0 X0 Y0 Z0	; ==> GKR pozíció X1.5 Y5 Z4
N120 R1=SETTCOR(_CORVAL,"G",1,1)	
N130 T1 D1 X0 Y0 Z0	; ==> GKR pozíció X2.5 Y6 Z5
N140 R3=$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	

A szerszámhossz korrekciója minden tengelyen 1 mm kell legyen (N80 ... N100). Az L2 és L3 hosszakban az eredeti hosszhoz 1-1 mm lesz hozzáadva. Az eredeti szerszámhosszhoz az L1-ben ezzel szemben a dupla korrekcióérték (2 mm) lesz hozzáadva, ezzel az össz-szerszámhossz az igény szerint 1 mm-rel változik. Az N110 és N130 mondatokban felvett pozíciók összehasonlításában felismerhető, hogy minden tengelypozíció 1 mm-rel megváltozott.

11.13 L1, L2, L3 szerszámhosszak hozzárendelése a koordináta-tengelyek olvasásához (LENTOAX)

A LENTOAX funkció információkat ad az **aktív** szerszám L1, L2 és L3 szerszámhosszainak a hozzárendeléséről az abszcisszához, ordinátához és applikátához. Az abszcissza, ordináta és aplikáta hozzárendelése a geometria-tengelyekhez a frame-k és az aktív sík (G17 -G19) által befolyásolva lesz.

Ennél egy szerszámnak csak a geometria része ($TC_DP3[<t>,<d>] ... $TC_DP5[<t>,<d>]) lesz figyelembe véve, azaz további komponenseknek (pl. kopás) egy esetleg attól eltérő tengely-hozzárendelésének az eredményre nincs hatása.

Szintaxis

```
<Status> = LENTOAX(<AxInd>, <Matrix>[, <Coord>])
```

Elv

LENTOAX(...):	A LENTOAX funkció információkat ad az aktív szerszám L1, L2 és L3 szerszámhosszainak a hozzárendeléséről az abszcisszához, ordinátához és applikátához.		
	egyedül a mondatban:		igen
<állapot>:	Funkció visszaadási érték. Negatív értékek hibaállapotot mutatnak.		
	adattípus:		INT
	érték:	0	Funkció OK Az információ az <AxInd>-ben elegendő a leíráshoz (az összes szerszámhossz-komponens párhuzamos a geometria-tengelyekkel).
		1	A funkció OK, a konkrét leíráshoz azonban a <Matrix> tartalmát ki kell értékelni (a szerszámhossz-komponensek nem párhuzamosak a geometria-tengelyekkel).
		-1	Érvénytelen string a <Coord> paraméterben.
		-2	Nincs aktív szerszám.
Paraméter			

1	<AxInd>:	Ha a szerszámhossz-komponensek párhuzamosak a geometria-tengelyekkel, a tengely indexek, amelyek az L1 ... L3 hossz-komponensekhez vannak rendelve, az **<**AxInd**>** mezőben lesznek visszaadva. • <AxInd> [0]: abszcissza • <AxInd> [1]: ordináta • <AxInd> [2]: aplikáta		
		adattípus:		INT[3]
		érték:	0	Nincs hozzárendelés (tengely nem létezik)
			1 ... 3 ill. -1 ... -3	A hossz száma, amelyik a megfelelő koordináta-tengelyen hat. Az előjel negatív, ha a szerszámhossz komponens a negatív koordináta-irányba mutat.
		Ha nem minden hossz-komponens párhuzamos ill. nem párhuzamos a geometria-tengelyekkel, az **<**AxInd**>**-ben azon tengelynek az indexe lesz visszaadva, amelyiknek egy szerszám-komponensben legnagyobb a része. Ebben az esetben (ha a funkció más okból nem jelez hibát) a visszaadási érték <Status> = 1. Az L1 ... L3 szerszámhossz-komponensek leképezése az 1 ... 3 geometria-tengelyekre a 2. **<**Matrix**>** paraméter tartalma által teljesen le van írva.		
2	<Matrix>:	A Matrix, amelyik a szerszámhosszak vektort (L1=1, L2=1, L3=1) a koordináta-tengelyek vektorba (abszcissza, ordináta, aplikáta) leképezi, azaz az oszlopokhoz a szerszámhossz-komponensek vannak hozzárendelve az L1, L2, L3 sorrendben, a sorokhoz a tengelyek az abszcissza, ordináta, aplikáta sorrendben.		
		adattípus:		REAL
		A Matrix-ban mindig az összes elem érvényes, akkor is, ha az egyik koordináta-tengelyhez tartozó geometria-tengely nem létezik, azaz a megfelelő érték az <AxInd>-ben nulla.		
3	<Coord>:	Koordinátarendszer, amelyre a hozzárendelés érvényes (opcionális)		
		adattípus:		STRING
		karakter:	MCS M	szerszámhossz leképezése a gép-koordinátarendszerbe
			BCS B	szerszámhossz leképezése az alap-koordinátarendszerbe
			MKR W	szerszámhossz leképezése a munkadarab-koordinátarendszerbe (alapérték)
			KCS K	szerszámhossz leképezése a kinematikai transzformáció szerszám-koordinátarendszerébe
			TCS T	szerszámhossz leképezése a szerszám-koordinátarendszerbe
		A karakterek írásmódja (kicsi vagy nagy) a stringben tetszőleges. Ha a **<**Coord**>** paraméter nincs megadva, az MKR lesz használva (alapérték).		

Megjegyzés

A TCS-ben a szerszámhossz-komponensek mindig párhuzamosak vagy nem párhuzamosak a tengelyekkel.

A komponensek csak akkor lehetnek nem párhuzamosak, ha a tükrözés aktív és következő beállítási adat be van állítva:

SD42900 $SC_MIRROR_TOOL_LENGTH (előjelváltó szerszámhossz tükrözésnél)

Példa

Marószerszám szabványos eset G17-nél

L1 Z-ben hat (aplikáta), L2 Y-ben hat (ordináta), L3 X-ben hat (abszcissza).

Funkció felhívása következő formában:
```
<Status>=LENTOAX(<AxInd>,<Matrix>,"WCS")
```

Az <AxInd> eredmény paraméter ekkor a következő értékeket tartalmazza:

<AxInd>[0] = 3

<AxInd>[1] = 2

<AxInd>[2] = 1

vagy röviden: (3, 2, 1)

A hozzá tartozó Matrix (<Matrix>) ebben az esetben:

$$<\text{Matrix}> = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

A váltás a G17-ről G18-ra vagy G19-re nem változtatja meg az eredményt, mivel a hossz-komponensek hozzárendelése a geometria-tengelyekhez azonos módon változik, mint az abszcissza, ordináta és aplikáta hozzárendelése.

Csak aktív G17-nél lesz egy frame-forgatás a Z körül 60 fokkal programozva, pl.:
```
ROT Z60
```

Az aplikáta iránya (Z irány) változatlan marad, az L2 fő része most az új X tengely irányában, az L1 fő része a negatív Y tengely irányában van A visszaadási érték (<állapot>) ezért "1", az <AxInd> a (2, -3, 1) értékeket tartalmazza.

A hozzá tartozó Matrix (<Matrix>) ebben az esetben:

$$<\text{Matrix}> = \begin{pmatrix} 0 & \sin 60° & \cos 60° \\ 0 & \cos 60° & -\sin 60° \\ 1 & 0 & 0 \end{pmatrix}$$

Pályaviselkedés 12

12.1 Érintőleges vezérlés

12.1.1 Csatolást definiálni (TANG)

Az előre definiált TANG(...) eljárás egy érintőleges csatolást definiál egy körtengely, mint követő tengely és két geometria-tengely, mint vezető tengelyek között. A követő tengely tengely ennél folyamatosan a vezető tengely pálya-érintőjére irányul.

Megjegyzés

csatolási tényező:

Az 1-es csatolási tényezőt nem kell külön programozni.
A -1-es csatolási tényezővel az érintő-tengely iránya megfordul.

Szintaxis

```
TANG(<követő tengely>, <vezető tengely_1>, <vezető tengely_2>,
<csatolási tényező>, <koordinátarendszer>, <optimalizálás>)
```

Jelentés

`TANG(...)`:	egy érintőleges csatolás definiálása		
`<követő tengely>`:	követő tengely tengely-neve (körtengely)		
	adattípus:	AXIS	
	értéktartomány:	csatornanevek	
`<vezető tengely_1>` `<vezető tengely_2>`:	vezető tengelyek (geometria-tengelyek) tengely-nevei [1]		
	adattípus:	AXIS	
	értéktartomány:	csatorna geometria-tengely neve	
`<csatolási tényező>`:	n tényező a követő tengely szögváltozása és a vezető tengelyek pálya-érintő változása között: $\text{szögváltozás}_{\text{követő tengely}} = \text{szögváltozás}_{\text{pálya-érintő}} * n$		
	adattípus:	REAL	
	alapérték:	1,0	
`<koordinátarendszer>`:	hatásos koordinátarendszer [2]		
	adattípus:	CHAR	
	érték:	`"B"`:	alap-koordinátarendszer (alapérték)
		`"W"`:	munkadarab-koordinátarendszer (nem áll rendelkezésre)

<optimalizálás>:	optimalizálási mód		
	adattípus:	CHAR	
	érték:	"S":	alap-beállítás (alapérték) A körtengely dinamikájának nincs visszahatása a vezető tengelyekre. Ha a körtengely dinamikája nagyobb, mint ami az utánvezetéshez szükséges, ez ez eljárás kielégítően pontos. Ha a körtengely dinamikája nem elég magas a pálya-érintő változásainak követéséhez, a körtengely iránya eltér a parancs-iránytól egy nem definiált átsimítási út mentén.
		"P":	A körtengely dinamikája a vezető tengelyek pálya-tervezésénél figyelembe lesz véve. Ehhez az érintőleges vezérlés bekapcsolásánál a TANGON()-nal meg kell adni két további paramétert. • átcsiszolási út • szögtűrés Lásd a "Csatolást bekapcsolni (TANGON) (Oldal 501)" fejezetben **Utalás** A kinematikai transzformációkkal összefüggésben ajánlott a "P" optimalizálási módot használni.
Utalás Az alapértékeket nem kell külön programozni.			
[1] **Utalás** Az érintőleges vezérlés vezető tengelyeiként azokat a geometria-tengelyeket kell használni, amelyek a gép alaphelyzetére vonatkoztatva, a gép-koordinátarendszerben (GKR) a programozott pályán mozognak. Ha pl. egy marógépen billenőfejjel használjuk a CYCLE800 billentés ciklust, a ciklus konfigurációjától függően az interpoláció MKR-ben történik az X és Y geometria-tengelyekkel. Az érintőleges csatolást azonban a geometria-tengelyekkel, mint vezető tengelyekkel kell definiálni, amelyek a GKR-ben programozott pályát teszik meg. Ehhez vezető tengelyekként a geometria-tengelyeket a gép **nem billentett** állapotában kell használni.			
[2] **Utalás** Az alap-koordinátarendszer (AKR) nem lehet elforgatva a GKR-hez képest. Ha pl. a ROT utasítással vagy a CYCLE800 billentés ciklussal az AKR el lesz forgatva, az érintőleges vezérlés többé nem működik helyesen.			

12.1.2 Közbenső mondat létrehozás bekapcsolása (TLIFT)

Ha a követő tengely érintő változása egy helyen túllépi a vezető tengelyek programozott pályájának egy helyén a MD37400 $MA_EPS_TLIFT_TANG_STEP gépadatban paraméterezett határértéket, a további pálya-tervezés a beállított sarok-viselkedéstől függően történik. Az előre definiált TLIFT(...) eljárás használat nélkül a mozgás a TANG(...) (Oldal 499)-gal és TANGON(...) (Oldal 501)-nal összefüggésben programozott átköszörülési eljárásnak megfelelően történik.

közbenső mondat létrehozás bekapcsolása

A TLIFT(...) programozásával a TANG(...) után az előrefutás a pálya ezen helyén egy sarok felismerésekor a vezérlés által automatikusan generált közbenső mondatot szúr be.

A program végrehajtásánál a közbenső mondat elérésekor a vezető tengelyek megállnak. A közbenső mondatban a követő tengely a maximális tengely-dinamikával a következő mondat pálya-érintőjének irányába lesz elforgatva. Ezután a vezető tengelyek tovább mennek a programozott pályán.

Közbenső mondat létrehozás kikapcsolása

A közbenső mondat létrehozás kikapcsolásához az érintőleges vezérlést TANG(...) újra definiálni kell, de utána nem kell bekapcsolni a közbenső mondat létrehozást TLIFT(...)-tel.

Szintaxis

```
TLIFT(<követő-tengely>)
```

Jelentés

`TLIFT(...)`:	sarok felismerés bekapcsolása közbenső mondat kiszámítással	
`<követő tengely>`:	követő tengely tengely-neve (körtengely)	
	adattípus:	AXIS
	értéktartomány:	csatornanevek

Követő tengely forgás sebessége

Pályatengely

Ha a követő tengely az érintőleges csatolás aktiválása előtt már pályatengelyként mozgott, a forgató mozgás a közbenső mondatban pályatengelyként történik.

A vonatkoztatási sugár `FGREF[<Achse>]=0.001` megadásával a forgató mozgás a paraméterezett maximális tengely-sebességgel történik.

MD32000 $MA_MAX_AX_VELO[<követő tengely>]

Pozicionáló tengely

Ha a követő tengely az érintőleges csatolás aktiválása előtt még nem pályatengelyként mozgott, a forgató mozgás a közbenső mondatban pozicionáló tengelyként történik.

A forgató mozgás ekkor a paraméterezett pozicionáló sebességgel történik:

MD32060 $MA_POS_AX_VELO[<követő tengely>]

12.1.3 Csatolást bekapcsolni (TANGON)

Az előre definiált TANGON(...) eljárással az előtte TANG(...) (Oldal 499)-gal definiált érintőleges vezérlés lesz bekapcsolva. A követő tengely ezt követően a vezető tengely mozgásánál folyamatosan a pálya-érintőjére irányul.

Követő tengely szöge

A szög, amit a követő tengely a pálya-érintőre vonatkoztatva felvesz, függ a TANG(...)-ban előre megadott áttételi aránytól, a MD37402 $MA_TANG_OFFSET gépadatban paraméterezett offset szögtől és a hozzáadódóan ható TANGON(...)-nál előre megadott offset szögtől.

"P" optimalizálás

Ha a (TANG(...)) érintőleges csatolás definiálásánál optimalizálási paraméterként a "P" érték lett megadva, a csatolás bekapcsolásánál meg kell adni az "átköszörülési út" paramétert és opcionálisan a "szögtűrés" paramétert.

Ha szögtűrésként a 0 érték lett megadva, csak az "átcsiszolási út" paraméter hat.

Ha szögtűrésként a 0-nál nagyobb érték lett megadva, a hatásos átcsiszolási út a paraméterezett átcsiszolási út és a paraméterezett szögtűrésen alapuló átcsiszolási út minimumából adódik.

Ha a követő tengely dinamikája nem elég a paraméterezett feltételek követéséhez, a vezető tengelyek pálya-sebessége megfelelően csökkentve lesz.

Szintaxis

```
TANGON(<követő tengely>, <offset szög>, <átsimítási út>, <szögtűrés>)
```

Jelentés

`TANGON(...)`:	érintőleges csatolás bekapcsolása	
`<követő tengely>`:	követő tengely tengely-neve (körtengely)	
	adattípus:	AXIS
	értéktartomány:	csatornanevek
`<offset-szög>`:	követő tengely offsetszöge a pálya-érintőhöz Vonatkoztatási pont a körtengely nullapontja.	
	adattípus:	REAL
`<átmenet simítás út>`:	maximális megengedett átmenet simítás út Ha az átmenet simítás út a dinamika feltételek alapján nagyobb lenne, a vezető tengelyek pálya-sebessége csökkentve lesz.	
	adattípus:	REAL
`<szögtűrés>`:	maximálisan megengedett tűrés a követő tengely és a pálya-érintő közötti előre megadott szög vonatkozásában	
	adattípus:	REAL

12.1.4 Csatolást kikapcsolni (TANGOF)

Az előre definiált TANGOF(...) eljárással a TANG(...) (Oldal 499)-gal definiált és TANGON(...) (Oldal 501)-nal bekapcsolt érintőleges vezérlés lesz kikapcsolva. A követő tengely tengely ezután többé nem a vezető tengely pálya-érintőjére irányul. Azonban a követő tengely csatolása a vezető tengelyhez a kikapcsolás után is megmarad, ami pl. meggátolja a következő funkciókat:

- síkváltás
- geometria-tengely átkapcsolás
- egy új érintőleges csatolás definiálása a követő tengelyre

A követő tengely csatolásának teljes megszüntetése a vezető tengelyhez csak a csatolásnak a TANGDEL(...) (Oldal 503)-lel való törlése után következik be.

Programozás

```
TANGOF(<követő tengely>)
```

Jelentés

TANGOF(...):	érintőleges csatolás kikapcsolása	
<követő tengely>:	követő tengely tengely-neve (körtengely)	
	adattípus:	AXIS
	értéktartomány:	csatornanevek

12.1.5 Csatolást törölni (TANGDEL)

Egy TANG(...) (Oldal 499)-gal definiált érintőleges csatolás az érintőleges csatolásnak TANGOF(...) (Oldal 503)-fal való kikapcsolása után is fennáll. A fennálló érintőleges csatolás megakadályozza pl. a következő funkciókat:

- síkváltás
- geometria-tengely átkapcsolás
- egy új érintőleges csatolás definiálása a követő tengelyre

Az előre definiált TANGDEL(...) eljárással az érintőleges csatolásnak a TANGOF(...)-fal való kikapcsolása után az érintőleges csatolás törölve lesz.

Szintaxis

```
TANGDEL(<követő tengely>)
```

Jelentés

TANGDEL(...):	egy TANG()-gal definiált érintőleges csatolás törlése	
	hatásosság:	mondatonként

<követő tengely>:	követő tengely tengely-neve, amelyiknek az érintőleges csatolását törölni kell	
	adattípus:	AXIS
	értéktartomány:	csatornanevek

Példák

Vezető tengely váltás

Mielőtt a követő tengelyre egy új érintőleges csatolást lehet definiálni egy másik vezető tengellyel, a fennálló érintőleges csatolást először törölni kell.

Programkód	Kommentár
N10 **TANG**(A, X, Y, 1)	; érintőleges csatolás a követő tengely A-ra definiálni A X-hez és Y-hoz
N20 TANGON(A)	; érintőleges csatolást a követő tengely A-ra bekapcsolni
N30 X10 Y20	
...	
N80 TANGOF(A)	; érintőleges csatolást a követő tengely A-ra kikapcsolni
N90 **TANGDEL**(A)	; érintőleges csatolást a követő tengely A-ra törölni
...	
N120 **TANG**(A, X, Z)	; új érintőleges csatolást a követő tengely A-ra definiálni
N130 TANGON(A)	; új érintőleges csatolást a követő tengely A-ra bekapcsolni
...	

geometria-tengely átkapcsolás

Mielőtt egy fennálló csatolásra egy geometria-tengely átkapcsolást végre lehet hajtani, a csatolást először törölni kell.

Programkód	Kommentár
N10 GEOAX(2, Y1)	; 2. geometria tengely = géptengely Y1
N20 TANG(A, X, Y)	; érintőleges csatolást a követő tengely A-ra definiálni
N30 TANGON(A, 90)	; érintőleges csatolást a követő tengely A-ra bekapcsolni
N40 G2 F8000 X0 Y0 I0 J50	; mozgás mondat
N50 TANGOF(A)	; érintőleges csatolást a követő tengely A-ra kikapcsolni
N60 **TANGDEL**(A)	; érintőleges csatolást a követő tengely A-ra törölni
N70 **GEOAX**(2, Y2)	; 2. geometria tengely = géptengely Y2
N80 TANG(A, X, Y)	; új érintőleges csatolást a követő tengely A-ra definiálni
N90 TANGON(A, 90)	; új érintőleges csatolást a követő tengely A-ra bekapcsolni
...	

12.2 Előtolás lefutás (FNORM, FLIN, FCUB, FPO)

Az előtolás lefutásának rugalmas megadására az előtolás DIN 66025 szerinti programozása ki lett bővítve a lineáris és a köbös lefutással.

A köbös lefutást lehet közvetlenül interpoláló Spline-ként programozni. Ezáltal lehetséges - a megmunkálandó munkadarab görbületeitől függően- folytonosan sima sebesség lefutást programozni.

Ezek a sebesség lefutások rándulásmentes gyorsulás változásokat és ezáltal egyenletesebb munkadarab felület készítését teszik lehetővé.

Szintaxis

```
F… FNORM
F… FLIN
F… FCUB
F=FPO(…,…,…)
```

Jelentés

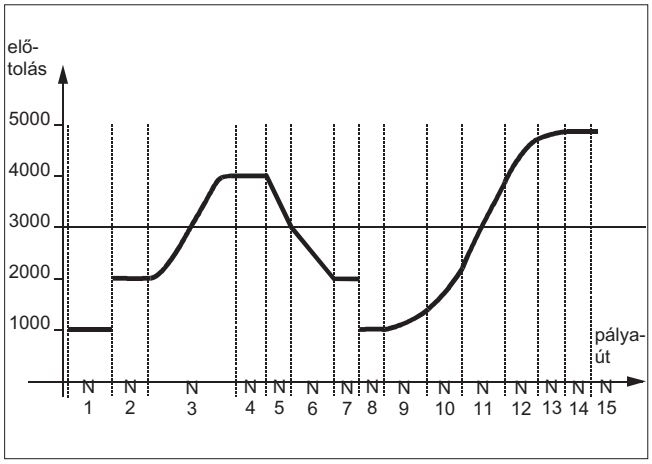
FNORM:	Alap-beállítás. Az előtolás értékét a mondat pályaútja adja meg és azután modális értékként hat.
FLIN:	Pálya sebesség-profil **lineáris:** Az előtolás értéket a mondat kezdeti aktuális értéktől a mondat végéig a pályaúton lineáris és azután modális értékként hat. Ezt az eljárást lehet G93 és G94-gyel kombinálni
FCUB:	Pálya sebesség-profil **köbös:** A mondatonként programozott F-értékek - a mondat végére vonatkoztatva - egy Spline-nal lesznek összekötve. A Spline az előző és a következő sebesség-megadásokhoz érintőlegesen csatlakozik és G93 és G94-gyel hat. Ha hiányzik egy mondatban az F-cím, akkor itt az utolsó programozott F-érték kerül alkalmazásra.
F=FPO… :	Pálya sebesség-profil **polinom:** Az F-cím az előtolás lefutását az aktuális értéktől a mondat végéig egy polinommal írja le. A vég-érték azután a modális érték.

Előtolást optimalizálás töredezett pályaszakaszokon

Az F=FPO előtolás-polinom és az FCUB előtolás-spline mindig CFC állandó vágósebességgel lesz megtéve. Ezáltal lehetséges egy egyenletes gyorsulású parancs sebesség-profil létrehozása.

Példa: Különféle sebesség-profilok

Ebben a példában a különféle sebesség-profilok grafikus ábrázolása és programozása látható.



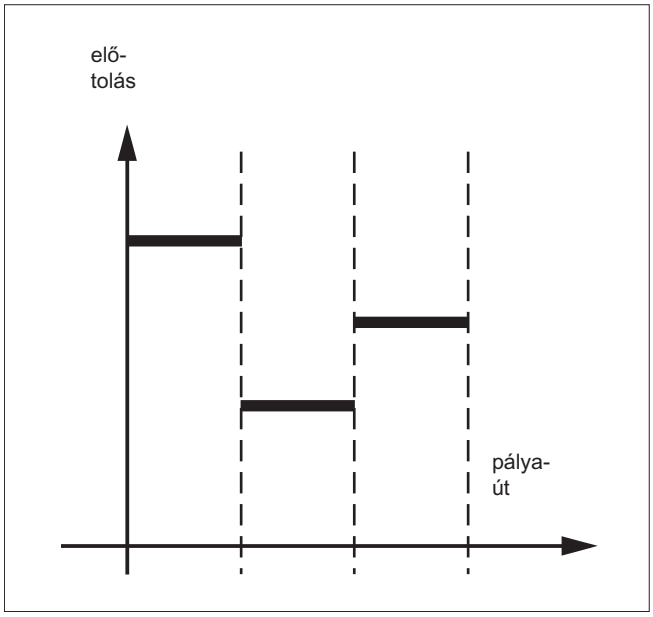
Programkód	Kommentár
N1 F1000 FNORM G1 X8 G91 G64	; Állandó előtolás-profil, láncméret megadás
N2 F2000 X7	; Ugrásszerű parancssebesség változás
N3 F=FPO(4000, 6000, -4000)	; Előtolás-profil polinommal 4000-es előtolással a mondat végén
N4 X6	; Polinom előtolás 4000 modális érték.
N5 F3000 FLIN X5	; Lineáris előtolás-profil
N6 F2000 X8	; Lineáris előtolás-profil
N7 X5	; Lineáris előtolás modális érték
N8 F1000 FNORM X5	; Állandó előtolás-profil ugrásszerű előtolás változással.
N9 F1400 FCUB X8	; Az összes következő mondatonként programozott F érték Spline-nal lesz összekötve.
N10 F2200 X6	
N11 F3900 X7	
N12 F4600 X7	
N13 F4900 X5	; Spline-profil kikapcsolása.
N14 FNORM X5	
N15 X20	

További információ

FNORM

Az F előtolás cím a pályaelőtolás értéket állandó értékként adja meg a DIN 66025 szerint.

Több információ erről az "Alapok" programozási kézikönyvben található.

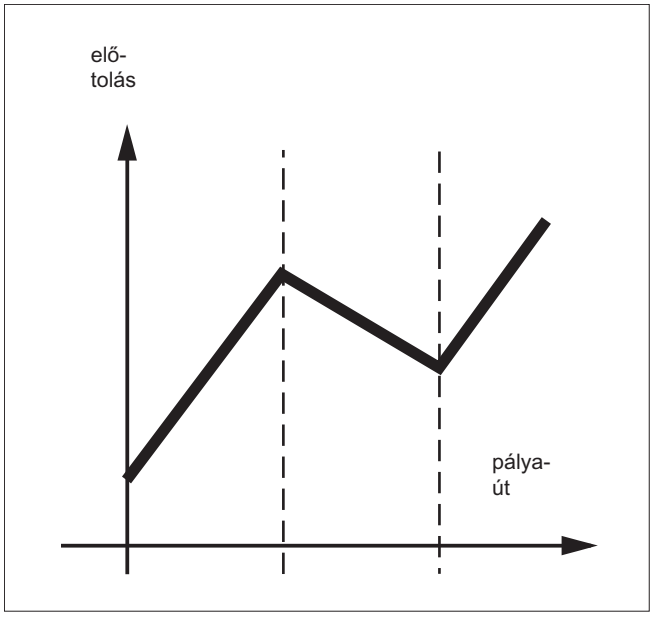


FLIN

Az előtolás lefutás az aktuális értéktől a programozott F-értékig lineárisan fut a mondat végéig.

Példa:

```
N30 F1400 FLIN X50
```

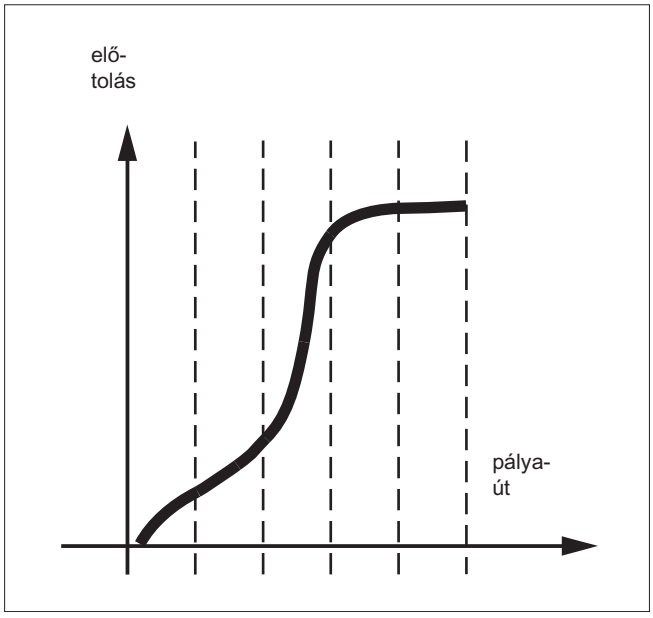


FCUB

Az előtolás lefutás az aktuális értéktől a programozott F-értékig köbös lefutásban változik a mondat végéig. A vezérlés az összes aktív FCUB-bal mondatonként programozott előtolás értéket egy Spline-nal köti össze Az előtolás értékek itt támpontokként szolgálnak a Spline interpoláció kiszámításához.

Példa:

```
N50 F1400 FCUB X50
N60 F2000 X47
N70 F3800 X52
```



F=FPO(...,...,...)

Az előtolás lefutás egy polinommal közvetlenül van programozva. A polinom együtthatóinak megadása a polinom interpolációval azonos.

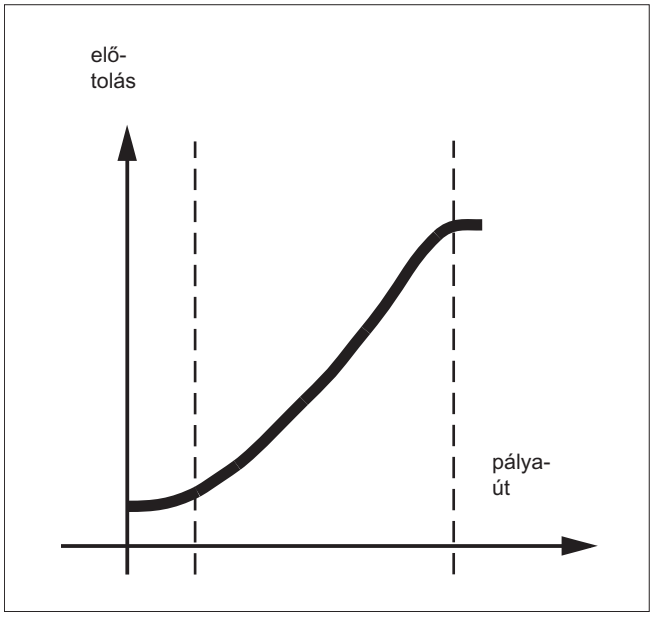
Példa:

```
F=FPO(endfeed, quadf, cubf)
```

endfeed, quadf és cubf előre definiált változók.

endfeed:	előtolás a mondatvégen
quadf:	négyzetes polinom együttható
cubf:	köbös polinom együttható

Aktív `FCUB`-nál a Spline a mondat elején és végén érintőlegesen kapcsolódik az `FPO`-val megadott lefutáshoz.



Peremfeltételek

- A programozott előtolás lefutástól függetlenül érvényesek a pálya-viselkedés programozás funkciók.
- A programozott előtolás lefutás alapvetően abszolút - függetlenül a `G90` vagy `G91`-től.
- Az `FLIN` és `FCUB` előtolás lefutás `G93`-cal és`G94`-gyel hat,`G95`, `G96/G961` és `G97/G971` esetén nem .
- Az aktív `COMPON` kompresszornál érvényes több mondat összefogásánál egy Spline-szegmensbe:

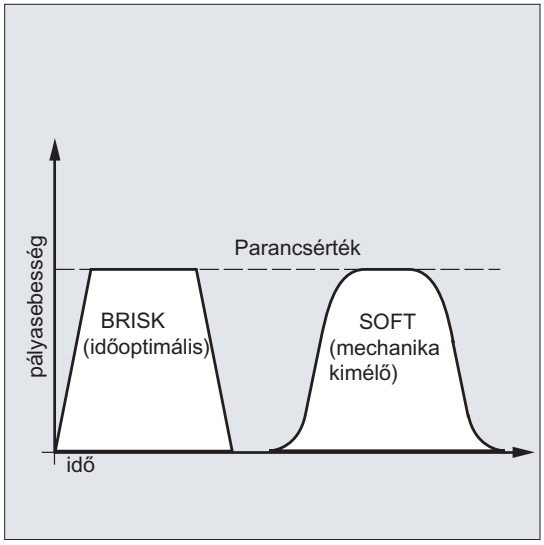
`FNORM`:	A Spline-szegmensre az utolsó hozzátartozó mondat F-szava érvényes.
`FLIN`:	A Spline-szegmensre az utolsó hozzátartozó mondat F-szava érvényes. A programozott F-érték a szegmens végére érvényes és akkor lineárisan lesz megtéve.
`FCUB`:	A létrehozott előtolás Spline maximum az MD20172 $MC_COMPRESS_VELO_TOL gépadatban megadott értékkel tér el a programozott végpontokon.
`F=FPO(…,…,…)`:	Ezek a mondatok nem lesznek komprimálva.

12.3 Gyorsulási viselkedés

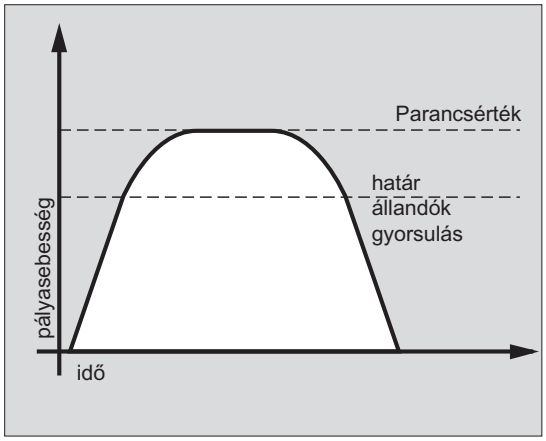
12.3.1 Gyorsulási modus (BRISK, BRISKA, SOFT, SOFTA, DRIVE, DRIVEA)

A gyorsulás modus programozásához a következő programutasítások állnak rendelkezésre:

- "BRISK, BRISKA"
  Az egyes tengelyek ill. pályatengelyek maximális gyorsítással mennek a programozott előtolási sebesség eléréséig.(**gyorsítás rándítás-határolás nélkül**).
- "SOFT, SOFTA"
  Az egyes tengelyek ill. pályatengelyek állandó gyorsítással mennek a programozott előtolási sebesség eléréséig.(**gyorsítás rándítás-határolással**).
- "DRIVE, DRIVEA"
  Az egyes tengelyek ill. pályatengelyek maximális gyorsítással mennek egy megadott sebességhatárig (gépadat beállítás!). Azután egy gyorsítás-csökkentés történik (gépadat beállítás!) a programozott előtolási sebesség eléréséig.



Kép 12-1 A pályasebesség lefutása `BRISK` és `SOFT` esetén



Kép 12-2 A pályasebesség lefutása DRIVE esetén

Szintaxis

```
BRISK
BRISKA(<tengely1>,<tengely2>,…)
SOFT
SOFTA(<tengely1>,<tengely2>,…)
DRIVE
DRIVEA(<tengely1>,<tengely2>,…)
```

Jelentés

BRISK:	utasítás a "Gyorsítás rándítás-határolás nélkül" bekapcsolására pályatengelyekre
BRISKA:	utasítás a "Gyorsítás rándítás-határolás nélkül" bekapcsolására egyes tengelyek mozgására (JOG, JOG/INC, pozícionáló tengely, ingázó tengely, stb.)
SOFT:	utasítás a "Gyorsítás rándítás-határolással" bekapcsolására pályatengelyekre
SOFTA:	utasítás a "Gyorsítás rándítás-határolással" bekapcsolására egyes tengelyek mozgására (JOG, JOG/INC, pozícionáló tengely, ingázó tengely, stb.)
DRIVE:	utasítás a csökkentett gyorsítás bekapcsolására egy megadott sebességhatáron (MD35220 $MA_ACCEL_REDUCTI-ON_SPEED_POINT) felül pályatengelyekre
DRIVEA:	utasítás a csökkentett gyorsítás bekapcsolására egy megadott sebességhatáron (MD35220 $MA_ACCEL_REDUCTI-ON_SPEED_POINT) felül egyes tengelyek mozgására (JOG, JOG/INC, pozícionáló tengely, ingázó tengely, stb.).
(<tengely1>,<tengely2>,…):	egyes tengelyek, amelyekre a felhívott gyorsítás modus érvényes kell legyen

Peremfeltételek

Gyorsítás modus váltása a megmunkálás közben

Ha egy munkadarabprogramban a gyorsítás modus a megmunkálás közben lesz váltva (BRISK ↔ SOFT), akkor az átmenetnél pályavezérlő-üzemben is egy mondatváltás történik pontos-állj-jal a mondat végén.

Példák

Példa 1: SOFT és BRISKA

Programkód

```
N10 G1 X… Y… F900 SOFT
N20 BRISKA(AX5,AX6)
...
```

Példa 2: DRIVE és DRIVEA

Programkód
N05 DRIVE
N10 G1 X… Y… F1000
N20 DRIVEA (AX4, AX6)
...

Irodalom

Alapfunkciók működési kézikönyv; Gyorsulás (B2)

12.3.2 Követő-tengelyek gyorsulásának befolyásolása (VELOLIMA, ACCLIMA, JERKLIMA)

A tengelycsatolásoknál (érintőleges utánvezetés, magával húzás, vezetőérték-csatolás elektronikus hajtómű: lásd "Tengely-csatolások (Oldal 545)") a követő-tengelyek/orsók egy vagy több vezető-tengelytől/-orsótól függően fognak mozogni.

A követő-tengelyek dinamikájának korlátozását a VELOLIMA, ACCLIMA és JERKLIMA funkciókkal befolyásolni lehet munkadarabprogramból vagy szinkronakciókból, már aktív tengely-csatolásnál is.

Megjegyzés

A JERKLIMA funkció nem áll rendelkezésre az összes csatolásmódra.

Irodalom:

- Különleges funkciók működési kézikönyv; Tengelycsatolások (M3)
- Bővítő funkciók működési kézikönyv; Szinkronakciók (S3)

Megjegyzés

Rendelkezésre állás SINUMERIK 828D esetén

A SINUMERIK 828D esetén a VELOLIMA, ACCLIMA és JERKLIMA funkciókat csak a "Magával húz" funkcióval lehet alkalmazni!

Szintaxis

```
VELOLIMA(<tengely>)=<érték>
ACCLIMA(<tengely>)=<érték>
JERKLIMA(<tengely>)=<érték>
```

Jelentés

VELOLIMA:	utasítás korrekcióra a paraméterezett maximális **sebességnél**
ACCLIMA:	utasítás korrekcióra a paraméterezett maximális **gyorsulásnál**

JERKLIMA:	utasítás korrekcióra a paraméterezett maximális **rándításnál**
<tengely>:	követő-tengely, amelynek a dinamika-határolását korrigálni kell
<érték>:	százalékos korrekció-érték

Példák

Példa 1: Dinamika-határolás korrekció egy követő-tengelyre (AX4)

```
Programkód                  Kommentár
...
VELOLIMA[AX4]=75            ; Határolás korrekció a gépadatban megadott tengely maximá-
                            lis sebesség 75 %-ára.
ACCLIMA[AX4]=50             ; Határolás korrekció a gépadatban megadott tengely maximá-
                            lis gyorsulás 50%-ára.
JERKLIMA[AX4]=50            ; Határolás korrekció a gépadatban megadott tengely maximá-
                            lis rándítás 50%-ára pályamozgásnál.
...
```

Példa 2: Elektronikus hajtómű

A tengely 4 egy "Elektronikus hajtómű" csatolással lesz a tengely X-hez csatolva. A követő-tengely gyorsulási képessége a maximális gyorsulás 70%-ára lesz korlátozva. A maximális megengedett sebesség a maximális sebesség 50%-ára lesz korlátozva. A csatolás bekapcsolása után a maximális megengedett sebesség ismét 100%-ra lesz beállítva.

```
Programkód                          Kommentár
...
N120 ACCLIMA[AX4]=70                ; Csökkentett maximális gyorsulás.
N130 VELOLIMA[AX4]=50               ; Csökkentett maximális sebesség.
...
N150 EGON(AX4,"FINE",X,1,2)         ; EH-csatolás bekapcsolása.
...
N200 VELOLIMA[AX4]=100              ; Teljes maximális sebesség.
...
```

Példa 3: Vezető-érték csatolás befolyásolására statikus szinkronakcióval

A tengely 4 egy vezető-érték csatolással lesz a tengely X-hez csatolva. A gyorsulási viselkedés a statikus szinkronakció 2-vel a 100-as pozíciótól 80 %-ra lesz korlátozva.

```
Programkód                                      Kommentár
...
N120 IDS=2 WHENEVER $AA_IM[AX4] >               ; szinkron-akció
100 DO ACCLIMA[AX4]=80
N130 LEADON(AX4, X, 2)                          ; vezetőérték csatolás be
...
```

12.3.3 Technológia-specifikus dinamika-értékek aktiválása (DYNNORM, DYNPOS, DYNROUGH, DYNSEMIFIN, DYNFINISH)

A "Technológia" G-csoporttal 5 különféle technológiai megmunkálás-lépésre a hozzá illó dinamikát lehet aktiválni.

A dinamika-értékek és a G-kódok beállíthatóak és ezzel a gépadat beállításoktól függőek (→ gépgyártó!)

Irodalom:
Alapfunkciók működési kézikönyv; Pályavezérlő üzem, Pontos-állj, LookAhead (B1)

Szintaxis

Dinamika-értéket aktiválni:

```
DYNNORM
DYNPOS
DYNROUGH
DYNSEMIFIN
DYNFINISH
```

Megjegyzés

A dinamika értékek már abban a mondatban hatásosak lesznek, amelyikben a hozzátartozó G-kód programozva lesz. Nem történik megmunkálási állj.

Megadott mezőelemet írni vagy olvasni:

```
R<m>=$MA...[n,X]
$MA...[n,X]=<érték>
```

Jelentés

DYNNORM:	G-utasítás a **normál dinamika** aktiválására	
DYNPOS:	G-utasítás a **pozícionáló üzem**, **menetfúrás** dinamika aktiválására	
DYNROUGH:	G-utasítás a **nagyolás** dinamika aktiválására	
DYNSEMIFIN:	G-utasítás a **simítás** dinamika aktiválására	
DYNFINISH:	G-utasítás a **finomsimítás** dinamika aktiválására	
R<m>:	számítási paraméter <m> számmal	
$MA...[n,X]:	gépadat dinamikát meghatározó mezőelemmel	
<n>:	mező-index	
	értéktartomány:	0 ... 4
	0	normál dinamika (DYNNORM)
	1	dinamika pozicionáló üzemhez (DYNPOS)
	2	dinamika nagyoláshoz (DYNROUGH)
	3	dinamika simításhoz (DYNSEMIFIN)
	4	dinamika finom simításhoz (DYNFINISH)

<X> :	tengelycím
<érték>:	dinamika-érték

Példák

Példa 1: Dinamika-értéket aktiválni

Programkód	Kommentár
DYNNORM G1 X10	; alaphelyzet
DYNPOS G1 X10 Y20 Z30 F…	; pozicionáló üzem, menetfúrás
DYNROUGH G1 X10 Y20 Z30 F10000	; nagyolás
DYNSEMIFIN G1 X10 Y20 Z30 F2000	; simítás
DYNFINISH G1 X10 Y20 Z30 F1000	; finom-simítás

Példa 2: Megadott mezőelemet írni vagy olvasni

Maximális gyorsítás nagyoláshoz, X tengely

Programkód	Kommentár
R1=$MA_MAX_AX_ACCEL[2,X]	; olvasás
$MA_MAX_AX_ACCEL[2,X]=5	; írás

12.4 Mozgás elővezérléssel (FFWON, FFWOF)

Az elővezérlés által pályamenetnél a sebességfüggő utánfutás-út nullára csökken le. Elővezérléses mozgás nagyobb pályapontosságot és ezzel jobb gyártási eredményeket tesz lehetővé.

Szintaxis

```
FFWON
FFWOF
```

Jelentés

FFWON:	utasítás az elővezérlés **be**kapcsolására
FFWOF:	utasítás az elővezérlés **ki**kapcsolására

Megjegyzés

A gépadatokkal meghatározásra kerül az elővezérlés jellege és az, hogy melyik pályatengelyt kell elővezérelten mozgatni.

Alap-beállítás: sebességfüggő elővezérlés

Opció: gyorsulásfüggő elővezérlés

Példa

Programkód

```
N10 FFWON
N20 G1 X… Y… F900 SOFT
```

12.5 Programozható kontúr-pontosság (CPRECON, CPRECOF)

A "Programozható kontúr-pontosság" funkció csökkenti a pályahibát az erősen görbült kontúrokon a sebesség illesztésével.

A betartandó kontúr-pontosság a gép beállításától (MD20470 $MC_MC_CPREC_WITH_FFW; lásd a gépgyártó tájékoztatásait) függően vagy a $SC_CONTPREC beállítási adattal vagy a CTOL programozott kontúrtűréssel lesz megadva. Minél kisebb az érték és minél kisebb a geometria-tengelyek K_V-tényezője, annál erősebben lesz lecsökkentve a pályaelőtolás a görbült kontúrokon.

A "Programozható kontúr-pontosság " funkció a `CPRECON` és `CPRECOF` utasításokkal az NC programban be- ill. kikapcsolható.

Szintaxis

```
CPRECON
...
CPRECOF
```

Jelentés

`CPRECON`:	G utasítás felhívás: "Programozható kontúr-pontosság" **be**kapcsolás	
	hatásosság:	modális
`CPRECOF`:	G utasítás felhívás: "Programozható kontúr-pontosság" **be**kapcsolás	
	hatásosság:	modális

CPRECON és CPRECOF együtt képezik a 39-es G funkció-csoportot (Programozható kontúr-pontosság).

Megjegyzés

A $SC_MINFEED (minimum előtolás CPRECON-nál) beállítási adattal a felhasználó megadhat egy minimum előtolást a pálya-előtolásra.

Az előtolás nem lesz ezen érték alá korlátozva, kivéve, ha alacsonyabb F érték lett korlátozva vagy a tengelyek dinamikus korlátozásai alacsonyabb pályasebességet kényszerítenek ki.

Megjegyzés

A "Programozható kontúr-pontosság" funkció csak a pálya geometria-tengelyeit nézi. A pozícionáló tengelyek sebességére nincs hatása.

Példa

Programkód	**Kommentár**
`N10 G0 X0 Y0`	
`N20 CPRECON`	`; "Programozható kontúr-pontosság" bekapcsolása.`
`N30 G1 G64 X100 F10000`	`; Megmunkálás 10 m/perc-cel pályavezérlő-üzemben.`

Programkód	Kommentár
N40 G3 Y20 J10	; Automatikus előtolás-behatárolás a körmondatban.
N50 G1 X0	; Előtolás ismét határolás nélkül (10 m/perc).
...	
N100 CPRECOF	; "Programozható kontúr-pontosság" kikapcsolása.
N110 G0 ...	

Irodalom

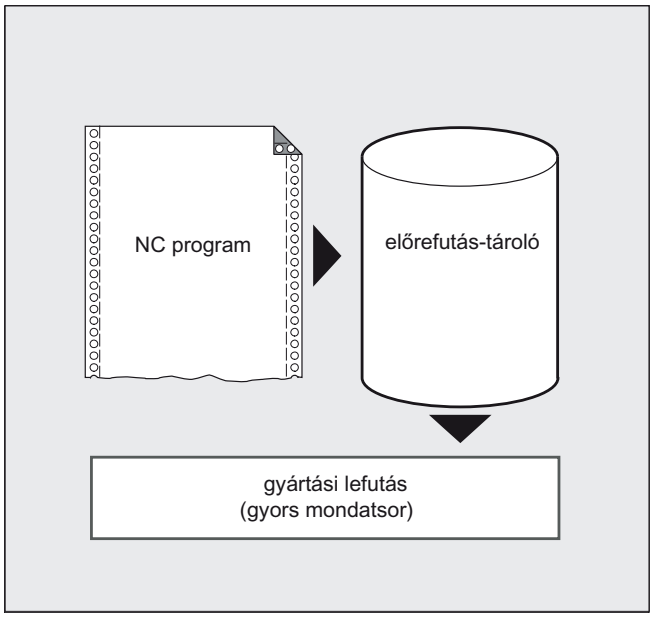
A CTOL programozásához lásd" Kontúr-/tájolás-tűrés programozása (CTOL, OTOL, ATOL) (Oldal 540)"

További információk a "Programozható kontúr-pontosság" funkcióhoz, lásd:

Különleges funkciók működési kézikönyv; kontúr-csatorna felügyelet (K6), fejezet: "Programozható kontúr-pontosság"

12.6 Programlefutás előrefutás tárolóval (STARTFIFO, STOPFIFO, STOPRE)

A vezérlésnek a kiépítettségétől függően van egy megadott méretű un. előrefutás tárolója, amelyik az előkészített mondatokat a feldolgozás előtt tárolja és a feldolgozás során gyorsan sorban kiadja. Ezáltal lehetséges rövid utakat nagy sebességgel megtenni. Amennyire a vezérlés maradékideje ezt lehetővé teszi, tölti az előrefutás tárolót.



Megmunkálási szakasz megjelölése

A megmunkálási szakaszt, amelyet az előrefutás tárolóban tárolni akarunk, az elején a "STOPFIFO"-val és a végén a "STARTFIFO"-val jelöljük meg. Az előkészített és közben tárolt mondatok feldolgozása csak "STARTFIFO" utasítás után kezdődik vagy ha az előrefutás tároló megtelt.

Automatikus előrefutás tároló vezérlés

Az automatikus előrefutás tároló vezérlést a "FIFOCTRL" utasítás hívja fel. A "FIFOCTRL" először ugyanúgy hat, mint a "STOPFIFO". Minden programozásnál addig lesz várva, amíg az előrefutás tároló megtelik, utána kezdődik a feldolgozás. Eltérő a viselkedés az előrefutás tároló kiürülésénél: a "FIFOCTRL"-lal a 2/3-os töltöttségi szintnél a pályasebesség csökkentve lesz a teljes kiürülés és a megállásig lefékezés elkerülésére.

Előrefutás-állj

A mondat-előkészítés és a közbenső tárolás meg lesz állítva. ha a mondatban a "STOPRE" programozva van. A következő mondat csak akkor lesz végrehajtva, ha az összes eddig

előkészített és tárolt mondat teljes feldolgozása befejeződik. Az előző mondat pontos állj-jal (mint G9) lesz megállítva.

FIGYELEM
Program megszakítás Bekapcsolt szerszámkorrekciónál és Spline-interpolációnál nem kellene "STOPRE"-t programozni, mert összetartozó mondat-blokkok megszakításra kerülhetnek.

Szintaxis

Táblázat 12-1 Megmunkálási szakasz megjelölése:

```
STOPFIFO
...
STARTFIFO
```

Táblázat 12-2 Automatikus előrefutás tároló vezérlés:

```
...
FIFOCTRL
...
```

Táblázat 12-3 Előrefutás-állj:

```
...
STOPRE
...
```

Megjegyzés

A "STOPFIFO", "STARTFIFO", "FIFOCTRL" és "STOPRE" utasításokat külön mondatban kell programozni.

Jelentés

STOPFIFO:	A "STOPFIFO" jelöli egy megmunkálási szakasz elejét, amit a közbenső tárolóban kell tárolni. A "STOPFIFO" a megmunkálás meg lesz állítva és az előrefutás tároló lesz töltve, amíg: • "STARTFIFO" vagy "STOPRE" lesz felismerve vagy • az előrefutás tároló tele lesz vagy • a program vége el lesz érve.
STARTFIFO:	A "STARTFIFO" indítja a megmunkálási szakasz gyors feldolgozását, ezzel párhuzamosan történik az előrefutás tároló feltöltése
FIFOCTRL:	az automatikus előrefutás tároló vezérlés bekapcsolása
STOPRE:	előrefutást megállítani

Megjegyzés

Az előrefutás tároló feltöltése nem lesz végrehajtva ill. meg lesz szakítva, ha a megmunkálási szakasz olyan utasításokat tartalmaz, amelyek egy pufferelés nélküli működést kényszerítenek (referencia pontra menet, mérési funkciók, ...)

Megjegyzés

A gép állapotadataihoz ($A…) hozzáférésnél a vezérlés egy belső előrefutás-álljt hoz létre.

Példa: előrefutást megállítani

Programkód	Kommentár
...	
N30 MEAW=1 G1 F1000 X100 Y100 Z50	; mérési mondat az első mérőbemenet mérőtapintójával és egyenes-interpoláció
N40 STOPRE	; előrefutás-állj
...	

12.7 Stop-Delay tartományt definiálni (DELAYFSTON, DELAYFSTOF)

Egy feltételesen megszakítható tartomány definiálására a munkadarabprogramban (Stop-Delay tartomány) a DELAYFSTON és DELAYFSTOF előre definiált eljárások szolgálnak.

Megjegyzés

Szinkronakciókban a DELAYFSTON és a DELAYFSTOF **nem** megengedett!

Szintaxis

```
DELAYFSTON
...
DELAYFSTOF
```

Jelentés

DELAYFSTON:	Egy Stopp-Delay tartomány elejét definiálni	
	egyedül a mondatban:	igen
DELAYFSTOF:	egy Stop-Delay tartomány végének definiálása	
	egyedül a mondatban:	igen

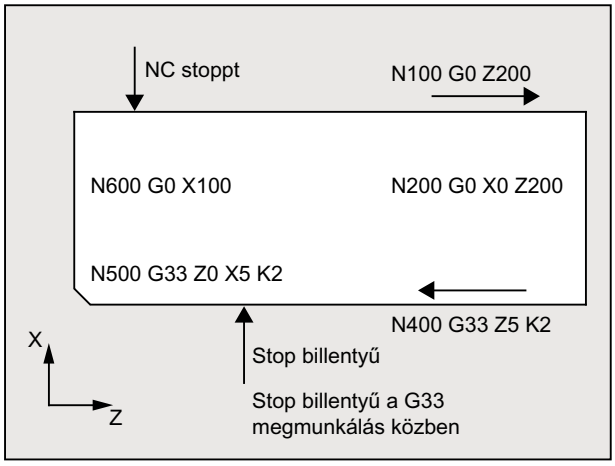
Program példa

Egy hurokban a következő programblokk lesz ismételve:

Programkód

```
...
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
...
```

A következő képen megállapítható, hogy a felhasználó a Stop-Delay tartományban "Stop"-ot nyom, és az NC a fékezési folyamatot a Stop-Delay tartományon kívül kezdi, azaz az N100 mondatban. Ezzel az NC az N100 előtti tartományban fog megállni.



További információk

Alprogramvég

Az alprogram végén, amelyikben a DELAYFSTON fel lett hívva, implicit DELAYFSTOF lesz aktiválva.

Skatulyázási mélység

Ha az alprogram 1 egy Stop-Delay tartományban az alprogram 2-t felhívja, akkor az alprogram 2 teljesen egy Stop-Delay tartományban van. Különben a DELAYFSTOF az alprogram 2-ben hatástalan.

Példa:

Programkód	Kommentár
N10010 **DELAYFSTON**	; mondatok N10xxx-szel programszint 1
N10020 R1 = R1 + 1	
N10030 G4 F1	; Stop-Delay tartomány kezdődik
...	
N10040 Unterprogramm2	
...	
...	; Unterprogramm 2 értelmezése
N20010 **DELAYFSTON**	; hatástalan, ismételt kezdet, 2. szint
...	
N20020 **DELAYFSTOF**	; hatástalan, vége a másik szinten
N20030 RET	
N10050 **DELAYFSTOF**	; Stop-Delay tartomány vége az azonos szinten
...	
N10060 R2 = R2 + 2	
N10070 G4 F1	; Stop-Delay tartomány vége Stop-ok mostantól közvetlenül hatnak

Rendszerváltozók

Azt, hogy a munkadarabprogram feldolgozása az aktuális időpontban egy Stop-Delay tartományban van-e, a következő rendszerváltozókkal lehet lekérdezni:

- munkadarab-programban $P_DELAYFST-vel
- szinkronakciókban $AC_DELAYFST-tel

Érték	Jelentés
0	Stop-Delay tartomány nem aktív
1	Stop-Delay tartomány aktív

12.8 Programhely tiltása SERUPRO számára (IPTRLOCK, IPTRUNLOCK)

A gép bizonyos komplikált mechanikai állapotaiban szükséges a SERUPRO mondatkeresést tiltani.

Egy programozható megszakítás-mutatóval van egy beavatkozási lehetőség a "Keresés megszakítás-helyre" esetében a keresésre alkalmatlan hely előttre mutatni.

Keresésre alkalmatlan tartományok is lehetnek definiálva a munkadarabprogram tartományokban, amelyekbe az NC nem tud újra belépni. A program megszakításával az NC megjegyzi az utoljára feldolgozott mondatot, amelyiket a HMI kezelőfelületen keresni lehet.

Szintaxis

```
IPTRLOCK
IPTRUNLOCK
```

Az utasítások egyedül állnak egy programsorban és lehetővé tesznek egy programozható megszakítás-mutatót

Jelentés

IPTRLOCK:	keresésre alkalmatlan programszakasz kezdete
IPTRUNLOCK:	keresésre alkalmatlan programszakasz vége

Mindkét utasítás csak munkadarabprogramokban használható, szinkron-akciókban azonban **nem**.

Példa

Keresésre alkalmatlan programszakaszok egymásba skatulyázása két programszinten implicit "IPTRUNLOCK"-kal. Az implicit "IPTRUNLOCK" az Unterprogramm 1-ben lezárja a keresésre alkalmatlan tartományt.

Programkód	**Kommentár**
N10010 IPTRLOCK()	
N10020 R1 = R1 + 1	
N10030 G4 F1	; Megállás mondat, a keresésre alkalmatlan program-szakasz kezdete.
...	
N10040 Unterprogramm2	
...	; Unterprogramm2 értelmezése
N20010 IPTRLOCK ()	; Hatástalan, ismételt kezdet.
...	
N20020 IPTRUNLOCK ()	; Hatástalan, vége a másik szinten.
N20030 RET	
...	
N10060 R2 = R2 + 2	
N10070 RET	; A keresésre alkalmatlan programszakasz vége.
N100 G4 F2	; A főprogram folytatva lesz.

Egy megszakítás 100-ra ismét a megszakítás-mutatót adja.

További információ

Keresésre alkalmatlan tartományok felismerése és keresése

A keresésre alkalmatlan programszakaszokat a "IPTRLOCK" és "IPTRUNLOCK" utasítások jelölik.

Az "IPTRLOCK" utasítás befagyasztja a megszakítás-mutatót egy, a főfutásban végrehajtható egyes mondatra (SBL1). A következőkben ez a mondat lesz megállási mondatként megjelölve. Ha a "IPTRLOCK" után egy programmegszakítás lép fel, akkor a HMI kezelőfelületen ez után az úgynevezett megállási mondat után lehet keresni.

Az aktuális mondatra ismét rámenni

A megszakítás-mutatót az "IPTRUNLOCK" a következő programszakaszra beállítja az aktuális mondatra.

Egy megtalált keresőcél után ugyanazzal a megállási mondattal meg lehet ismételni egy új keresőcélt.

Egy, a felhasználó által szerkesztett megszakítás-mutatót a HMI-vel ismét el kell távolítani.

Szabály egymásba skatulyázásnál

A következő pontok szabályozzák a "IPTRLOCK" és "IPTRUNLOCK" utasítások együttműködését az egymásba skatulyázásokkal és az alprogramvégekkel:

1. Az alprogram végén, amelyikben a "IPTRLOCK" fel lett hívva, implicit "IPTRUNLOCK" lesz aktiválva.
2. Az "IPTRLOCK" egy keresésre alkalmatlan tartományban hatástalan.
3. Ha az Unterprogramm1 egy keresésre alkalmatlan tartományban felhívja az Unterprogramm2-t, akkor az Unterprogramm2 teljesen keresésre alkalmatlan marad. Az "IPTRUNLOCK" az Unterprogramm2-ben hatástalan.

További információk ehhez, lásd
/FB1/ Funktionshandbuch Grundfunktionen; BAG, Kanal, Programmbetrieb (K1).

Rendszerváltozó

Egy keresésre alkalmatlan tartományt a "$P_IPTRLOCK" által lehet a munkadarabprogramban felismerni.

Automatikus megszakítás-mutató

Az automatikus megszakítás-mutató funkció automatikusan keresésre alkalmatlanná minősít egy korábban megadott csatolási módot. Gépadattal

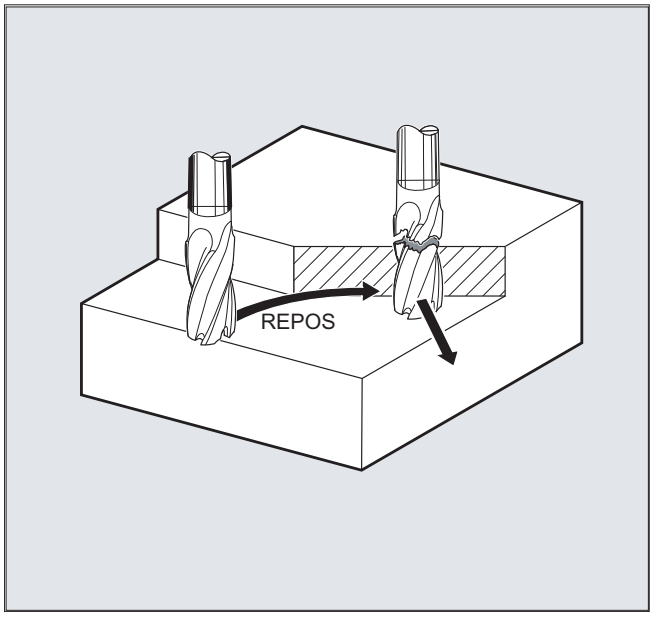
- elektronikus hajtómű "EGON"-nál
- tengely vezetőérték-csatolás "LEADON"-nál

az automatikus megszakítás-mutató aktiválva lesz. Ha a programozott és a gépadattal aktiválható automatikus megszakítás-mutatók átfedik egymást, akkor a lehető legnagyobb keresésre alkalmatlan tartomány lesz képezve.

12.9 Újra-rámenet a kontúrra (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL)

Ha a megmunkálás közben a futó programot megszakítjuk és a szerszámot szabadba hozzuk – pl. szerszámtörés miatt vagy a munkadarab megméréséhez – a kontúrra egy választható ponton programvezérelten újra rá tudunk menni.

A `REPOS` utasítás egy ASUP-ban úgy hat, mint egy alprogram visszaugrás (pl. M17). A következő mondatok már nem lesznek végrehajtva. A programfutás megszakításához lásd "Interrupt rutinok (ASUP) (Oldal 125)".



Szintaxis

```
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=…
```

Jelentés

Rámeneti út kiválasztása

REPOSA:	Újra rámenet a kontúrra a geometria-tengelyekkel egy egyenesen. Az összes többi csatorna-tengely szintén újra pozicionálva lesz.
REPOSL:	Újra rámenet a kontúrra a geometria-tengelyekkel egy egyenesen. Minden más csatorna-tengelyt külön programozni kell.
REPOSQ DISR=… :	Újra rámenet a kontúrra a geometria-tengelyekkel egy negyedkörön DISR sugárral. Minden más csatorna-tengelyt külön programozni kell.
REPOSQA DISR=… :	Újra rámenet a kontúrra a geometria-tengelyekkel egy negyedkörön DISR sugárral. Az összes többi csatorna-tengely szintén újra pozicionálva lesz.
REPOSH DISR=… :	Újra rámenet a kontúrra a geometria-tengelyekkel egy félkörön DISR sugárral. Minden más csatorna-tengelyt külön programozni kell.
REPOSHA DISR=… :	Újra rámenet a kontúrra a geometria-tengelyekkel egy félkörön DISR sugárral. Az összes többi csatorna-tengely szintén újra pozicionálva lesz.

Újra rámeneti pont kiválasztása

RMIBL:	megszakításpontra rámenni
RMIBL DISPR=…:	belépési pont DISPR távolságra mm/hüv.-ben a megszakításpont előtt
RMBBL:	mondat kezdőpontra rámenni
RMEBL:	mondat végpontra rámenni
RMEBL DISPR=… :	mondat végpontra rámenni DISPR távolsággal a végpont előtt
RMNBL:	a legközelebbi pályapontra rámenni
A0 B0 C0 :	tengelyek, amelyekben a rámenet történik

Megjegyzés

Kompatibilitás

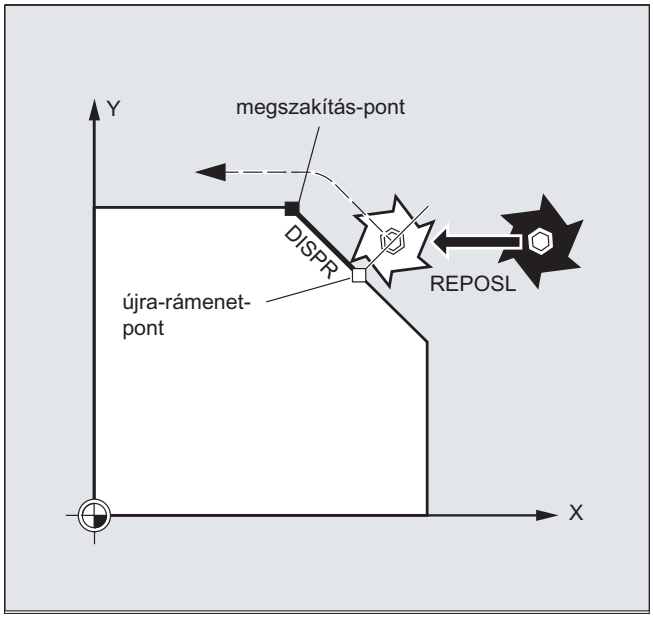
A régebbi szoftver kiadásokkal kompatibilitáshoz a REPOS rámeneti módust továbbra is lehet programozni a modális RMI, RMB, RME és RMN utasításokkal. Egy ASUP-on belüli alkalmazásnál ez el kell legyen látva a SAVE jellemzővel a PROC utasításban. Különben az ASUP-ban alkalmazott modális REPOS rámenet módus, ha az eltér az RMI elő-beállítástól, a következő REPOS eljárásoknál is hat.

Újra rámenet a kontúrra egy egyenesen, REPOSA, REPOSL

A szerszám az újra rámeneti pontra közvetlenül egy egyenesen megy rá.

Példa

```
REPOSL RMIBL DISPR=6 F400
```

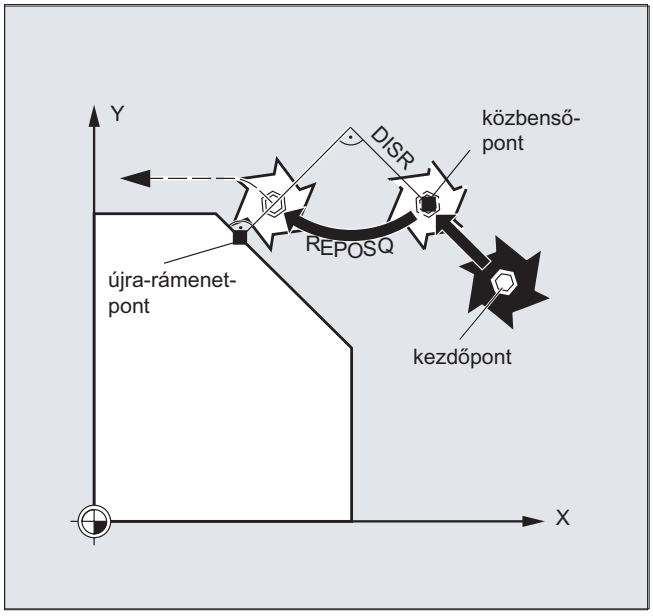


Újra rámenet a kontúrra negyedkörben, REPOSQ, REPOSQA

A szerszám az újra-rámeneti pontra egy negyedkörön `DISR=...` sugárral megy rá. A kezdő és az újra-rámeneti pont között szükséges közbenső pontot a vezérlés automatikusan számítja ki.

Példa

```
REPOSQ RMIBL DISR=10 F400
```

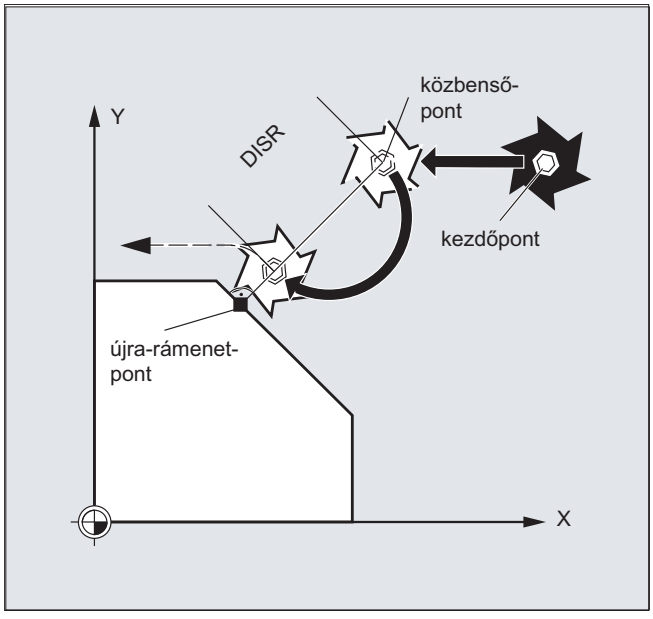


Újra rámenet a kontúrra félkörben, REPOSH, REPOSHA

A szerszám az újra-rámeneti pontra egy félkörön `DISR=...` átmérővel megy rá. A kezdő és az újra-rámeneti pont között szükséges közbenső pontot a vezérlés automatikusan számítja ki.

Példa

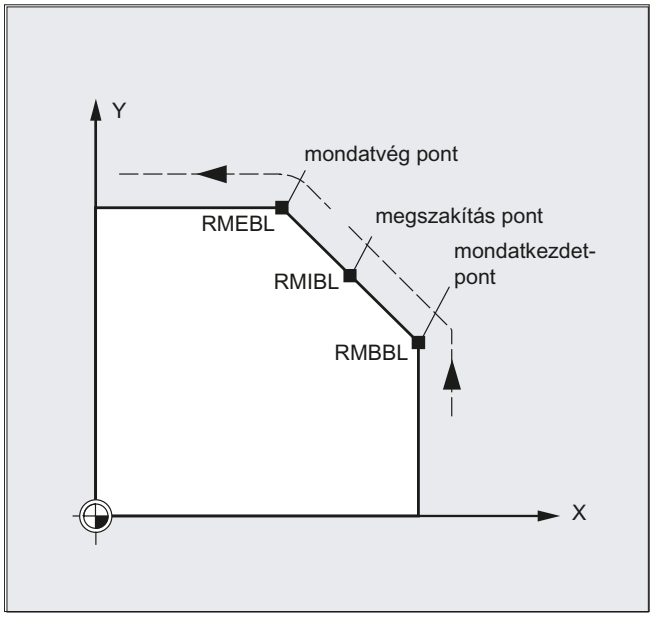
```
REPOSH RMIBL DISR=20 F400
```



Újra-rámeneti pont megadása (nem SERUPRO rámenethez RMNBL-lel)

Az NC mondatra vonatkoztatva, amelyben a program futás meg lett szakítva, három újra-rámeneti pont között lehet választani:

- RMIBL, megszakítás pont
- RMBBL, mondat kezdőpont ill. utolsó végpont
- RMEBL, mondat végpont



Az RMIBL DISPR=... ill. az RMEBL DISPR=... alkalmazásával egy a megszakítás-pont ill. a mondat végpont előtti újra-rámeneti pontot lehet megadni.

A DISPR=... leírja a kontúrútat mm/hüv.-ben, amennyivel az újra-rámeneti pont a megszakításpont ill. a mondat végpont előtt fekszik. Ez a pont - nagyobb értékek esetén is - maximum a mondat kezdőpontban lehet.

Ha nincs DISPR=... programozva, DISPR=0 érvényes és ez a megszakításpont (RMIBL-nél) ill. a mondat végpont (RMEBL-nél).

DISPR előjele

A DISPR előjele ki lesz értékelve. Pozitív előjelnél a viselkedés olyan, mint eddig.

Negatív előjelnél a megszakításpont mögött ill. RMBBL-nél a kezdőpont mögött lesz a rámenet.

A megszakításpont-rámenetpont közötti távolság a DISPR értékéből adódik. Ez a pont nagyobb értékek esetén is maximum a mondat kezdőpontban lehet.

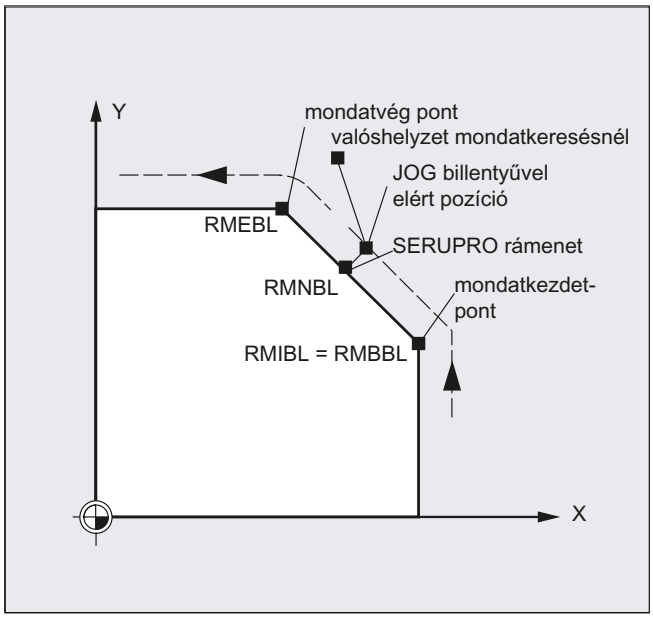
Alkalmazási példa:

Egy szenzor érzékeli a közelítést egy rögzítő pofához Egy ASUP lesz elindítva, amivel megkerüljük a rögzítő pofát.

Ezután egy negatív DISPR-rel egy, a rögzítő pofa mögötti pontra történik az újra-pozicionálás és a program folytatódik.

SERUPRO rámenet RMNBL-lel

Ha a megmunkálás során tetszőleges helyen egy megszakítás lesz kiváltva, akkor SERUPRO rámenetnél RMNBL-lel a megszakítási helytől a legrövidebb út lesz megtéve, hogy utána csak a maradékutat kelljen megmunkálni. Ehhez a felhasználó elindít egy SERUPRO eljárást a megszakítási mondatra és a JOG-billentyűkkel pozícionál a károsodott hely elé a célmondatban.



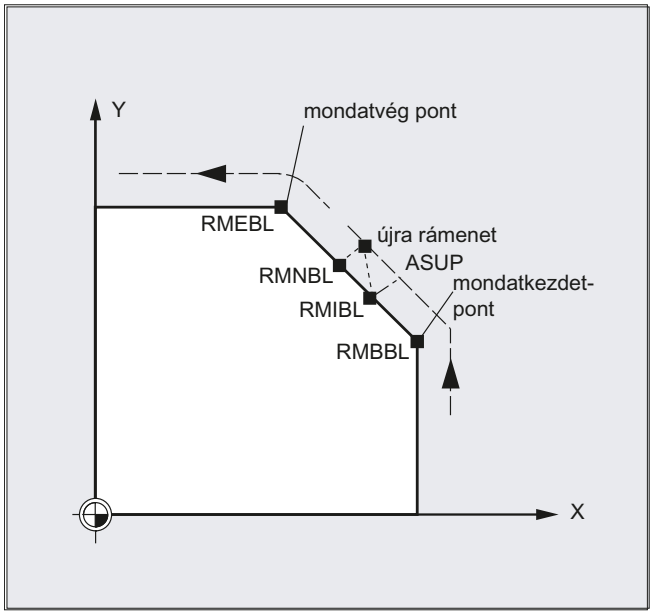
Megjegyzés

SERUPRO

A SERUPRO számára az RMIBL és RMBBL azonos. RMNBL nincs a SERUPRO-ra korlátozva, hanem általánosan érvényes.

Rámenet a legközelebbi pályapontól RMNBL

A REPOSA interpoláció-időpontjában egy megszakítás után az újra-rámeneti mondat RMNBL-lel nem lesz teljesen újrakezdve, hanem csak a maradékút lesz feldolgozva. A rámenet a megszakított mondat legközelebbi pályapontjára történik.



Az érvényes REPOS módus állapota

A megszakított mondat érvényes REPOS módusa olvasható szinkronakciókkal a $AC_REPOS_PATH_MODE változóból:

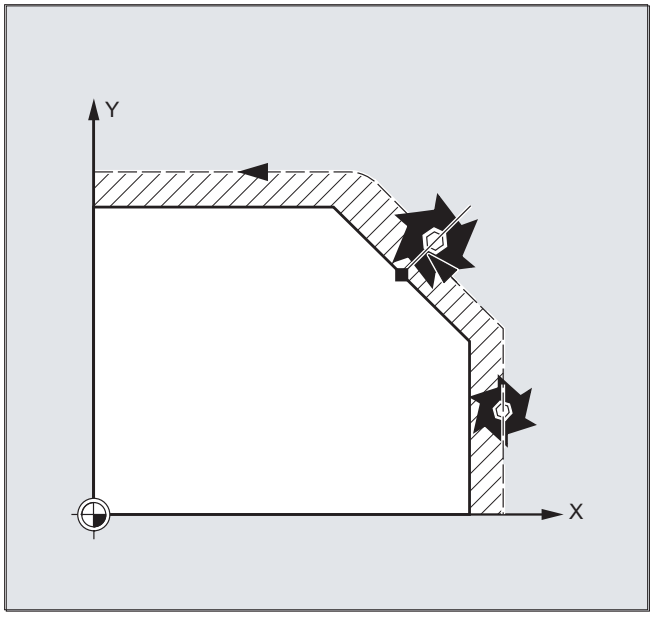
0 rámenet nincs definiálva
1 RMBBL: rámenet az elejére
2 RMIBL: rámenet a megszakításpontra
3 RMEBL: rámenet a mondat végpontra
4 RMNBL: rámenet a megszakított mondat legközelebbi pályapontjára

Rámenet új szerszámmal

Ha a programlefutás során szerszámtörés miatt megállás volt:

Az új D-szám programozásával a program az újra-rámeneti ponttól a megváltozott szerszámkorrekció értékekkel folytatódik.

Megváltozott szerszámkorrekciós értékeknél esetleg a megszakítási pontra nem lehet újra rámenni. Ebben az esetben a megszakítási ponthoz az új kontúron legközelebb levő (esetleg DISPR-rel módosított) pontra történik a rámenet.



Kontúrra rámenni

A mozgás, amivel a szerszám újra rámegy a kontúrra, programozható. A rámenet tengelyeinek címeit nulla értékkel adjuk meg.

A REPOSA, REPOSQA és REPOSHA utasításokkal automatikusan az összes tengely repozícionál. Nem szükséges tengely megadás.

A REPOSL, REPOSQ és REPOSH programozásánál a rámenet az összes geometria-tengelyen automatikusan, tehát az utasításban való megadás nélkül is megtörténik. Az összes többi repozícionálandó tengelyt az utasításban meg kell adni.

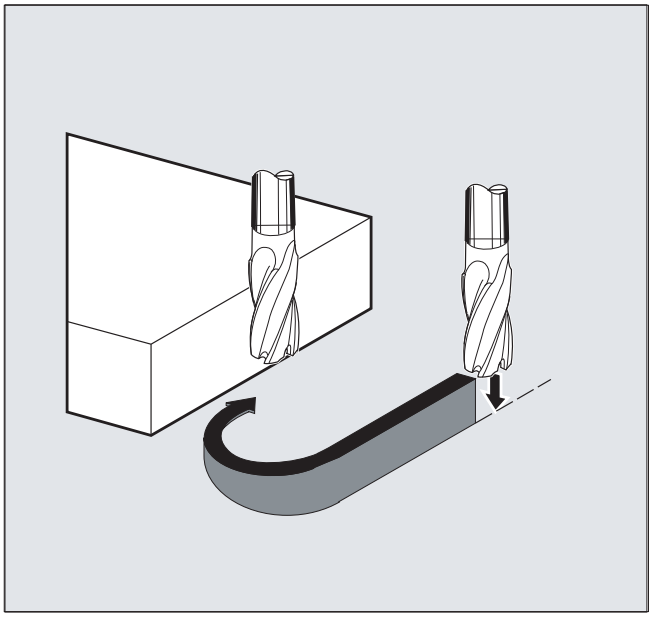
A REPOSH és REPOSQ körmozgásokra érvényes:

A kör a megadott G17 ... G19 munkasíkban lesz megtéve.

Ha a rámeneti mondatban a harmadik geometriatengely (fogásvétel irány) meg van adva, az újra-rámeneti pont abban az esetben, ha a szerszám pozíció és a programozott pozíció a fogásvételi irányban nem esik egybe, egy csavarvonalon lesz felvéve.

A következő esetekben automatikusan a REPOSL egyenes rámenetre lesz átkapcsolás:

- Nem adtunk meg értéket DISR-re.
- Nincs definiált rámeneti irány (program megszakítás egy mozgási információ nélküli mondatban).
- Az aktuális munkasíkra merőleges rámeneti iránynál.



12.10 Mozgás-vezetés befolyásolása

12.10.1 Százalékos rándítás-korrekció (JERKLIM)

A "JERKLIM" NC utasítással kritikus programszakaszokban a rándítás korlátozásának egy gépadatban beállított maximálisan lehetséges értékét lehet csökkenteni vagy növelni.

Előfeltétel

A SOFT gyorsítási módus aktív kell legyen.

Hatásosság

A funkció:

- az AUTOMATIK üzemmódokban.
- csak pályatengelyekre hat.

Szintaxis

```
JERKLIM(<tengely>)=<érték>
```

Jelentés

JERKLIM:	utasítás rándítás korrekcióra	
<tengely>:	géptengely, amelynek rándítás-határértékét illeszteni kell	
<érték>:	százalékos korrekció-érték, a pályamozgásnál a tengelyre beállított maximális értékre vonatkoztatva (MD32431 $MA_MAX_AX_JERK).	
	értéktartomány:	1 ... 200
	A 100-as érték nem befolyásolja a rándítást	

Megjegyzés

A JERKLIM viselkedése a munkadarabprogram végén és csatorna Reset-nél az MD32320 $MA_DYN_LIMIT_RESET_MASK bit 0-ban van beállítva:

- bit 0 = 0:
  A JERKLIM-re programozott érték a csatorna Reset / M30 esetén 100 %-ra lesz visszaállítva.
- bit 0 = 1:
  A JERKLIM-re programozott érték a csatorna Reset / M30 után is megmarad..

Példa

Programkód	Kommentár
...	

Programkód	Kommentár
N60 JERKLIM[X]=75	; A tengelyszán az X irányban csak a tengelyre megengedett rándítás maximum 75%-val gyorsítható/lassítható.
...	

12.10.2 Százalékos sebesség-korrekció (VELOLIM)

A VELOLIM utasítással a munkadarabprogramban vagy szinkronakcióban a gépadattal beállított maximálisan lehetséges sebességet egy tengelyre ill. a maximálisan lehetséges hajtóműfokozat független fordulatszámot egy orsó csökkenteni lehet.

Hatásosság

A funkció:

- az AUTOMATIK üzemmódokban.
- pálya- és pozícionáló-tengelyekre
- orsókra az orsó-/tengelyüzemben

Szintaxis

```
VELOLIM[<tengely/orsó>]=<érték>
```

Jelentés

VELOLIM:	utasítás a sebesség-korrekcióra	
<tengely/orsó>:	géptengely vagy orsó, amelynek sebesség- vagy fordulatszám-határértékét illeszteni kell **VELOLIM orsókra** Gépadattal (MD30455 $MA_MISC_FUNCTION_MASK, bit 6) be lehet állítani a munkadarabprogram programozásához, hogy a "VELOLIM" az aktuális orsó vagy tengely alkalmazástól függetlenül hasson (bit 6 = 1) vagy külön lehessen minden üzemmódra programozni (bit 6 = 0). Ha a külön hatás van beállítva, akkor a választás a jelölővel a programozásnál történik: • orsó-jelölő S<n> az orsó-üzemmódokra • tengely-jelölő, pl. "C", a tengely-üzemre	
<érték>:	százalékos korrekció-érték A százalékos érték vonatkozik: • tengelyeknél / orsóknál tengely-üzemben (ha MD30455, bit 6 = 0): a beállított maximális tengelysebességre (MD32000 $MA_MAX_AX_VELO). • orsóknál orsó- vagy tengely-üzemben (ha MD30455 bit 6 == 1): az aktív hajtóműfokozat maximális fordulatszámára (MD35130 $MA_GEAR_STEP_MAX_VELO_LIMIT[<n>])	
	értéktartomány:	1 ... 100
	A 100-as érték nem befolyásolja a sebességet ill. a fordulatszámot.	

Megjegyzés

Viselkedés munkadarabprogram végén és csatorna-Reset-nél

A "VELOLIM" viselkedése munkadarabprogram végén és csatorna Reset-nél egy gépadattal beállítható: MD32320 $MA_DYN_LIMIT_RESET_MASK, Bit 0

Egy aktív fordulatszám-határolás felismerése orsó-üzemben

Egy fordulatszám-határolást "VELOLIM" által (kisebb 100 %-nál) orsó-üzemben a következő rendszerváltozókkal lehet felismerni:

- $AC_SMAXVELO (maximálisan lehetséges orsó-fordulatszám)
- $AC_SMAXVELO_INFO (fordulatszám határoló ok jelölése)

Példák

Példa 1: Géptengely sebesség-határolás

Programkód	Kommentár
...	
N70 VELOLIM[X]=80	; A tengelyszán az X irányban csak a tengelyre megengedett sebesség maximum 80 %-val mozogjon.
...	

Példa 2: Orsó fordulatszám-határolás

Programkód	Kommentár
N05 VELOLIM[S1]=90	; Az orsó 1 maximális fordulatszámának határolása az 1000 ford/perc 90 %-ára.
...	
N50 VELOLIM[C]=45	; A fordulatszám határolása az 1000 ford/perc 45 %-ára, C az S1 tengely-jelölője.
...	

Orsó 1 (AX5) gépadat beállítások

- hajtómű fokozat 1 maximális fordulatszám = 1000 ford/perc:
  MD35130 $MA_GEAR_STEP_MAX_VELO_LIMIT[1, AX5] = 1000
- A "VELOLIM" programozása együtt hat az orsó- és a tengely-üzemre a programozott jelölőtől függetlenül:
  MD30455 $MA_MISC_FUNCTION_MASK[AX5], bit 6 = 1

12.10.3 Programpélda VELOLIM-ra és JERKLIM-re

A következő program egy alkalmazási példa a százalékos rándítás- és sebesség-határolásra:

Programkód	Kommentár
N1000 G0 X0 Y0 F10000 SOFT G64	
N1100 G1 X20 RNDM=5 ACC[X]=20	
ACC[Y]=30	
N1200 G1 Y20 VELOLIM[X]=5	; A tengelyszán az X irányban csak a tengelyre megengedett sebesség maximum 5%-val mozogjon.
JERKLIM[Y]=200	; A tengelyszán az Y irányban csak a tengelyre megengedett rándítás maximum 200%-val gyorsítható/lassítható.
N1300 G1 X0 JERKLIM[X]=2	; A tengelyszán az X irányban csak a tengelyre megengedett rándítás maximum 2%-val gyorsítható/lassítható.
N1400 G1 Y0	
M30	

12.11 Kontúr-/tájolás-tűrés programozása (CTOL, OTOL, ATOL)

A CTOL, OTOL és ATOL címekkel a gép- és beállítási-adatok paraméterezett megmunkálási tűrések a kompresszor funkciókhoz, átsimításhoz és tájolás-simításhoz a munkadarabprogramban illeszthetők.

A programozott tűrésértékek az újra programozásukig vagy egy negatív érték hozzárendelésével a törlésükig érvényesek. Ezek távolabb lesznek törölve a programvégnél, ill. Reset-nél. A törlés után ismét a paraméterezett tűrésértékek hatásosak.

Szintaxis

```
CTOL=<Value>
CTOL=<Value>
ATOL[<Axis>]=<Value>
```

Jelentés

CTOL:	cím a **kontúr-tűrés** programozására			
	alkalmazási terület:	• összes kompresszor funkcióra • összes átsimítási módra G641 és G644 kivételével		
	előrefutás-állj:	nem		
	hatásosság:	modális		
	<Value>:	A kontúr-tűrés értéke egy hosszadat.		
		típus:	REAL	
		egység:	hüv/mm (a méretmegadás aktuális beállításától függően)	
		értéktartomány:	≥ 0:	tűrésérték
			< 0:	programozott tűrésérték törlése ⇒ Ismét a gépadatban ill. beállítási adatban paraméterezett tűrésérték érvényes.
OTOL:	cím a **tájolás-tűrés** programozására			
	alkalmazási terület:	• összes kompresszor funkcióra • ORISON tájolás-simításra • összes átsimítási módra G641, G644 és OSD kivételével		
	előrefutás-állj:	nem		
	hatásosság:	modális		
	<Value>:	A tájolás-tűrés értéke egy szögadat.		
		típus:	REAL	
		egység:	fok	
		értéktartomány:	≥ 0:	tűrésérték
			< 0:	programozott tűrésérték törlése ⇒ Ismét a gépadatban ill. beállítási adatban paraméterezett tűrésérték érvényes.

ATOL:	cím egy **tengely-specifikus tűrés** programozására			
	alkalmazási terület:	• összes kompresszor funkcióra • ORISON tájolás-simításra • összes átsimítási módra G641, G644 és OSD kivételével		
	előrefutás-állj:	nem		
	hatásosság:	modális		
	<Axis>:	csatorna-tengely, amelyre a programozott tűrés hasson		
	<Value>:	tengely-tűrés értéke a tengelytípustól függően (egyenes- vagy körtengely) egy hossz- vagy szögadat		
		típus:	REAL	
		egység:	egyenes-tengelyekhez:	hüv/mm (a méretmegadás aktuális beállításától függően)
			körtengelyekhez::	fok
		értéktartomány:	≥ 0:	tűrésérték
			< 0:	programozott tűrésérték törlése ⇒ Ismét a gépadatban ill. beállítási adatban paraméterezett tűrésérték érvényes.

Megjegyzés

A CTOL-lal és OTOL-lal programozott csatorna-specifikus tűrésértékek magasabb prioritásúak az ATOL-lal programozott tengely-specifikus tűrésértékeknél.

Megjegyzés

Skálázandó frame-k

A skálázandó frame-k a programozott tűrésekre úgy hatnak, mint a tengelypozíciókra, vagyis a relatív tűrés azonos marad.

Példa

Programkód	Kommentár
COMPCAD G645 G1 F10000	; COMPCAD kompresszor funkció aktiválni.
X... Y... Z...	; Itt hatnak a gépadatok és a beállítási adatok.
X... Y... Z...	
X... Y... Z...	
CTOL=0.02	; Ettől kezdve hatásos a 0,02 mm kontúrtűrés.
X... Y... Z...	
X... Y... Z...	
X... Y... Z...	
ASCALE X0.25 Y0.25 Z0.25	; Ettől kezdve hatásos a 0,005 mm kontúrtűrés.
X... Y... Z...	
X... Y... Z...	
X... Y... Z...	

Programkód	Kommentár
CTOL=-1	; Ettől kezdve hatnak ismét a gépadatok és a beállítási adatok.
X... Y... Z...	
X... Y... Z...	
X... Y... Z...	

rendszerváltozók

Olvasás előrefutás-állj-jal

A következő rendszerváltozókkal a munkadarabprogramban és a szinkronakcióban az aktuálisan hatásos tűrések olvashatók:

- $AC_CTOL
  Csatorna-specifikus kontúrtűrés, amelyik az aktuális főfutam mondat feldolgozásánál hatásos volt
  Ha egy kontúrtűrés hatásos, a $AC_CTOL a geometria-tengelyek tűrések négyzetei összegének a négyzetgyökét adja
- $AC_OTOL
  Csatorna-specifikus tájolás-tűrés, amelyik az aktuális főfutam mondat feldolgozásánál hatásos volt
  Ha nincs hatásos tájolás-tűrés , a $AC_OTOL egy aktív tájolási transzformáció alatt a tájolás-tengelyek tűrések négyzetei összegének a négyzetgyökét adja, egyébként a "-1" értéket.
- $AA_ATOL[<tengely>]
  Tengely-specifikus kontúrtűrés, amelyik az aktuális főfutam mondat feldolgozásánál hatásos volt
  Ha egy kontúrtűrés aktív, a $AA_ATOL[<geometria-tengely>] a kontúrtűrés osztva a geometria-tengelyek számának gyökével értéket adja.
  Ha egy tájolás-tűrés és egy tájolási transzformáció aktív, az $AA_ATOL[<tájolás-tengely>] a tájolás-tűrés osztva a tájolás-tengelyek számának gyökével értéket adja.

Megjegyzés

Ha nem lettek programozva tűrésértékek, akkor a $A-változók nem eléggé differenciáltak, az egyes funkciók tűréseinek a megkülönböztetésére.

Ilyen esetek felléphetnek, ha a gépadatok és beállítási adatok a kompresszor funkciókra, átsimításra és tájolás simításra különböző tűréseket állítanak be. A rendszerváltozók ekkor az éppen aktív funkcióknál fellépő legnagyobb értéket adják. Ha pl. egy tájolás funkció 0,1° tájolás-tűréssel és egy ORISON tájolás simítás 1°-kal aktív, az $AC_OTOL az "1" értéket adja. Ha tájolás simítás ki van kapcsolva, az $AC_OTOL a "0,1" értéket adja.

Olvasás előrefutás-állj nélkül

A következő rendszerváltozókkal a munkadarabprogramban az aktuálisan hatásos tűrések olvashatók:

- $P_CTOL
  Aktuálisan hatásos csatorna-specifikus kontúr-tűrés.
- $P_OTOL
  Aktuálisan hatásos csatorna-specifikus tájolás-tűrés.
- $PA_ATOL
  Aktuálisan hatásos tengely-specifikus kontúr-tűrés.

Peremfeltételek

A CTOL-lal, OTOL-lal és ATOL-lal programozott tűrésértékek olyan funkciókra is hatnak, amelyek közvetetten ezektől a tűrésektől függnek,

- A húr-hiba határolása a parancsérték számításánál
- A szabad formájú felület módus alapfunkciói

A következő átsimítás funkciók a CTOL, OTOL és ATOL programozásával **nem** lesznek befolyásolva:

- Tájolás átmenet simítása OSD-val
  Az OSD nem használ tűrést, hanem egy távolságot a mondat-átmenethez.
- Átsimítás G644-gyel
  A G644 nem megmunkálásra szolgál, hanem a szerszámváltás és más mozgások optimalizálására megmunkálás nélkül.
- Átsimítás G645-tel
  A G645 szinte mindig úgy viselkedik, mint a G642 és ezzel a programozott tűréseket használja. Csak állandó érintőjű mondat-átmeneteknél görbület-ugrással, pl. egy érintőleges kör-egyenes átmenet, lesz használva a tűrésérték az MD33120 $MA_PATH_TRANS_POS_TOL gépadatból. Az átsimítási út ezeken a helyeken lehet a programozott kontúr külső oldalán is, ahol sok alkalmazás kevésbé toleráns. Ezen kívül általában elegendő egy kicsi, fixen beállított tűrés a görbületi változások kiegyenlítéséhez, amelyekkel az NC programozónak nem kell törődnie.

12.12 Mondatváltás viselkedés aktív csatolásnál (CPBC)

A CPBC utasítással megadásra kerül a mondatváltás kritérium, amelynek teljesülnie kell, hogy aktív csatolásnál egy mondatváltás végre legyen hajtva.

Szintaxis

```
CPBC[<követő tengely>] = <kritérium>
```

Jelentés

CPBC:	Mondatváltás kritérium aktív csatolásnál	
<követő tengely>:	követő tengely tengely-jelölő	
<kritérium>:	mondatváltás kritérium	
	típus:	STRING
	Érték	**Jelentés:** Mondatváltás történik
	"NOC"	függetlenül a csatolás állapotától
	"IPOSTOP"	parancsérték oldali szinkronfutásnál
	"COARSE"	valósérték oldali "durva" szinkronfutásnál
	"FINE"	valósérték oldali "finom" szinkronfutásnál

Példa

Programkód

```
; mondatváltás történik:
;  - csatolás a követő tengelyhez X2 == aktív
;  - parancsérték oldali szinkronfutás == aktív
CPBC[X2]="IPOSTOP"
```

Tengely-csatolások 13

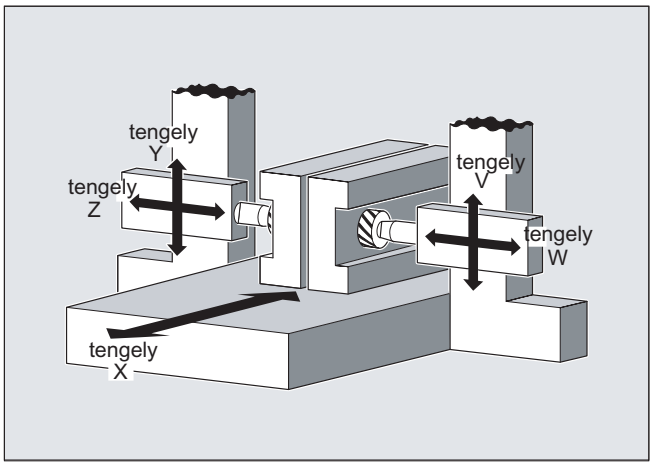
13.1 Vontatás (TRAILON, TRAILOF)

Egy megadott vezető tengely mozgatásánál a hozzárendelt vontatott tengelyek (= követő tengelyek) egy csatolási tényező figyelembe vételével a vezető tengelytől levezetett mozgás utat teszik meg.

A vezető tengely és a követő tengelyek egy vontatás egyesülést képeznek.

Alkalmazási terület

- Egy tengely mozgatása egy szimulált tengely által. A vezető tengely egy szimulált tengely és a vontatott tengely egy valódi tengely. Ezzel egy valódi tengelyt a csatolási tényező figyelembe vételével lehet mozgatni.
- Kétoldalas megmunkálás 2 vontatási egyesüléssel:
  1. vezető tengely Y, vontatott tengely V
  2. vezető tengely Z, vontatott tengely W



Szintaxis

```
TRAILON(<követő tengely>,<vezető tengely>,<csatolási tényező>)
TRAILOF(<követő tengely>,<vezető tengely>,<vezető tengely 2>)
TRAILOF(<követő tengely>)
```

Jelentés

`TRAILON`:	utasítás egy vontatási egyesülés definiálására és bekapcsolására	
	hatásosság:	modális
`<követő tengely>`:	paraméter 1: követő tengely tengely-jelölője **Utalás:** Egy vontatott tengely lehet vezető tengely további vontatott tengelyek számára.. Ezen a módon különböző vontatási egyesüléseket lehet felépíteni:	

`<vezető tengely>`:	paraméter 2: vezető tengely tengely-jelölője	
`<csatolás>`:	paraméter 3: csatolási tényező: A csatolási tényező megadja vezető tengely és a vontatott tengelyek útjainak arányát: <csatolási tényező> = vontatott tengely út / vezető tengely út	
	típus:	REAL
	elő-beállítás:	1
	Egy negatív érték megadása a vezető és a vontatott tengelyek ellentétes mozgását okozza. Ha a programozásnál nem lesz megadva csatolási tényező, akkor automatikusan a csatolási tényező 1 érvényes.	
`TRAILOF`:	utasítás a vontatási egyesülés kikapcsolására	
	hatásosság:	modális

	`TRAILOF` 2 paraméterrel csak a megadott vezető tengely csatolását kapcsolja ki.:
	`TRAILOF(<követő tengely>,<vezető tengely>)`
	Ha egy vontatott tengelynek 2 vezető tengelye van, a csatolás kikapcsolásához a `TRAILOF`-ot 3 paraméterrel kell felhívni:
	`TRAILOF(<követő tengely>,<vezető tengely>,<vezető tengely 2>)`
	Ugyanazt az eredményt adja a `TRAILOF` programozása a vezető tengely megadása nélkül:
	`TRAILOF(<követő tengely>)`	

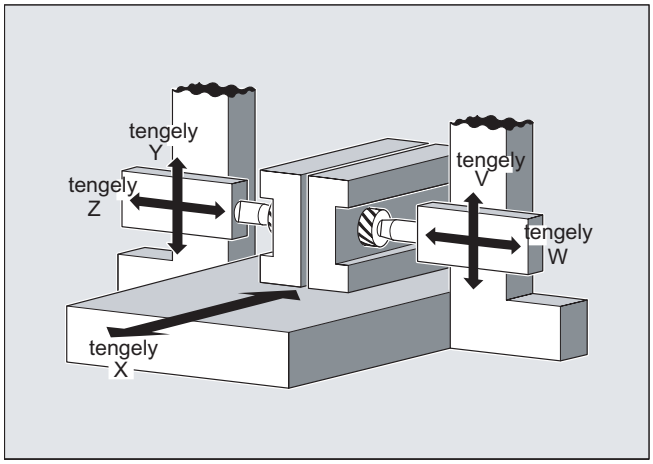
Megjegyzés

A vontatás mindig az alap-koordinátarendszerben (AKR) történik.

Az egyidejűleg aktiválható vontatási egyesülések számát csak a gép tengelyeinek kombinációs lehetőségei korlátozzák.

Példa

A munkadarabot kétoldalúan az ábrázolt tengely-konstellációval kell megmunkálni. Ehhez 2 vontatási egyesülést képezünk.



Programkód	Kommentár
…	
N100 TRAILON(V,Y)	; 1. vontatási egyesülés bekapcsolása
N110 TRAILON(W,Z,-1)	; 2. vontatási egyesülés bekapcsolása. A csatolási tényező negatív: a vontatott tengely mindig a vezető tengellyel ellentétes irányban mozog.
N120 G0 Z10	; Fogásvétel a Z és W tengelyeken ellentétes irányban.
N130 G0 Y20	; Fogásvétel az Y és V tengelyeken azonos tengelyirányban.
…	
N200 G1 Y22 V25 F200	; Egy függő és egy független mozgás összeadódása a V vontatott tengelyen.
…	
TRAILOF(V,Y)	; 1. vontatási egyesülés kikapcsolása.
TRAILOF(W,Z)	; 2. vontatási egyesülés kikapcsolása.

További információk

Tengelytípusok

Egy vontatási egyesülés az egyenes- és körtengelyek tetszőleges kombinációjából áll. Vezető tengelyként lehet egy szimulált tengelyt definiálni.

Vontatott tengelyek

Egy vontatott tengelyt egyidejűleg maximum 2 vezető tengelyhez lehet hozzárendelni. A hozzárendelés különböző vontatási egyesülésekben történik.

Egy vontatott tengelyt minden rendelkezésre álló mozgás-utasítással programozható (G0, G1, G2, G3, …). A függetlenül definiált utakhoz kiegészítőleg a vontatott tengely a csatolási tényezővel a vezető tengelyéből levezetett utat teszi meg.

Dinamika korlátozás

A dinamika korlátozása függ a vontatási egyesülés aktiválásának módjától:

- Aktiválás munkadarabprogramban
  Ha az aktiválás a munkadarabprogramban történik és az összes vezető tengely program-tengely az aktiváló csatornában, a vezető tengely mozgatásánál az összes vontatott tengely dinamikája úgy lesz figyelembe véve, hogy egyetlen vontatott tengely se legyen túlterhelve.
  Ha az aktiválás a munkadarabprogramban olyan vezető tengelyekkel történik, amelyek nem program-tengely az aktiváló csatornában ($AA_TYP ≠ 1), a vezető tengely mozgatásánál a vontatott tengelyek dinamikája nem lesz figyelembe véve. Ezáltal a vontatott tengelyeknél a csatoláshoz szükségesnél alacsonyabb dinamikával létre jöhet egy túlterhelés.
- Aktiválás szinkronakcióban
  Ha az aktiválás egy szinkronakcióban történik, a vezető tengely mozgatásánál a vontatott tengelyek dinamikája nem lesz figyelembe véve. Ezáltal a vontatott tengelyeknél a csatoláshoz szükségesnél alacsonyabb dinamikával létre jöhet egy túlterhelés.

⚠ VIGYÁZAT

Tengely túlterhelés

Ha egy vontatási egyesülés aktiválása

- szinkronakciókban
- a munkadarabprogramban olyan vezető tengelyekkel történik, amelyek nem program-tengelyek a vontatott tengely csatornájában

akkor a felhasználó/gépgyártó különleges felelőssége megfelelő intézkedéseket bevezetni, hogy a vezető tengely mozgásai ne okozzanak túlterhelést a követő tengelyeknél.

Csatolás állapota

A munkadarabprogramban a csatolás állapotát a következő rendszerváltozókkal lehet lekérdezni:

$AA_COUP_ACT[<tengely>]

Érték	Jelentés
0	nincs aktív csatolás
8	vontatás aktív

Vontatás tengely maradék út kijelzése modulo körtengelyeknél

Ha a vezető és a vontatott tengelyek modulo körtengelyek, a vezető tengely elmozdulásai n * 360° , n = 1, 2, 3... összegződnek a vontatott tengely maradék út kijelzésében a csatolás kikapcsolásáig.

Példa: Programszakasz TRAILON-nal és vezető tengely B és követő tengely C

Programkód	Kommentár
`TRAILON(C,B,1)`	`; csatolást bekapcsolni`
`G0 B0`	`; kiinduló pozíció`
	`; maradék út kijelzés a mondat kezdeten:`

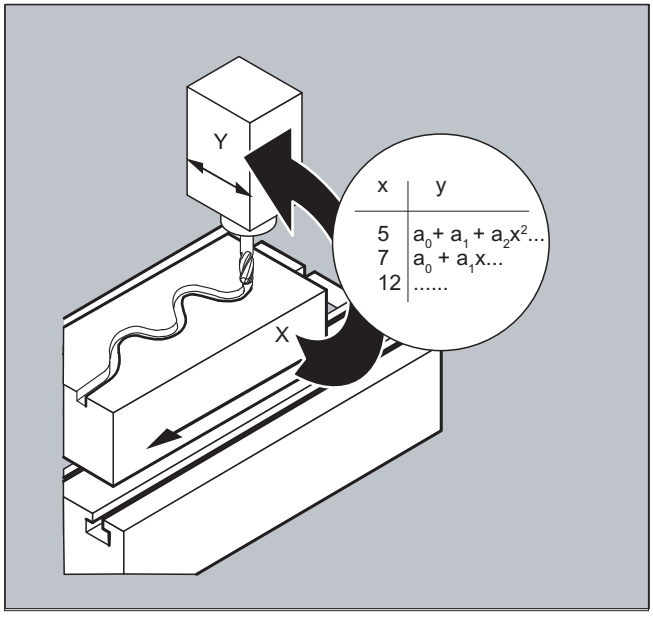
Programkód	Kommentár
G91 B360	; B=360, C=360
G91 B720	; B=720, C=1080
G91 B360	; B=360, C=1440

13.2 Görbe-táblázatok (CTAB)

A görbe-táblázatok segítségével lehet két tengely (vezető és követő tengely) között pozíció- és sebesség-összefüggéseket programozni. A görbe-táblázatok definíciója a munkadarabprogramban történik.

Alkalmazás

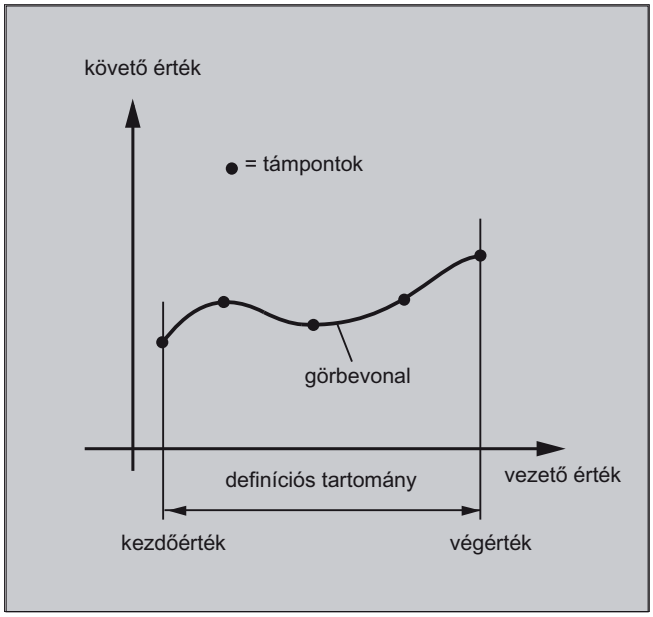
A görbe-táblázatok helyettesítik a görbe-tárcsákat. A görbe-táblázatok adják az alapját a tengelyek vezető érték csatolásának, amennyiben funkcionális összefüggést teremtenek a vezető és a követő értékek között: A vezérlés a megfelelő programozásnál a vezető és követő tengelyek egymáshoz rendelt pozícióiból egy polinomot számít ki, ami megfelel görbe-tárcsának.



13.2.1 Görbe-táblázatot definiálni (CTABDEF, CATBEND)

Egy görbe-táblázat egy munkadarab-program vagy egy munkadarab-program részlet, amely a CTABDEF utasítással kezdődik és a CTABEND utasítással zárul le.

Egy ilyen munkadarab-program részleten belül mozgás utasításokkal a vezető tengely egyes pozícióihoz egyértelmű követő tengely pozíció van rendelve, amelyek támpontonként szolgálnak a görbe kiszámításához egy maximum 5. fokú polinom formájában.



Előfeltétel

A görbe-táblázatok definíciójához a megfelelő gépadat beállítással kell foglalni tárolóhelyet (→ gépgyártó!)

Szintaxis

```
CTABDEF(<követő tengely>,<vezető
tengely>,<n>,<periodicitás>[,<tárolóhely>])
...
CTABEND
```

Jelentés

`CTABDEF( )`:	görbe-táblázat definíció kezdete	
`CTABEND`:	görbe-táblázat definíció vége	
`<követő tengely>`:	tengely, amelynek mozgását a görbe-táblázat alapján kell kiszámítani	
`<vezető tengely>`:	tengely, amely a vezető értéket adja a követő tengely mozgásának kiszámításához	
`<n>`:	görbe-táblázat száma (**ID**) Egy görbe-táblázat száma egyértelmű és független a tárolóhelytől. Nem lehetnek azonos számú táblázatok az NC statikus és dinamikus tárolójában.	
`<periodicitás>`:	táblázat periodicitás	
	`0`	táblázat nem periodikus (csak egyszer lesz feldolgozva, körtengelyeknél is)
	`1`	táblázat periodikus a vezető tengely vonatkozásában
	`2`	táblázat periodikus a vezető tengely és a követő tengely vonatkozásában

<tárolási hely>:	tárolóhely megadása (opcionális)	
	"SRAM"	A görbe-táblázat a **statikus** NC-tárolóban lesz eltárolva.
	"DRAM"	A görbe-táblázat a **dinamikus** NC-tárolóban lesz eltárolva.
	Utalás: Ha erre a paraméterre nincs érték programozva, az MD20905 $MC_CTAB_DEFAULT_MEMORY_TYPE gépadatban beállított tárolóhely lesz alkalmazva	

Megjegyzés

Átírás

Egy görbe-táblázat át lesz írva. ha egy új táblázat definíciója ennek az (<n>) számát használja (kivétel: a görbe-táblázat egy tengely-csatolásban aktív vagy CTABLOCK-kal zárolva van). **Az átírásnál nem lesz kiadva megfelelő figyelmeztetés!**

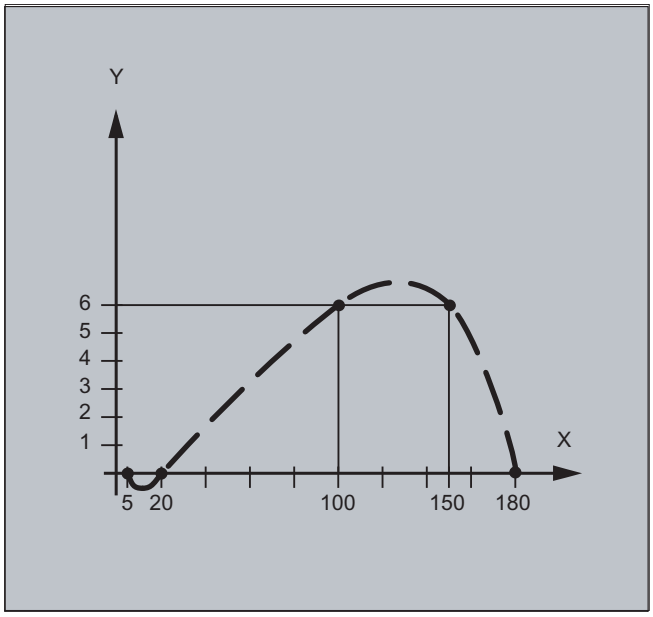
Példák

Példa 1: Programszakasz görbe-táblázat definíciójaként

Egy programszakaszt változás nélkül kell egy görbe-táblázat definíciójához használni. A benne levő STOPRE előrefutás állj utasítás megmaradhat és ismét azonnal aktív lesz, ha a programszakasz nem táblázat definícióhoz lesz felhasználva és a CTABDEF és a CTABEND el lett távolítva.

Programkód	Kommentár
…	
CTABDEF(Y,X,1,1)	; Egy görbe-táblázat definíciója.
…	
IF NOT ($P_CTABDEF)	
STOPRE	
ENDIF	
…	
CTABEND	

Példa 2: Egy nem-periódikus görbe-táblázat definíciója



Programkód	Kommentár
N100 CTABDEF(Y,X,3,0)	; A 3-as számú, nem-periódikus görbe-táblázat definíciójának kezdete.
N110 X0 Y0	; 1. mozgás-utasítás, megadja a kezdőértéket és az 1. támpontot:
	vezető érték: 0; követő érték: 0
N120 X20 Y0	; 2. támpont:
	vezető érték: 0...20, követő érték: kezdőérték…0
N130 X100 Y6	; 3. támpont:
	vezető érték: 20...100, követő érték: 0…6
N140 X150 Y6	; 4. támpont:
	vezető érték: 100...150, követő érték: 6…6
N150 X180 Y0	; 5. támpont:
	vezető érték: 150...180, követő érték: 6…0
N200 CTABEND	; A definíció vége. A görbe-táblázat a belső ábrázolásban maximum 5. fokú polinomként lesz létrehozva. A görbe kiszámítása a megadott támpontokkal függ a modálisan kiválasztott interpolációs módtól (kör-, egyenes-, Spline-interpoláció). A munkadarab-program definíció kezdete előtti állapota helyre lesz állítva.

Példa 3: Egy periódikus görbe-táblázat definíciója

A 2-es számú periodikus görbe-táblázat definíciója, vezető érték tartomány 0 ...360, követő-tengely mozgása 0-tól 45-ig és vissza 0-ra:

Programkód	Kommentár
N10 DEF REAL DEPPOS	
N20 DEF REAL GRADIENT	

```
Programkód                                          Kommentár
N30 CTABDEF(Y,X,2,1)                                ; A definíció kezdete.
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                                        ; A definíció vége.
;görbe tesztje az Y csatolásával X-hez
N120 G1 F1000 X0
N130 LEADON(Y,X,2)
N140 X360
N150 X0
N160 LEADOF(Y,X)

N170 DEPPOS=CTAB(75.0,2,GRADIENT)                   ; A táblázat funkció olvasása ve-
                                                    zető érték 75.0-nél.
N180 G0 X75 Y=DEPPOS                                ; Vezető és követő tengely pozici-
                                                    onálása.
; A csatolás bekapcsolása után nem szükséges követő tengely szinkronizálása.
N190 LEADON(Y,X,2)
N200 G1 X110 F1000
N210 LEADOF(Y,X)
N220 M30
```

További információk

Görbe-táblázat kezdő és vég értéke

A görbe-táblázat definíciós-tartományának kezdete az első összetartozó tengely-pozíciók (az első mozgás-utasítások) a görbe-táblázat definíción belül. A görbe-táblázat definíciós-tartományának végét az utolsó mozgás-utasítás határozza meg.

Rendelkezésre álló nyelvi terjedelem

A görbe-táblázat definíciójában a teljes NC-nyelv rendelkezésre áll.

Megjegyzés

A következő megadások nem megengedettek a görbe-táblázatokban:

- előrefutás-állj
- ugrások a vezető tengelyek mozgásában (pl. transzformációk váltása)
- mozgás utasítás csak a követő tengelyre
- mozgásirány váltás a vezető tengelyen, vagyis a vezető tengely pozíciója mindig egyértelmű kell legyen
- CTABDEF és CTABEND utasítás különböző programszinteken.

Modális utasítások hatásossága

Az összes modálisan hatásos utasítás, amelyek a görbe-táblázat definíción belül vannak, a táblázat definíció lezárása után érvénytelen. A munkadarab-program, amelyben a táblázat definíció van, ezzel a táblázat definíció előtt és után azonos állapotban van.

Hozzárendelés R paraméterekhez

A hozzárendelések az R-paraméterekhez a táblázat definíción belül a CTABEND után törölve lesznek.

Példa:

Programkód	Kommentár
...	
R10=5 R11=20	; R10=5
...	
CTABDEF	
G1 X=10 Y=20 F1000	
R10=R11+5	; R10=25
X=R10	
CTABEND	
...	; R10=5

ASPLINE, BSPLINE, CSPLINE aktiválása

Ha egy CTABDEF ... CTABEND görbe-táblázaton belül egy ASPLINE, BSPLINE vagy CSPLINE lesz aktiválva, akkor ezen Spline aktiválás előtt legalább egy kezdőpontot kell programozni. Azonnali aktiválást a CTABDEF után el kell kerülni, mert különben a Spline a görbe-táblázat definíció előtti aktuális tengelypozíciótól függ.

Példa:

Programkód
...
CTABDEF(Y,X,1,0)
X0 Y0

Programkód

```
ASPLINE
X=5 Y=10
X10 Y40
...
CTABEND
```

Görbe-táblázatok ismételt használata

A vezető és követő tengelyeknek a görbe-táblázat által megadott összefüggése a megadott táblázatszámmal megmarad a munkadarabprogram vége és a Power-Off után is, ha a táblázat a statikus NC-tárolóban (SRAM) van elhelyezve.

A dinamikus tárolóban (DRAM) elhelyezett táblázat Power-On-nál törölve lesz és esetleg még egyszer létre kell hozni.

Az egyszer létrehozott görbe-táblázat a vezető és követő tengelyek tetszőleges tengely-kombinációira használható attól függetlenül, hogy melyik tengelyek voltak a görbe-táblázat létrehozásához használva.

Görbe-táblázatok átírása

Egy görbe-táblázat átírásra kerül, ha egy új táblázat definíció a számát használja.

Kivétel: Egy görbe-táblázat egy tengely-csatolásban aktív vagy CTABLOCK-kal zárolva van.

Megjegyzés

A görbe-táblázatok átírásánál nem lesz megfelelő figyelmeztetés kiadva!

Görbe-táblázat definíció aktív?

A `$P_CTABDEF` rendszerváltozóval mindig le lehet kérdezni egy munkadarabprogramból, hogy egy görbe-táblázat definíció aktív-e.

Görbe-táblázat definíció megszüntetése

A munkadarabprogram részlet a görbe-táblázat definíció utasítások kihagyása után valódi munkadarabprogramként használható.

Görbe-táblázatok betöltése "Feldolgozás kívülről" által

A görbe-táblázatok kívülről történő feldolgozásánál az utántöltés puffer (DRAM) méretét az MD18360 $MN_MM_EXT_PROG_BUFFER_SIZE gépadatban úgy kell beállítani, hogy a teljes görbe-táblázat definíció egyidőben elhelyezhető legyen az utántöltés pufferben. A munkadarabprogram feldolgozása egyébként egy vészjelzéssel megszakad.

Követő tengely ugrások

A
MD20900 $MC_CTAB_ENABLE_NO_LEADMOTION
gépadat beállításától függően a követő tengely ugrásai elfogadhatók a vezető tengely mozgása nélkül.

13.2.2 Egy görbe-táblázat előfordulását megvizsgálni (CTABEXISTS)

A CTABEXISTS utasítással lehet megvizsgálni egy adott görbe-táblázat előfordulását az NC-tárolóban.

Szintaxis

```
CTABEXISTS(<n>)
```

Jelentés

CTABEXISTS:	Megvizsgálja az <n> számú görbe-táblázat előfordulását a statikus vagy dinamikus NC-tárolóban.	
	0	táblázat nem létezik
	1	táblázat létezik
<n>:	görbe-táblázat száma (**ID**)	

13.2.3 Görbe-táblázatok törlése (CTABDEL)

A CTABDEL-lel lehet a görbe-táblázatokat törölni.

Megjegyzés

A görbe-táblázatok, amelyek egy csatolásban aktívak, nem törölhetők.

Szintaxis

```
CTABDEL(<n>)
CTABDEL(<n>,<m>)
CTABDEL(<n>,<m>,<tárolóhely>)
CTABDEL()
CTABDEL(,,<tárolóhely>)
```

Jelentés

CTABDEL:	utasítás a görbe-táblázatok törléséhez
<n>:	törlendő görbe-táblázat száma (**ID**) Egy CTABDEL(<n>,<m>) görbe-táblázat tartomány törlésénél az <n> megadja a tartomány első görbe-táblázatát.
<m>:	Egy CTABDEL(<n>,<m>) görbe-táblázat tartomány törlésénél az <m> megadja a tartomány utolsó görbe-táblázatát. <m> nagyobb kell legyen, mint az <n>!

<tárolási hely>:	tárolóhely megadása (opcionális) Egy törlésnél a tárolóhely **megadása nélkül** a megadott görbe-táblázatok a statikus és a dinamikus NC-tárolóban törölve lesznek. Egy törlésnél a tárolóhely **megadásával** a megadott görbe-táblázatok csak a megadott tárolóban lesznek törölve. A többiek megmaradnak.	
	"SRAM"	törlés a **statikus** NC-tárolóban
	"DRAM"	törlés a **dinamikus** NC-tárolóban

Ha a CTABDEL a törlendő görbe-táblázat megadása nélkül van programozva, akkor az **összes** görbe-táblázat(ok) törölve lesznek a megadott tárolóban.

CTABDEL():	Törli az összes görbe-táblázatot a statikus és a dinamikus NC-tárolóban.
CTABDEL(,,"SRAM"):	Törli az összes görbe-táblázatot a statikus NC-tárolóban.
CTABDEL(,,"DRAM"):	Törli az összes görbe-táblázatot a dinamikus NC-tárolóban.

Megjegyzés

Ha a CTABDEL(<n>,<m>) vagy CTABDEL() többszörös törlésnél legalább egy törlendő görbe-táblázat egy csatolásban aktív, akkor a törlés utasítás nem lesz végrehajtva, vagyis a megcímzett görbe-táblázatokból **egy sem** lesz törölve.

13.2.4 Görbe-táblázatok zárolása törlésre és átírásra (CTABLOCK, CTABUNLOCK)

A görbe-táblázatokat a szándékolatlan törlés és átírás ellen zárolás beállításával lehet védeni. Egy beállított zárolást bármikor fel lehet oldani.

Szintaxis

Zárolást beállítani:

```
CTABLOCK(<n>)
CTABLOCK(<n>,<m>)
CTABLOCK(<n>,<m>,<tárolóhely>)
CTABLOCK()
CTABLOCK(,,<tárolóhely>)
```

Zárolást megszüntetni:

```
CTABUNLOCK(<n>)
CTABUNLOCK(<n>,<m>)
CTABUNLOCK(<n>,<m>,<tárolóhely>)
CTABUNLOCK()
CTABUNLOCK(,,<tárolóhely>)
```

Jelentés

CTABLOCK:	utasítás törlés/átírás elleni zárolás **beállítására**
CTABUNLOCK:	utasítás törlés/átírás elleni zárolás **feloldására** A CTABUNLOCK a CTABLOCK-kal zárolt görbe-táblázatokat ismét szabaddá teszi. Egy aktív csatolásban résztvevő görbetáblázatok továbbra is zároltak maradnak és nem lehet őket törölni. A zárolás a CTABLOCK-kal megszűnik, amikor az aktív csatolás miatti zárolás megszűnik a csatolás deaktiválása miatt. Ezután lehet törölni ezeket a táblázatokat. Egy ismételt CTABUNLOCK hívás nem szükséges.
<n>:	zárolandó/feloldandó görbe-táblázat száma (**ID**) Egy CTABLOCK(<n>,<m>)/CTABUNLOCK(<n>,<m>) görbe-táblázat zárolásnál/feloldásnál az <n> megadja a tartomány első görbe-táblázatának számát.
<m>:	Egy CTABLOCK(<n>,<m>)/CTABUNLOCK(<n>,<m>) görbe-táblázat zárolásnál/feloldásnál az <m> megadja a tartomány utolsó görbe-táblázatának számát. <m> nagyobb kell legyen, mint az <n>!
<tárolási hely>:	tárolóhely megadása (opcionális) Egy zárolás beállításánál/feloldásánál a tárolóhely **megadása nélkül** a megadott görbe-táblázatok a statikus és a dinamikus NC-tárolóban zárolva/feloldva lesznek. Egy zárolás beállításánál/feloldásánál a tárolóhely **megadásával** a megadott görbe-táblázatok csak a megadott tárolóban lesznek zárolva/feloldva. A többiek nem lesznek zárolva/feloldva.
	"SRAM" — zárolás beállítása/feloldása a **statikus** NC-tárolóban
	"DRAM" — zárolás beállítása/feloldása a **dinamikus** NC-tárolóban

Ha a CTABLOCK/CTABUNLOCK a zárolandó/feloldandó görbe-táblázat megadása nélkül van programozva, akkor az **összes** görbe-táblázat a megadott tárolóban zárolva/feloldva lesz.

CTABLOCK():	Zárolja az összes görbe-táblázatot a statikus és a dinamikus NC-tárolóban.
CTABLOCK(,,"SRAM"):	Zárolja az összes görbe-táblázatot a statikus NC-tárolóban.
CTABLOCK(,,"DRAM"):	Zárolja az összes görbe-táblázatot a dinamikus NC-tárolóban.
CTABUNLOCK():	Feloldja az összes görbe-táblázatot a statikus és a dinamikus NC-tárolóban.
CTABUNLOCK(,,"SRAM"):	Feloldja az összes görbe-táblázatot a statikus NC-tárolóban.
CTABUNLOCK(,,"DRAM"):	Feloldja az összes görbe-táblázatot a dinamikus NC-tárolóban.

13.2.5 Görbe-táblázatok: Táblázat tulajdonságok megállapítása (CTABID, CTABISLOCK, CTABMEMTYP, CTABPERIOD)

Ezekkel az utasításokkal egy görbe-táblázat fontos tulajdonságait (táblázat szám, zárolás állapot, tárolóhely, periodicitás) lehet lekérdezni.

Szintaxis

```
CTABID(<p>)
CTABID(<p>,<tárolóhely>)
CTABISLOCK(<n>)
CTABMEMTYP(<n>)
TABPERIOD(<n>)
```

Jelentés

CTABID:	visszaadja **táblázat számot**, amelyik a megadott tárolóban a <p>-edik görbe-táblázatként van bevive Példa: CTABID(1,"SRAM") az első görbe-táblázat számát adja meg a statikus NC-tárolóban. Az első görbe-táblázat megfelel a legnagyobb táblázat számú görbe-táblázatnak. **Utalás:** Ha a CTABID két egymás utáni felhívása között a görbe-táblázatok sorrendje meg lett változtatva, pl. a görbe-táblázatok törlése által CTABDEL-lel, a CTABID(<p>,...) ugyanazzal a <p> számmal egy másik görbe-táblázatot adhat vissza, mint előtte.	
CTABISLOCK:	visszaadja **zárolás állapotot** az <n> számú görbe-táblázatra:	
	0	táblázat nincs zárolva
	1	táblázat zárolva van CTABLOCK-kal
	2	táblázat zárolva van aktív csatolás miatt
	3	táblázat zárolva van CTABLOCK és aktív csatolás miatt
	-1	táblázat nem létezik
CTABMEMTYP:	visszaadja **tárolóhelyet** az <n> számú görbe-táblázatra:	
	0	táblázat a statikus NC-tárolóban
	1	táblázat a dinamikus NC-tárolóban
	-1	táblázat nem létezik
CTABPERIOD:	visszaadja **periodicitást** az <n> számú görbe-táblázatra:	
	0	táblázat nem periodikus
	1	táblázat periodikus a vezető tengelyen
	2	táblázat periodikus a vezető tengelyen és a követő tengelyen
	-1	táblázat nem létezik
<p>:	beviteli szám a tárolóban	
<n>:	görbe-táblázat száma (**ID**)	
<tárolási hely>:	tárolóhely megadása (opcionális)	
	"SRAM"	**statikus** NC-tároló
	"DRAM"	**dinamikus** NC-tároló
	Utalás: Ha erre a paraméterre nincs érték programozva, az MD20905 $MC_CTAB_DEFAULT_MEMORY_TYPE gépadatban beállított tárolóhely lesz alkalmazva	

13.2.6 Görbe-táblázat értékeinek olvasása (CTABTSV, CTABTEV, CTABTSP, CTABTEP, CTABSSV, CTABSEV, CTAB, CTABINV, CTABTMIN, CTABTMAX)

A következő görbe-táblázati értékeket lehet olvasni a munkadarabprogramban:

- követő és vezető tengely értékek egy görbe-táblázat elején és végén
- követő tengely értékek egy görbe-szegmens elején és végén
- követő tengely érték egy vezető tengely értékhez
- vezető tengely érték egy követő tengely értékhez
- követő tengelyek minimum és maximum értéke
  - a görbe-táblázat teljes definíciós tartományában vagy
  - egy megadott görbe-táblázati tartományban

Szintaxis

```
CTABTSV(<n>,<gradiens>[,<követő tengely>])
CTABTEV(<n>,<gradiens>[,<követő tengely>])
CTABTSP(<n>,<gradiens>[,<vezető tengely>])
CTABTEP(<n>,<gradiens>[,<vezető tengely>])
CTABSSV(<vezető érték>,<n>,<gradiens>[,<követő tengely>])
CTABSEV(<vezető érték>,<n>,<gradiens>[,<követő tengely>])
CTAB(<vezető érték>,<n>,<gradiens>[,<követő tengely>,<vezető
tengely>]
CTABINV(<követő érték>,<közelítő érték>,<n>,<gradiens>[,<követő
tengely>,<vezető tengely>]
CTABTMIN(<n>[,<követő tengely>])
CTABTMAX(<n>[,<követő tengely>])
CTABTMIN(<n>,<a>,<b>[,<követő tengely>,<vezető tengely>])
CTABTMAX(<n>,<a>,<b>[,<követő tengely>,<vezető tengely>])
```

Jelentés

CTABTSV:	követő tengely értéket olvasni az **elején** az <n> számú görbe-táblázatnak
CTABTEV:	követő tengely értéket olvasni a **végén** az <n> számú görbe-táblázatnak
CTABTSP:	vezető tengely értéket olvasni az **elején** az <n> számú görbe-táblázatnak
CTABTEP:	vezető tengely értéket olvasni az **végén** az <n> számú görbe-táblázatnak
CTABSSV:	követő tengely értéket olvasni az **elején** a megadott vezető tengely értékhez (<vezető érték>) tartozó görbe-szegmensnek
CTABSEV:	követő tengely értéket olvasni az **végén** a megadott vezető tengely érték-hez (<vezető érték>) tartozó görbe-szegmensnek
CTAB:	követő tengely értéket olvasni a megadott vezető tengely értékhez (<vezető érték>)
CTABINV:	vezető tengely értéket olvasni a megadott követő tengely értékhez (<követő érték>)

CTABTMIN:	követő tengely **minimum értékét** meghatározni: • a görbe-táblázat teljes definíciós tartományában vagy • egy megadott <a> ... <b> intervallumban
CTABTMAX:	követő tengely **maximum értékét** meghatározni: • a görbe-táblázat teljes definíciós tartományában vagy • egy megadott <a> ... <b> intervallumban
<n>:	görbe-táblázat száma (**ID**)
<gradiens>:	a <gradiens> paraméter adja vissza a görbe-táblázat funkció **meredekségét** a megállapított pozíciónál
<követő tengely>:	tengely, amelynek mozgását a görbe-táblázat alapján kell kiszámítani (opcionális)
<vezető tengely>:	tengely, amely a vezető értéket adja a követő tengely mozgásának kiszámításához (opcionális)
<követő érték>:	követő tengely érték a hozzátartozó vezető tengely érték olvasásához CTABINV-nél
<vezető érték>:	vezető tengely érték: • a hozzátartozó követő tengely érték olvasásához CTAB-nél vagy • a görbe-szegmens kiválasztásához CTABSSV/CTABSEV-nél
<közelítő érték>:	Egy vezető tengely érték hozzárendelése egy követő tengely értékhez CTABINV-nél nem kell mindig egyértelmű legyen. A CTABINV-nek szüksége van paraméterként egy közelítő értékre az elvárt vezető tengely értékhez.
<a>:	**alsó** határa a vezető érték tartománynak CTABTMIN/CTABTMAX-nál
<b>:	**felső** határa a vezető érték tartománynak CTABTMIN/CTABTMAX-nál
	Utalás: Az <a> ... <b> vezető érték tartománynak a görbe-táblázat definíciós tartományán belül kell legyen..

Példák

Példa 1:

Követő tengely és vezető tengely értékek a görbe-táblázat elején és végén valamint a követő tengely minimum és maximum értékeinek meghatározása a görbe-táblázat teljes definíciós tartományában

Programkód	Kommentár
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	

```
Programkód                                Kommentár
...
N100 CTABDEF(Y,X,1,0)                     ; görbe-táblázat definíció kezdete
N110 X0 Y10                               ; 1. táblázat-szegmens kezdő pozíció
N120 X30 Y40                              ; 1. táblázat-szegmens vég pozíció =
                                          2. táblázat-szegmens kezdő pozíció
N130 X60 Y5                               ; 2. táblázat-szegmens vég pozíció = ...
N140 X70 Y30
N150 X80 Y20
N160 CTABEND                              ; táblázat definíció vége.
...
N200 STARTPOS=CTABTSV(1,GRADIENT)         ; követő tengely érték görbe-táblázat elején
                                          = 10
N210 ENDPOS=CTABTEV(1,GRADIENT)           ; követő tengely érték görbe-táblázat végén=
                                          20
N220 STARTPARA=CTABTSP(1,GRADIENT)        ; vezető tengely érték görbe-táblázat elején
                                          = 0
N230 ENDPARA=CTABTEP(1,GRADIENT)          ; vezető tengely érték görbe-táblázat végén =
                                          80
N240 MINVAL=CTABTMIN(1)                   ; követő tengely minimum értéke Y=5-nél
N250 MAXVAL=CTABTMAX(1)                   ; követő tengely maximum értéke Y=40-nél
```

Példa 2:

Követő tengely értékek meghatározása X = 30 vezető értékhez tartozó görbe-szegmens elején és végén

```
Programkód                                Kommentár
N10 DEF REAL STARTPOS
N20 DEF REAL ENDPOS
N30 DEF REAL GRADIENT
...
N100 CTABDEF(Y,X,1,0)                     ; táblázat definíció kezdete
N110 X0 Y0                                ; 1. táblázat-szegmens kezdő pozíció
N120 X20 Y10                              ; 1. táblázat-szegmens vég pozíció =
                                          2. táblázat-szegmens kezdő pozíció
N130 X40 Y40                              ; 2. táblázat-szegmens vég pozíció = ...
N140 X60 Y10
N150 X80 Y0
N160 CTABEND                              ; táblázat definíció vége.
...
N200 STARTPOS=CTABSSV(30.0,1,GRADIENT)    ; Y kezdő pozíció a 2. szegmensben = 10
N210 ENDPOS=CTABSEV(30.0,1,GRADIENT)      ; Y vég pozíció a 2. szegmensben = 40
```

További információk

Alkalmazás szinkronakciókban

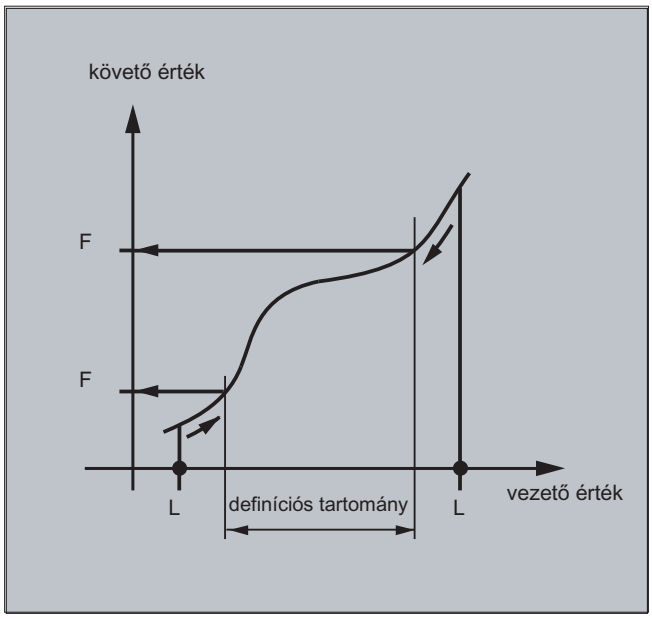
Az összes utasítás a görbe-táblázatok értékeinek olvasására használható a szinkronakciókban is (lásd a "Mozgásszinkron-akciók" fejezetet).

A CTABINV, CTABTMIN és CTABTMAX utasítások használatánál figyelni kell arra, hogy:

- a végrehajtás időpontjában legyen elegendő NC teljesítmény
  vagy
- a felhívás előtt le kell kérdezni a görbe-táblázat szegmenseinek számát, hogy esetleg az érintett táblázatot fel lehessen osztani.

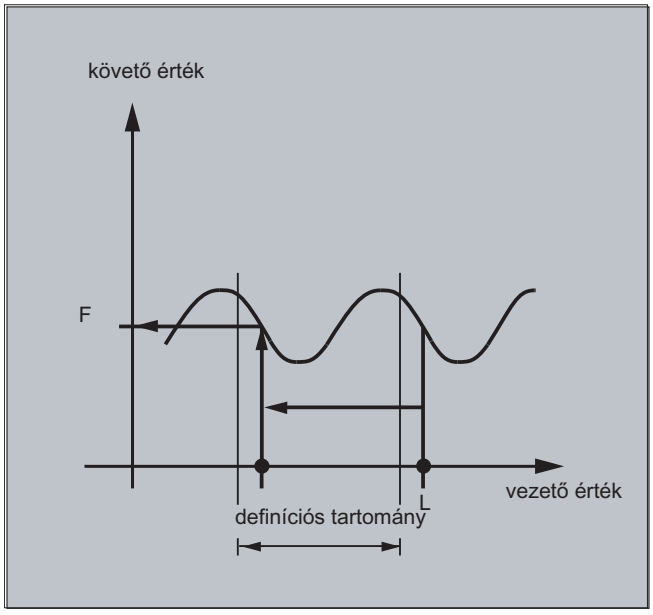
CTAB nem periodikus görbe-táblázatoknál

Ha a megadott `<vezető érték>` a definíciós tartományon kívül van, követő értékként a felső vagy az alsó határ lesz kiadva:



CTAB periodikus görbe-táblázatoknál

Ha a megadott `<vezető érték>` a definíciós tartományon kívül van, a vezető érték a definíciós tartományon modulo lesz kiértékelve és a megfelelő követő érték lesz kiadva:



Közelítő érték CTABINV-nél

A CTABINV utasításnak szüksége van egy közelítő értékre az elvárt vezető értékhez. A CTABINV azt a vezető értéket adja vissza, amelyik a közelítő értékhez legközelebb van. A közelítő érték lehet pl. a vezető érték az előző interpolációs ütemből.

A görbe-táblázat funkció meredeksége

A meredekség (`<gradiens>`) kiadása lehetővé teszi a vezető vagy követő tengely sebességének kiszámítását a megfelelő pozícióban.

Vezető vagy követő tengely megadása

A vezető és/vagy követő tengely opcionális megadása fontos, ha a vezető és a követő tengely különböző hosszegységekre van beállítva.

CTABSSV, CTABSEV

A CTABSSV és CTABSEV utasítások a következő esetekben **nem** alkalmasak**,** programozott szegmensek lekérdezésére:

- körök vagy evolvensek vannak programozva
- `CHF`/`RND` letörés ill. lekerekítés aktív
- `G643` átmenet simítás aktív
- `COMPON`/`COMPCURV`/`COMPCAD` NC kompresszió aktív.

13.2.7 Görbe-táblázatok: Erőforrások használatának vizsgálata (CTABNO, CTABNOMEM, CTABFNO, CTABSEGID, CTABSEG, CTABFSEG, CTABMSEG, CTABPOLID, CTABPOL, CTABFPOL, CTABMPOL)

Ezekkel az utasításokkal a programozónak lehetősége van görbe-táblázatok, táblázat-szegmensek és polinomok erőforrásainak felhasználásáról informálódni.

Szintaxis

```
CTABNO
CTABNOMEM(<tárolóhely>)
CTABFNO(<tárolóhely>)
CTABSEGID(<n>,<tárolóhely>)
CTABSEG(<tárolóhely>,<szegmens fajta>)
CTABFSEG(<tárolóhely>,<szegmens fajta>)
CTABMSEG(<tárolóhely>,<szegmens fajta>)
CTABPOLID(<n>)
CTABPOL(<tárolóhely>)
CTABFPOL(<tárolóhely>)
CTABMPOL(<tárolóhely>)
```

Jelentés

CTABNO:	a **definiált** görbe-táblázatok össz-számát meghatározni (a statikus és dinamikus NC-tárolóban)
CTABNOMEM:	a **definiált** görbe-táblázatok számát a megadott <tárolóhely>-en meghatározni
CTABFNO:	a **még lehetséges** görbe-táblázatok számát a megadott <tárolóhely>-en meghatározni
CTABSEGID:	a megadott <szegmens fajta> görbe-táblázatainak számát meghatározni, amelyeket az <n> számú görbetáblázat használ
CTABSEG:	a **használt** görbe-szegmensek számát a megadott <szegmens fajtá>-ból a megadott <tárolóhely>-en meghatározni
CTABFSEG:	a **még lehetséges** görbe-szegmensek számát a megadott <szegmens fajta>-ból a megadott <tárolóhely>-en meghatározni
CTABMSEG:	a **maximálisan lehetséges** görbe-szegmensek számát a megadott <szegmens fajta>-ból a megadott <tárolóhely>-en meghatározni
CTABPOLID:	a görbe-polinomok számát meghatározni, amelyeket az <n> számú görbe-táblázat használ
CTABPOL:	a **használt** görbe-polinomok számát a megadott <tárolóhely>-en meghatározni
CTABFPOL:	a **még lehetséges** görbe-polinomok számát a megadott <tárolóhely>-en meghatározni
CTABMPOL:	a **maximálisan lehetséges** görbe-polinomok számát a megadott <tárolóhely>-en meghatározni
<n>:	görbe-táblázat száma (**ID**)

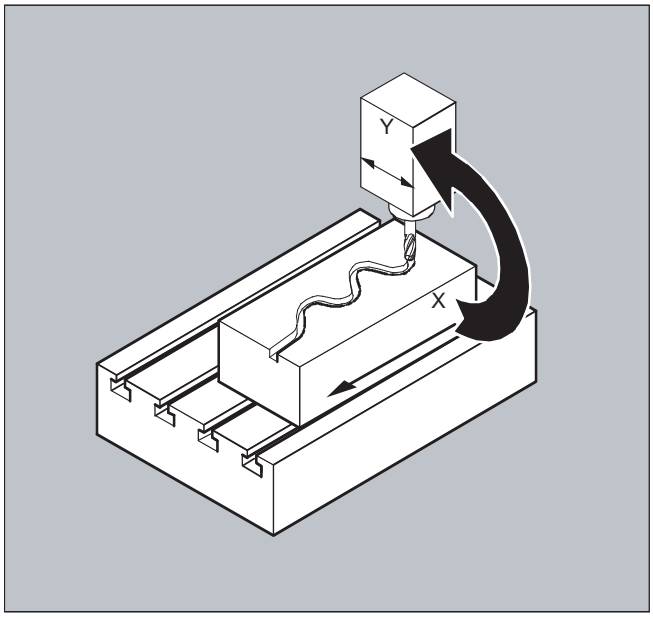
<tárolási hely>:	tárolóhely megadása (opcionális)	
	"SRAM"	**statikus** NC-tároló
	"DRAM"	**dinamikus** NC-tároló
	Utalás: Ha erre a paraméterre nincs érték programozva, az MD20905 $MC_CTAB_DEFAULT_MEMORY_TYPE gépadatban beállított tárolóhely lesz alkalmazva	
<szegmens fajta>:	szegmens fajta megadása (opcionális)	
	"L"	lineáris szegmensek
	"P"	polinom-szegmensek
	Utalás: Ha erre a paraméterre nincs érték programozva, akkor a lineáris és polinom-szegmensek összege lesz kiadva..	

13.3 Tengely vezető érték csatolás (LEADON, LEADOF)

Megjegyzés

Ez a funkció a SINUMERIK 828D-nél nem áll rendelkezésre

A tengely vezető érték csatolásnál egy vezető és egy követő tengely szinkron mozog. Ennek során a követő tengely mindenkori pozíciója egy görbe táblázattal ill. az abból kiszámított polinommal egyértelműen van egy - esetleg szimulált - vezető tengelyhez hozzárendelve.



Vezető tengely az a tengely, amelyik a görbe-táblázat bemenő értékeit adja. **Követő tengely** az a tengely, amelyik a görbe-táblázattal kiszámított pozíciókat felveszi.

Valós- és parancsérték csatolás

Vezető értékként, vagyis kimenő értékként a követő tengely pozíciójának meghatározásához a következőket lehet használni:

- vezető tengely valós pozíció: valósérték csatolás
- vezető tengely parancs pozíció: parancsérték csatolás

A vezető érték csatolás mindig az alap-koordinátarendszerben érvényes.

Görbe-táblázatok létrehozásához lásd a "Görbe-távlázatok" fejezetet.

Szintaxis

```
LEADON(<követő tengely>,<vezető tengely>,<n>)
LEADOF(<követő tengely>,<vezető tengely>)
```

vagy kikapcsolás a vezető tengely megadása nélkül

```
LEADOF(<követő tengely>)
```

A vezető érték csatolást a munkadarabprogramban és szinkronakció mozgás közben is lehet be- és kikapcsolni.

Jelentés

LEADON:	vezető érték csatolás bekapcsolása
LEADOF:	vezető érték csatolás kikapcsolása
<követő tengely>:	követő tengely
<vezető tengely>:	vezető tengely
<n>:	görbe-táblázat száma
$SA_LEAD_TYPE:	átkapcsolás parancs- és valósérték csatolás között

Vezető érték csatolás kikapcsolása, LEADOF

A vezető érték csatolás kikapcsolásával a követő tengely ismét normális parancs-tengely lesz!

Tengely vezető érték csatolás és különböző üzemállapotok, RESET

A gépadat beállításától függően történik a vezető érték csatolások kikapcsolása RESET-tel.

Példa vezető érték csatolásra szinkronakcióból

Egy sajtoló berendezésnél kell a szokásos mechanikus csatolást egy vezető tengely (bélyegző tengely) és egy szállító-tengelyekből és segéd-tengelyekből álló szállítórendszer tengelyei között egy elektronikus csatoló rendszerrel kiváltani.

Bemutatjuk, hogyan lesz egy sajtoló berendezésnél egy mechanikus szállítórendszer egy elektronikus szállítórendszerrel kiváltva. A csatolást létrehozó és lebontó eljárások **statikus szinkronakcióként** vannak megvalósítva.

A vezető tengely LW (bélyegző tengely) vezéreli a szállító-tengelyeket és a segédtengelyeket követő tengelyekként a görbe-táblázatban definiált módon.

Követő-tengelyek

X előtoló- ill. hossz-tengely
YL záró- ill. kereszt-tengely
ZL löket-tengely
U henger-előtolás, segédtengely
V beállító-fej, segédtengely
W zsírozás, segédtengely

Akciók

A szinkronakciókban pl. a következő akciók fordulnak elő:

- becsatolás, `LEADON(<követő tengely>,<vezető tengely>,<görbe-táblázat szám>)`
- kicsatolás, `LEADOF(<követő tengely>,<vezető tengely>)`
- valós érték állítás, `PRESETON(<tengely>,<érték>)`
- jelölő állítás, `$AC_MARKER[i] =<érték>`

- csatolási mód: reális/virtuális vezető érték
- tengely-pozíciók felvétele, `POS[<tengely>]=<érték>`

Feltételek

Feltételként gyors, digitális bemenetek, `$AC_MARKER` valósidejű változók és pozíció-összehasonlítások lesznek kiértékelve, az AND logikai művelettel összekapcsolva.

Megjegyzés

IA következő példában a sorváltás, behúzás és **kövér** szedés kizárólag a program olvashatóságának javítása céljából van alkalmazva. A vezérlés számára az egy sorszám után álló rész számít egy sornak.

Kommentár

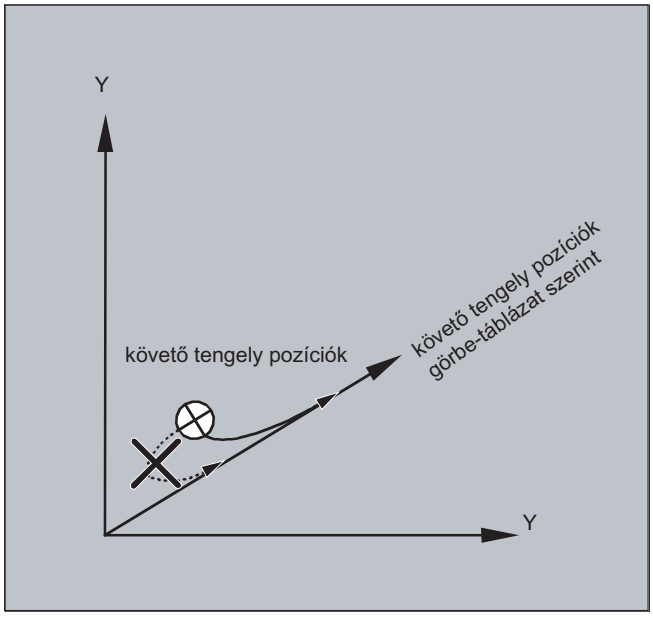
Programkód	Kommentár
	; Az összes statikus szinkronakció definíciója.
	; **** Jelölő visszaállítás
N2 $AC_MARKER[0]=0 $AC_MARKER[1]=0 $AC_MARKER[2]=0 $AC_MARKER[3]=0 $AC_MARKER[4]=0 $AC_MARKER[5]=0 $AC_MARKER[6]=0 $AC_MARKER[7]=0	
	; **** E1 0=>1 szállítás csatolás BE
N10 IDS=1 EVERY ($A_IN[1]==1) AND ($A_IN[16]==1) AND ($AC_MARKER[0]==0) DO LEADON(X,LW,1) LEADON(YL,LW,2) LEADON(ZL,LW,3) $AC_MARKER[0]=1	
	; **** E1 0=>1 henger-előtolás csatolás BE
N20 IDS=11 EVERY ($A_IN[1]==1) AND ($A_IN[5]==0) AND ($AC_MARKER[5]==0) DO LEADON(U,LW,4) PRESETON(U,0) $AC_MARKER[5]=1	
	; **** E1 0>1 beállító-fej csatolás BE
N21 IDS=12 EVERY ($A_IN[1]==1) AND ($A_IN[5]==0) AND ($AC_MARKER[6]==0) DO LEADON(V,LW,4) PRESETON(V,0) $AC_MARKER[6]=1	
	; **** E1 0->1 zsírozás csatolás BE
N22 IDS=13 EVERY ($A_IN[1]==1) AND ($A_IN[5]==0) AND ($AC_MARKER[7]==0) DO LEADON(W,LW,4) PRESETON(W,0) $AC_MARKER[7]=1	
	; **** E2 0=>1 csatolás KI
N30 IDS=3 EVERY ($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	

Leírás

A vezető érték csatolás igényli a vezető és a követő tengely szinkronizációját. Ezt a szinkronizációt csak akkor lehet elérni, ha a követő tengely a vezető érték csatolás bekapcsolásakor a görbe-táblázatból kiszámított görbe tűréssávján belül van.

A követő tengely helyzetének tűréstartománya az `MD 37200: COUPLE_POS_POL_COARSE_A_LEAD_TYPE` gépadatban van definiálva.

Ha a követő tengely a vezető érték csatolás bekapcsolásakor még nincs a megfelelő helyzetben, a szinkronfutás automatikusan akkor jön létre, amikor követő tengelyre kiszámított pozíció érték a tényleges követő tengely pozícióhoz közelít. A követő tengely a szinkronizációs folyamat közben a követő tengely parancs-sebessége (számítás a vezető tengely sebesség és CTAB alapján) által meghatározott irányba mozog.

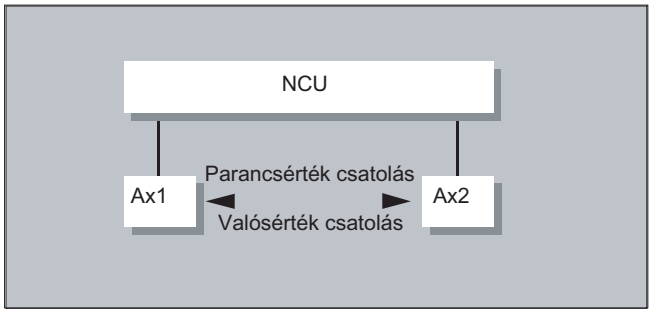


Nincs szinkronfutás

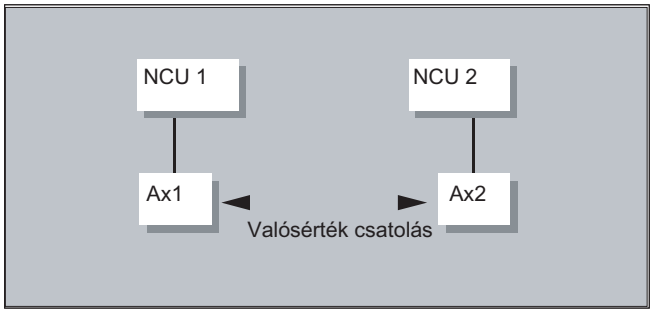
Ha a kiszámított követő tengely pozíció eltávolodik a vezető érték csatolás bekapcsolásával az aktuális követő tengely pozíciótól, a szinkronfutás nem jön létre.

Valós- és parancsérték csatolás

A parancsérték csatolás a valósérték csatolással összehasonlítva jobb szinkronfutást biztosít a vezető és a követő tengely között és ezért alaphelyzetként ez van beállítva.



A parancsérték csatolás csak akkor lehetséges, ha a vezető és a követő tengelyt ugyanaz az NCU interpolálja. Egy külső vezető tengelynél a követő tengelyt csak a valósértékkel lehet a vezető tengelyhez csatolni.



Az **átkapcsolás** $SA_LEAD_TYPE beállítási adattal lehetséges.

Az átkapcsolás a valós- és a parancsérték csatolás között mindig a követő tengely nyugalmi állapotában történjen. Mivel csak a nyugalmi helyzetben történik meg újra a szinkronizáció az átkapcsolás után.

Alkalmazási példa

A valósérték olvasása a gép erős rázkódása mellett nem lehet hibamentes. A sajtoló továbbító berendezéseknél a vezető érték csatolás esetén szükséges lehet a jelentős rázkódással járó munkalépésekben a valósérték csatolásról átkapcsolni a parancsérték csatolásra.

Vezető érték szimuláció a parancsérték csatolásnál

Gépadat segítségével az interpolátort a vezető tengelynél le lehet választani a szervóról. Ezzel a parancsérték csatolásnál a parancsértéket a vezető tengely tényleges mozgása nélkül csatolni lehet.

A parancsérték csatolással létrehozott vezető értékek olvashatók pl. a szinkronakciókból a következő változókkal:

- $AA_LEAD_P	vezető érték pozíció
- $AA_LEAD_V	vezető érték sebesség

Vezetőértékek létrehozása

Vezető értékeket választhatóan lehet más, saját programozású módszerrel is létrehozni. Az így létrehozott vezető értéket a

- $AA_LEAD_SP	vezető érték pozíció
- $AA_LEAD_SV	vezető érték sebesség

változókba lehet beírni és onnan kiolvasni. Ezen változók használatához a $SA_LEAD_TYPE = 2 beállítási adatot be kell állítani.

A csatolás állapota

Az NC munkadarab-programban a csatolás állapotát a következő rendszerváltozókkal lehet lekérdezni:

```
$AA_COUP_ACT[tengely]
```
0: nincs aktív csatolás
16: vezető érték csatolás aktív

Állapot kezelés szinkronakcióknál

A kapcsolási és csatolási folyamatok a

`$AC_MARKER[i] = n`
valós idejű változóval lesznek kezelve, ahol:
i jelölő szám
n állapotérték

13.4 Elektronikus hajtómű (EG)

Az "Elektronikus hajtómű" funkcióval lehetséges egy **követő-tengely** mozgását lineáris mozgás-mondatban max. öt **vezető-tengelytől** függően vezérelni. A vezető-tengelyek és a követő-tengely összefüggéseit vezető-tengelyekként egy csatolás-tényezővel lehet definiálni.

A kiszámított követő-tengely mozgásrész az egyes vezető-tengely mozgásrészek és megfelelő csatolás-tényezők szorzatainak összegéből képződik. Egy EG tengely-egyesülés aktiválásánál lehet kezdeményezni a követő-tengely szinkronizációját egy definiált pozícióba. Egy hajtómű egyesülést a munkadarabprogramból lehet:

- definiálni,
- bekapcsolni,
- kikapcsolni,
- törölni

lehet.

A követő-tengely mozgását választhatóan le lehet vezetni

- a vezető-tengelyek parancsértékeiből ill.
- a vezető-tengelyek valósértékeiből.

Bővítésként lehetséges a vezető-tengelyek és a követő-tengely között nem-lineáris összefüggéseket is megvalósítani **görbe-táblázatokkal** (lásd a Pályaviselkedés fejezetben). Az elektronikus hajtóműveket sorba lehet kapcsolni, azaz egy elektronikus hajtómű követő tengelye lehet egy következő elektronikus hajtómű vezető tengelye.

13.4.1 Elektronikus hajtómű definíció (EGDEF)

Egy EK tengely-egyesülést a követő tengely és legalább egy, de maximum öt vezető tengely és a megfelelő csatolás típusok megadásával lehet megadni.

Előfeltétel

Egy EG tengely-egyesülés definíciójának előfeltétele:

A követő-tengelyre nem lehet még tengely-csatolás definiálva (esetleg egy már meglevőt előbb EGDEL-lel törölni kell).

Szintaxis

```
EGDEF(követő tengely,vezető tengely 1,csatolás típus 1,vezető
tengely 2,csatolás típus 2,...)
```

Jelentés

`EGDEF`:	elektronikus hajtómű definíció
`követő tengely`:	tengely, amelyet a vezető-tengelyek befolyásolnak

`vezető tengely 1`
`,...,`
`vezető tengely 5`
	tengelyek, amelyek követő-tengelyt befolyásolnak	

`csatolás típus 1`
`,...,`
`csatolás típus 5`
	csatolás típus
	A csatolás típus nem kell azonos legyen minden vezető tengelyre és ezért egyenként kell megadni minden vezető tengelyre.	

	Érték:	**Jelentés:**
	0	követő-tengely befolyásolása a **valósértéke** megfelelő vezető-tengelynek
	1	követő-tengely befolyásolása a **parancsértéke** megfelelő vezető-tengelynek

Megjegyzés

A csatolási tényezők az EG csatolási egyesülés definíciójánál nullával lesznek feltöltve.

Megjegyzés

EGDEF előrefutás-álljt vált ki. A hajtómű definíciót EGDEF -fel akkor is változatlanul kell, ha a rendszerben egy vagy több vezető-tengely hat a követő-tengelyre **görbe-táblázattal**.

Példa

Programkód	Kommentár
`EGDEF(C,B,1,Z,1,Y,1)`	`; Egy EK tengely-egyesülés definíciója. A B, Z, Y vezető tengelyek befolyásolják a C követő tengelyt parancsértékkel.`

13.4.2 Elektronikus hajtóművet bekapcsolni (EGON, EGONSYN, EGONSYNE)

Egy EK tengely-egyesülés bekapcsolására 3 változat van.

Szintaxis

Változat 1:

Az EG tengely-egyesülés szinkronizáció nélkül szelektív bekapcsolása:
`EGON(FA,"mondatváltás-módus",LA1,Z1,N1,LA2,Z2,N2,...,LA5,Z5,N5)`

Változat 2:

Az EG tengely-egyesülés szinkronizációval szelektív bekapcsolása:
`EGONSYN(FA,"mondatváltás-módus",SynPosFA,[,LAi,SynPosLAi,Zi,Ni])`

Változat 3:

Az EG tengely-egyesülés szinkronizációval szelektív bekapcsolása és a rámeneti modus megadása:

```
EGONSYNE(FA,"mondatváltás-módus",SynPosFA,rámeneti
módus[,LAi,SynPosLAi,Zi,Ni])
```

Jelentés

Változat 1:

FA	követő tengely	
mondatváltás-módus:	következő módusokat lehet kiválasztani:	
	"NOC"	mondatváltás azonnal
	"FINE"	mondatváltás "szinkronfutás finom"-nál
	"COARSE"	mondatváltás "szinkronfutás durva"-nál
	"IPOSTOP"	mondatváltás parancsérték szinkronfutásnál
LA1, ... LA5	vezető tengelyek	
Z1, ... Z5	csatolási tényező i számláló	
N1, ... N5	csatolási tényező i nevező csatolási tényező i = számláló i / nevező i	

Csak azokat a vezető-tengelyeket szabad programozni, amelyek előbb EGDEF-fel specifikálva lettek. Legalább egy vezető-tengelyt kell programozni.

Változat 2:

FA	követő tengely	
mondatváltás-módus:	következő módusokat lehet kiválasztani:	
	"NOC"	mondatváltás azonnal
	"FINE"	mondatváltás "szinkronfutás finom"-nál
	"COARSE"	mondatváltás "szinkronfutás durva"-nál
	"IPOSTOP"	mondatváltás parancsérték szinkronfutásnál
[,LAi,SynPosLAi,Zi,Ni]	(ne írjunk szögletes zárójelet) min. 1, max. 5 egymás után:	
LA1, ... LA5	vezető tengelyek	
SynPosLAi	szinkronpozíció az i. vezető-tengelyre	
Z1, ... Z5	csatolási tényező i számláló	
N1, ... N5	csatolási tényező i nevező csatolási tényező i = számláló i / nevező i	

Csak azokat a vezető-tengelyeket szabad programozni, amelyek előbb EGDEF-fel specifikálva lettek. A követő-tengelyre (SynPosFA) és a vezető-tengelyekre (SynPosLA) programozott "szinkronpozíciók" olyan pozíciókat definiálnak, amelyekre a csatolási-egyesülés *szinkron*. Ha az elektronikus hajtómű a bekapcsolásnál nincs szinkron állapotban, a követő-tengely a definiált szinkronpozíciójába megy.

Változat 3:

A paraméterek megfelelnek a változat 2-nek, továbbá:

`rámenet módus:`	következő módusokat lehet kiválasztani:	
	`"NTGT"`	következő fogközre idő-optimáltan menni
	`"NTGP"`	következő fogközre út-optimáltan menni
	`"ACN"`	körtengelyt negatív forgásirányban abszolút mozgatni
	`"ACP"`	körtengelyt pozitív forgásirányban abszolút mozgatni
	`"DCT"`	idő-optimalizáltan a programozott szinkronpozícióra
	`"DCP"`	út-optimalizáltan a programozott szinkronpozícióra

A változat 3-nak csak a modulo követő tengelyekre van hatása, amelyek a modulo vezető tengelyekhez csatolva vannak. Az idő-optimalizálás figyelembe veszi a követő tengely sebességhatárait.

További információk

Bekapcsolási változatok leírása

Változat 1:

A vezető-tengelyek és a követő-tengely pozíciói a bekapcsolás időpontjában "Szinkron-pozíció"-ként lesznek tárolva. A "Szinkron-pozíció"-kat a `$AA_EG_SYN` rendszerváltozóval lehet olvasni.

Változat 2:

Ha a csatolási-egyesülésben vannak modulo-tengelyek, a pozíció értékeik modulo csökkentve lesznek. Ezzel garantált, hogy ezek a lehető legközelebbi szinkronpozícióra mennek (un. *relatív szinkronizáció*: pl. következő fogrés). Ha a követő-tengelynek "Követő- tengely átlapolás engedélyezés" interfészjel DB(30 +tengelyszám), DBX26 bit 4 nincs beállítva, nem fog a szinkronpozícióra menni. Ehelyett a program az EGONSYN-t tartalmazó mondatnál megáll és 16771 öntörlő vészjelzés jelentkezik addig, amíg a fenti jel be nem lesz állítva.

Változat 3:

A fogtávolság (fok) a következő: 360 * Zi/Ni. Abban az esetben, ha a követő tengely a felhívás időpontjában áll, az út-optimált és az idő-optimált viselkedés azonos.

A már mozgó követő-tengely az NTGP-vel függetlenül a követő-tengely aktuális sebességétől a következő fogközre fog szinkronizálni. A már mozgó követő-tengely az NTGT-vel függetlenül a követő-tengely aktuális sebességétől a következő fogközre fog szinkronizálni. A tengely ehhez adott esetben le is lesz fékezve.

Görbe-táblázatok

Ha egy vezető-tengelyhez egy **görbe-táblázat** lesz használva, akkor:

Ni a lineáris csatolások csatolási tényezőjének a nevezője 0 kell legyen. (0 nevező lineáris csatolásokra nem lenne megengedett). A 0 nevező jelzi a vezérlésnek, hogy

Zi az alkalmazott görbe-táblázat számaként értelmezendő. A megadott számú görbe-táblázat a bekapcsolás időpontjában már definiálva kell legyen.

LAi A vezető tengely megadása megfelel a vezető tengely megadásának a csatolási tényezővel való csatolásnál (lineáris csatolás).

További utalások a görbe-táblázatok csatolásához és az elektronikus hajtóművek sorba-kapcsolásához és azok szinkronizálásához található:
Irodalom:
Funktionshandbuch Sonderfunktionen; Achskopplungen und ESR (M3), Kapitel "Mitschleppen und Leitwertkopplung".

Elektronikus hajtómű viselkedése Power On, RESET, üzemmód-váltás, mondatkeresés esetén

- Power On után **nincs** aktív csatolás.
- Aktív csatolások megmaradnak RESET és üzemmód-váltás után.
- Keresésénél az elektronikus hajtóműre vonatkozó kapcsolás, törlés és definiálás utasítások nem lesznek végrehajtva vagy összegyűjtve, hanem ki lesznek hagyva.

Elektronikus hajtómű rendszerváltozói

Az elektronikus hajtómű rendszerváltozói segítségével a munkadarabprogram meg tudja állapítani egy EG tengely-egyesülés állapotát és esetleg tud arra reagálni.

Az elektronikus hajtómű rendszerváltozóit a függelékben találjuk.

$AA_EG_ ...

vagy

$VA_EG_ ...

Irodalom:
Handbuch der Systemvariablen

13.4.3 Elektronikus hajtómű kikapcsolása (EGOFS)

Egy aktív EK tengely-egyesülés kikapcsolására 3 változat van.

Programozás

Változat 1:

Szintaxis	Jelentés
`EGOFS(követő tengely)`	Az elektronikus hajtómű kikapcsol. A követő tengely megállásig fékezése.. A hívás előrefutás-álljt vált ki.

Változat 2:

Szintaxis	Jelentés
`EGOFS(követő-tengely,vezető-tengely1, …,vezető-tengely5)`	Ennek az utasításnak a paraméterezése lehetővé teszi az egyes vezető tengelyeknek a követő tengely mozgására gyakorolt hatását **szelektív** kikapcsolni.

Legalább egy vezető tengelyt meg kell adni. A megadott vezető tengelyek hatása a követő tengelyre célzottan lesz kikapcsolva. A hívás előrefutás-álljt vált ki. Ha még maradnak aktív

vezető tengelyek, a követő tengely ennek befolyásával fut tovább. Ha az összes vezető tengely befolyás ezen a módon ki lesz kapcsolva, a követő tengely megállásig lesz lefékezve.

Változat 3:

Szintaxis	Jelentés
`EGOFC(követő orsó 1)`	Az elektronikus hajtómű kikapcsol. A követő orsó a kikapcsolási időpontban aktuális fordulatszámmal/sebességgel fut tovább. A hívás előrefutás-álljt vált ki.

Megjegyzés

Ez a funkció csak az orsókra megengedett.

13.4.4 Egy elektronikus hajtómű definícióját törölni (EGDEL)

Egy EK tengely-egyesülést a definíciójának törlése előtt ki kell kapcsolni.

Programozás

Szintaxis	Jelentés
`EGDEL(követő tengely)`	A tengely-egyesülés csatolás-definíciójának törlése. Az egyidejűleg aktivált tengely-egyesülések maximális számának eléréséig lehetséges további tengely-szövetségek új definíciója `EGDEF`-fel. A hívás előrefutás-álljt vált ki.

13.4.5 Fordulat-előtolás (G95) / Elektronikus hajtómű (FPR)

Az FPR utasítással egy elektronikus hajtómű követő-tengelyét is meg lehet adni a fordulat-előtolás előtolás-meghatározó tengelyeként. Erre az esetre a következő viselkedés érvényes:

- Az előtolás az elektronikus hajtómű követő-tengelye parancs-sebességétől függ.
- A parancssebesség a vezető orsó és modulo vezető tengelyek (amelyek nem pályatengelyek) sebességeiből és az ezekhez hozzárendelt csatolási tényezőkből lesz kiszámítva.
- A lineáris ill. nem modulo vezető tengelyek és követő tengely átlapolt mozgásainak sebességrészei nem lesznek figyelembe véve.

13.5 Szinkron-orsó

IA szinkron-üzemben van egy vezető-orsó (LS) és egy követő-orsó (FS), az úgynevezett **szinkronorsó pár**. A követő-orsó aktív csatolásnál (szinkron-üzem) követi a vezető-orsó mozgásait a megadott funkció összefüggésnek megfelelően..

A szinkronorsó-párokat a gépekre lehet a csatorna-specifikus gépadatokkal fixen beállítani vagy a CNC munkadarabprogrammal alkalmazás-specifikusan definiálni. NC csatornánként maximum 2 szinkronorsó-pár használható egyidejűleg.

A csatolást a munkadarabprogramból lehet:

- definiálni ill. megváltoztatni
- bekapcsolni
- kikapcsolni
- törölni

lehet.

Ezen kívül a szoftver-kiadástól függően lehet

- szinkronfutás feltételre várni
- mondatváltási viselkedést megváltoztatni
- a csatolási módot vagy parancsérték csatolásként vagy valósérték csatolásként vagy szögeltolásként a vezető- és követő-orsó között megadni
- bekapcsolásnál a követő-orsó megelőző programozásából a csatolást átvenni
- egy már megmért vagy ismert szinkronfutás eltérést korrigálni

lehet.

13.5.1 Szinkronorsó: Programozás (COUPDEF, COUPDEL, COUPON, COUPONC, COUPOF, COUPOFS, COUPRES, WAITC)

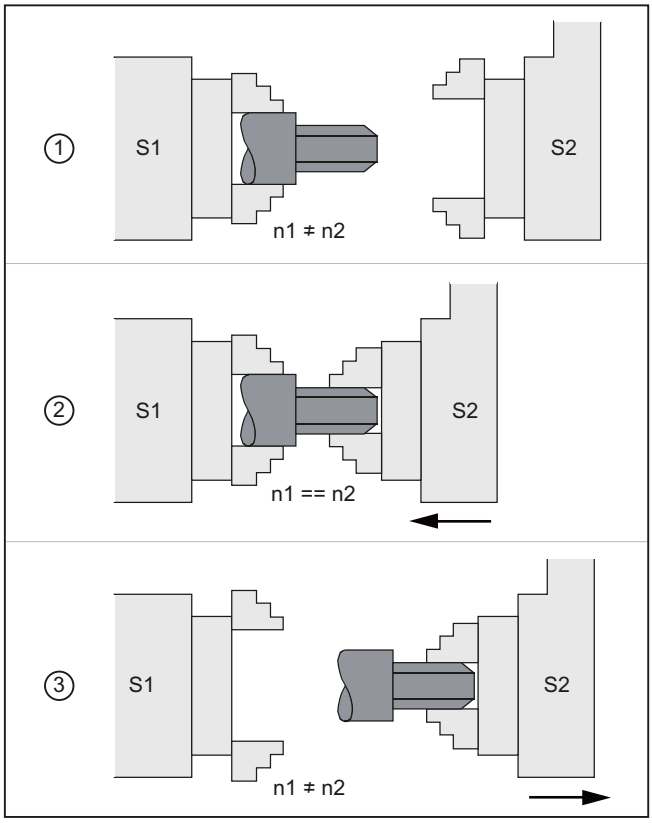
A "Szinkron-orsó" funkció lehetővé teszi a követő-orsó (FS) és vezető-orsó (LS) szinkron mozgását egy programozható áttétel aránnyal.

A funkció a következő modusokat nyújtja:

- fordulatszám-szinkronitás ($n_{FS} = n_{LS}$)
- helyzet-szinkronitás ($\phi_{FS} = \phi_{LS}$)
- helyzet-szinkronitás szög-eltolással ($\phi_{FS} = \phi_{LS} + \Delta\phi$)

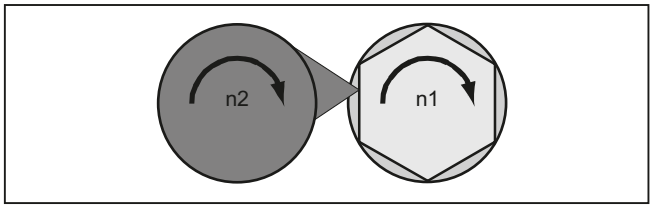
Alkalmazási példák:

- Munkadarab repülő átadása pl. hátoldal megmunkálásához, áttételi aránnyal: 1:1



① fordulatszámot szinkronizálni
② munkadarabot átadni
③ hátoldalt megmunkálni

- Sokszög megmunkálása (poligon esztergálás), fordulatszám szinkronitás, áttételi arány: n_1:n_2



Szintaxis

```
COUPDEF(<FS>,<LS>,<ÜFS>,<ÜLS>,<mondatváltás>,<csatolásmód>)
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>,<mondatváltás>,<LS>,<csatolásmód>)
```

Megjegyzés

Rövidített írásmód

A COUPOF, COUPOFS, COUPRES és COUPDEL utasításoknál lehetséges a rövidített megadás vezető-orsó nélkül is.

Jelentés

COUPDEF:	Felhasználó-specifikus csatolást definiálni/változtatni
COUPON:	Csatolást bekapcsolni. A követő-orsó az aktuális fordulatszámból kiindulva a vezető-orsóhoz szinkronizálódik
COUPONC:	Csatolás bekapcsolásánál az előzőleg programozott M3 S... vagy M4 S... átvétele. A követő-orsó fordulatszám eltérése azonnal át lesz véve.
COUPOF:	Csatolást kikapcsolni. • azonnali mondatváltással: COUPOF(<S2>,<S1>) • Mondatváltás csak <POSFS> ill. <POSLS> kikapcsolási pozíció(ko)n túlhaladás után: COUPOF(<S2>,<S1>,<POSFS>) COUPOF(<S2>,<S1>,<POSFS>,<POSLS>)
COUPOFS:	Csatolás kikapcsolása, követő-orsó állj. Lehető leggyorsabb mondatváltás azonnali mondatváltással: COUPOFS(<S2>,<S1>) Mondatváltás csak a kikapcsolási pozíción túlhaladás után: COUPOFS(<S2>,<S1>,<POSFS>)
COUPRES:	Csatolási paraméterek visszaállítása a megadott MD és SD-re
COUPDEL:	Felhasználó által definiált csatolás törlése
WAITC:	Várakozás szinkronfutási feltételre (NOC megszüntetése IPO-ra mondatváltásnál)
<FS>:	követő-orsó jelölése
Opcionális paraméterek:	
<LS>:	vezető-orsó jelölése megadás orsószámmal: pl. S2, S1
<ZFS>, <NLS>:	áttételi arány FS és LS között <ZFS> / <NLS> = számláló / nevező elő-beállítás: ZFS> / <NLS> = 1.0 ; nevező megadása opcionális

<mondatváltás>:	mondatváltás viselkedés mondatváltás bekövetkezése:	
	"NOC"	azonnal
	"FINE"	"szinkronfutás finom" eléréséné l
	"COARSE"	"szinkronfutás durva" eléréséné l
	"IPOSTOP"	IPOSTOP elérésével, azaz parancsérték oldali szinkronfutás után (előbeállítás)
	A mondatváltási viselkedés modálisan hat.	
<csatolási mód>:	csatolási mód: csatolás FS és LS között	
	"DV"	parancsérték-csatolás (előbeállítás)
	"AV"	valósérték csatolás
	"VV"	sebesség-csatolás
	A csatolási mód modálisan hatásos.	
<POSFS>:	szögeltolás a vezető- és követő-orsó között	
	értéktartomány:	0°… 359,999°
<POSFS>,<POSLS>:	követő- és vezető-orsó kikapcsolási pozíciók "A mondatváltás a POS_{FS}, POS_{LS} túlhaladása után lesz engedélyezve"	
	értéktartomány:	0°… 359,999°

Példák

Munka vezető- és követő-orsóval

Programkód	Kommentár
	vezető-orsó = mester-orsó = orsó 1
	követő-orsó = orsó 2
N05 M3 S3000 M2=4 S2=500	vezető-orsó forgás 3000 ford/perc, követő-orsó forgás 500 ford/perc
N10 COUPDEF(S2,S1,1,1,"NOC","Dv")	csatolás definíciója (lehet megtervezett is)
...	
N70 SPCON	vezető-orsót helyzetszabályozásba venni (parancsérték-csat.)
N75 SPCON(2)	követő-orsót helyzetszabályozásba venni
N80 COUPON(S2,S1,45)	csatolás röptében offsetpozíció = 45 fokra
...	
N200 FA[S2]=100	pozícionáló-sebesség = 100 fok/perc
N205 SPOS[2]=IC(-90)	90 fokot átlapolva negatív irányba menni
N210 WAITC(S2,"Fine")	várakozás szinkronfutás "finom"-ra
N212 G1 X... Y... F...	megmunkálás
...	
N215 SPOS[2]=IC(180)	180 fokot átlapolva pozitív irányba menni
N220 G4 S50	várakozási idő = 50 mester-orsó fordulat
N225 FA[S2]=0	beállított sebességet (MD) aktiválni
N230 SPOS[2]=IC(-7200)	20 fordulatot a tervezett sebességgel a negatív irányba menni

Programkód	Kommentár
...	
N350 COUPOF(S2,S1)	röptében csatolást bontani, S=S2=3000
N355 SPOSA[2]=0	FS-t nulla foknál megállítani
N360 G0 X0 Y0	
N365 WAITS(2)	várakozás orsó 2-re
N370 M5	FS-t megállítani
N375 M30	

Fordulatszám-eltérés programozása

Programkód	Kommentár
	vezető-orsó = mester-orsó = orsó 1
	követő-orsó = orsó 2
N01 M3 S500	vezető-orsó forgás 500 ford/perc
N02 M2=3 S2=300	követő-orsó forgás 300 ford/perc
...	
N10 G4 F1	mester-orsó várakozási idő
N15 COUPDEF (S2,S1,-1)	csatolási tényező 1:1-es áttételi viszonnyal
N20 COUPON (S2,S1)	csatolást aktiválni A követő-orsó fordulatszáma a vezető-orsó fordulatszámából és a csatolási tényezőből adódik.
...	
N26 M2=3 S2=100	fordulatszám-eltérés programozása

Példák mozgás átvételre fordulatszám-eltéréssel

1. Csatolás bekapcsolása a követő-orsó előzetes programozásánál COUPON-nal

Programkód	Kommentár
	vezető-orsó = mester-orsó = orsó 1
	követő-orsó = orsó 2
N05 M3 S100 M2=3 S2=200	vezető-orsó forgás 100 ford/perc,követő-orsó forgás 200 ford/perc
N10 G4 F5	várakozási idő = 5 mp. mester-orsóra
N15 COUPDEF(S2,S1,1)	követő-orsó áttételi aránya a vezető-orsóhoz 1,0 (előbeállítás)
N20 COUPON (S2,S1)	csatolás röptében vezető-orsóhoz
N10 G4 F5	követő-orsó forgás 100 ford/perc

2. Csatolás bekapcsolása a követő-orsó előzetes programozásánál COUPONC-vel

Programkód	Kommentár
	vezető-orsó = mester-orsó = orsó 1
	követő-orsó = orsó 2
N05 M3 S100 M2=3 S2=200	vezető-orsó forgás 100 ford/perc, követő-orsó forgás 200 ford/perc
N10 G4 F5	várakozási idő = 5 mp. mester-orsóra

Programkód	Kommentár
N15 COUPDEF(S2,S1,1)	követő-orsó áttételi aránya a vezető-orsóhoz 1,0 (előbeállítás)
N20 COUPONC(S2,S1)	csatolás röptében vezető-orsóhoz és a megadott fordulatszám átvétele S2-be
N10 G4 F5	S2 forgása 100 ford/perc + 200 ford/perc = 300 ford/perc

3. Csatolás bekapcsolása álló követő-orsónál COUPON-nal

Programkód	Kommentár
	vezető-orsó = mester-orsó = orsó 1
	követő-orsó = orsó 2
N05 SPOS=10 SPOS[2]=20	S2 követő-orsó pozícionáló üzemben
N15 COUPDEF(S2,S1,1)	követő-orsó áttételi aránya a vezető-orsóhoz 1,0 (előbeállítás)
N20 COUPON (S2,S1)	csatolás röptében vezető-orsóhoz
N10 G4 F1	Csatolás zárása, S2 20 fokon marad állva.

4. Csatolás bekapcsolása álló követő-orsónál COUPONC-vel

Megjegyzés

Pozícionáló- vagy tengelyüzem

Ha a követő-orsó a csatolás előtt pozícionáló- vagy tengelyüzemben volt, akkor a követő-orsó a COUPON(<FS>,<LS>) és COUPONC(<FS>,<LS>) esetében azonosan viselkedik..

Megjegyzés

Vezető-orsó és tengelyüzem

Ha a vezető-orsó a csatolás definiálása előtt tengelyüzemben volt, akkor a csatolás bekapcsolása után is hat a sebesség-határérték a következő gépadatból:

MD32000 $MA_MAX_AX_VELO (maximális tengelysebesség)

Ennek elkerülésre a tengelyt a csatolás definiálása előtt orsóüzembe (M3 S... vagy M4 S...) kell kapcsolni.

További információk

Beállított csatolás

A beállított csatolásnál a vezető-és követő-orsót gépadatokban állítjuk be. A beállított orsók munkadarabprogramból nem változtathatók. A csatolás paraméterezése COUPDEF-fel a munkadarabprogramban történhet (előfeltétel: nincs írásvédelem megadva).

Felhasználó által definiált csatolás

A COUPDEF utasítással lehet egy csatolást a munkadarabprogramban létrehozni és megváltoztatni. Ha egy csatolás már aktív, az új csatolás definíciója előtt azt COUPDEL-lel törölni kell.

Egy csatolást teljesen definiálnak a következők:

```
COUPDEF(<FS>,<LS>,<ÜFS>,<ÜLS>, mondatváltás viselkedés, csatolási
mód)
```

Követő-orsó (FS) és vezető-orsó (LS)

A követő-orsó (FS) és a vezető-orsó (LS) tengelyneveivel a csatolás egyértelműen meg van határozva. A tengelyneveket a `COUPDEF` utasítással kell programozni. A többi csatolási paramétereket modálisan hatásosak és csak akkor kell ezeket újra programozni, ha meg akarjuk változtatni

Példa:

```
COUPDEF(S2,S1)
```

Áttételi arány

Az áttételi arány az FS követő-orsó és az LS vezető-orsó közötti fordulatszám arányként van megadva:

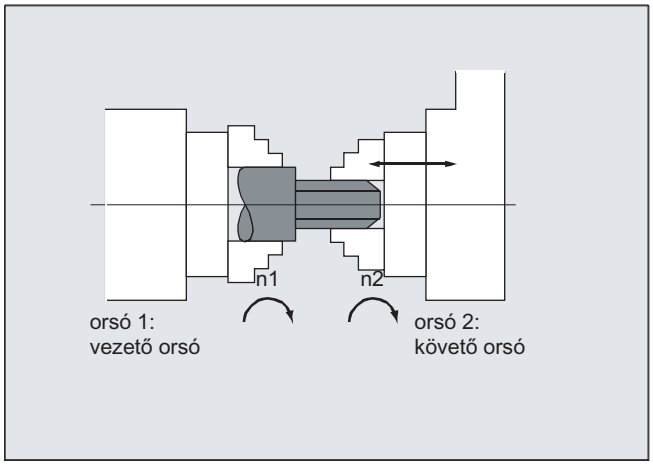
követő-orsó / vezető-orsó = számláló / nevező

A számláló programozva kell legyen. A nevező nem kell programozva legyen. A nevezőként ekkor az 1.0 alap-beállítás érték lesz használva.

Példa:

S2 követő-orsó és S1 vezető-orsó, áttételi arány = 1 / 1

```
COUPDEF(S2, S1, 1.0)
```



Megjegyzés

Az áttételi arányt lehet bekapcsolt csatolásnál és forgó orsóknál is változtatni.

Mondatváltás-viselkedés NOC, FINE, COARSE, IPOSTOP

A mondatváltás-viselkedés programozásánál a következő rövidített írásmód lehetséges:

- `"NO"`: azonnal (előbeállítás)
- `"FI"`: "szinkronfutás finom" eléréséné l

- "CO": "szinkronfutás durva" elérésénél
- "IP": IPOSTOP elérésével, azaz parancsérték oldali szinkronfutás után

csatolási mód

Megjegyzés

A csatolásmódot csak kikapcsolt csatolásnál szabad megváltoztatni!

Szinkronüzem bekapcsolás COUPON, <POSFS>

- Csatolás bekapcsolása tetszőleges szögeltolással az LS vezető-orsó és az FS követő-orsó között:
  - COUPON(S2,S1)
  - COUPON(S2)
- csatolás bekapcsolása szögeltolással <POSFS>
  <POSFS> vezető-orsó 0°-pozíciójára vonatkozik a pozitív forgásirányban
  értéktartomány<POSFS>: 0°… 359,999°
  - COUPON(S2,S1,30)

Megjegyzés

A szögeltolást lehet aktív csatolásnál is változtatni.

Követő-orsó pozícionálása

Bekapcsolt szinkron-orsó csatolásnál is lehet az FS követő-orsót az LS vezető-orsótól függetlenül a ±180° tartományban pozícionálni.

- FS követő-orsó orsó-pozícionálása SPOS-sal
  Példa: SPOS[2]=IC(-90)
  További információk az SPOS-hoz :
  Irodalom:
  Programozási kézikönyv, Alapok

Fordulatszám eltérés

Fordulatszám eltérés keletkezik a fordulatszám-vezérelt üzemben és aktív szinkron-orsó csatolásnál egy követő-orsó fordulatszám előjeles hozzá-adódásával a vezető-orsó mozgása miatt az orsó programozással megadott követő-orsó fordulatszámhoz:

- szinkron-orsó csatolás `COUPONC`-vel
- `S<FS>=<fordulatszám>` [`M<FS>=<forgásirány>`]

Megjegyzés

Peremfeltételek

- Az `M3` / `M4` forgásiránnyal az `S...` fordulatszámot is újra kell programozni.
- Egy orsó-fordulatszám átlapolása (`M<forgásirány> S<FS>`) vezető-orsó mozgással a `COUPONC` szinkron-orsó csatolásnál csak akkor lesz hatásos, ha az átlapolás engedélyezve van.
- A dinamikát a vezető-orsónál annyira be kell korlátozni, hogy a követő-orsó átlapolásánál annak a dinamika határértékei ne legyenek túllépve.

További információk a fordulatszám-eltéréshez:
Irodalom:
Bővítő funkciók működési kézikönyv; Szinkronorsó (S3)

Sebesség, gyorsulás: FA, ACC, OVRA, VELOLIMA

Egy követő-orsó sebessége és gyorsulása a következőkkel programozható:

- `FA[SPI(S<n>)]` ill. `FA[S<n>]` (tengely sebesség)
- `ACC[SPI(S<n>)]` ill. `ACC[S<n>]` (tengely gyorsulás)
- `OVRA[SPI(S<n>)]` ill. `OVRA[S<n>]` (tengely override)
- `VELOLIMA[SPI(S<n>)]` ill. `VELOLIMA[S<n>]` (tengely sebesség-túllépés ill. -csökkentés)

`<n>` = 1, 2, 3, ... -mal (orsószám követő-orsóként)

Irodalom:
Programozási kézikönyv, Alapok

Megjegyzés

A maximális tengely-rándulás csökkentése vagy növelése az orsónál nem hatásos.

További információk a tengely dinamikához:
Irodalom:
Bővítő funkciók működési kézikönyv; Körtengelyek (R2)

Programozható mondatváltás viselkedés WAITC

A `WAITC`-vel lehet a mondatváltás viselkedést, pl. a csatolási paraméterek vagy pozícionálások változtatása után különböző szinkronfutás-feltételekkel (durva, finom `IPOSTOP`) megadni. Ha nincsenek szinkronfutás-feltételek megadva, akkor a `COUPDEF` definíciónál megadott mondatváltás-viselkedés érvényes.

Példa

- Várakozás a `FINE` szinkronfutás-feltétel elérésére az S2 követő-orsónál és `COARSE`-ra az S4 követő-orsónál: `WAITC(S2,"FINE",S4,"COARSE")`
- Várakozás a szinkronfutás-feltételek elérésére a `COUPDEF`-nek megfelelően: `WAITC( )`

Csatolást kikapcsolni COUPOF

`COUPOF`-fal meg lehet adni a csatolás kikapcsolási viselkedését:

- csatolás kikapcsolása azonnali mondatváltással:
  - `COUPOF(S2,S1)` (vezető-orsó megadásával)
  - `COUPOF(S2)` (vezető-orsó megadása nélkül)
- csatolás kikapcsolása a kikapcsolási pozíciókon túlhaladás után A mondatváltás a kikapcsolási pozíción túlhaladás után történik meg.
  - `COUPOF(S2,S1,150)` (kikapcsolási pozíció FS: 150°)
  - `COUPOF(S2,S1,150,30)` (kikapcsolási pozíció FS: 150°, LS: 30°)

Csatolás kikapcsolása követő-orsó állj-jal COUPOFS

`COUPOFS`-sel meg lehet adni a csatolás kikapcsolási viselkedését követő-orsó állj-jal:

- csatolás kikapcsolása követő-orsó állj-jal és azonnali mondatváltással:
  - `COUPOFS(S2,S1)` (vezető-orsó megadásával)
  - `COUPOFS(S2)` (vezető-orsó megadása nélkül)
- csatolás kikapcsolása a kikapcsolási pozíciókon túlhaladás után követő-orsó állj-jal A mondatváltás a kikapcsolási pozíción túlhaladás után történik meg.
  - `COUPOFS(S2,S1,150)` (kikapcsolási pozíció FS: 150°)

Csatolást törölni COUPDEL

`COUPDEL`-lel a csatolás törölve lesz:

- `COUPDEL(S2,S1)` (vezető-orsó megadásával)
- `COUPDEL(S2)` (vezető-orsó megadása nélkül)

Csatolás-paramétereket törölni COUPRES

`COUPRES`-sel lesznek a csatolás gép-és beállítási adatokban megadott paraméterei aktiválva:

- `COUPRES(S2,S1)` (vezető-orsó megadásával)
- `COUPRES(S2)` (vezető-orsó megadása nélkül)

Rendszerváltozók

- Követő-orsó aktuális csatolási állapota
  Egy követő-orsó aktuális csatolási állapotát bit-kódoltan lehet olvasni:
  `<érték> = $AA_COUP_ACT[<FS>]`

Bit	<érték>	Jelentés
-	0	nincs aktív csatolás
2	4	szinkronorsó-csatolás aktiv

Utalás

- Az összes többi érték a tengely-üzemre vonatkozik
- Ha az orsó több csatolásban követő-orsó, az összes csatolás csatolási állapotaként egy összegzett állapot lesz visszaadva.

- Aktuális szögeltolás
  A követő-orsó aktuális szögeltolását a vezető-orsóhoz képest olvasni lehet.
  - `$AA_COUP_OFFS[<FS>]` (parancsérték-oldali szögeltolás)
  - `$VA_COUP_OFFS[<FS>]` (valósérték-oldali szögeltolás)

  Alkalmazási példa
  Szögeltolás eltérés korrigálása az NC programban az után-vezetés üzem megszüntetése után:
  szögeltolás eltérés = programozott szögeltolás - rendszerváltozó

Irodalom

A rendszerváltozókhoz részletes információk találhatók:

Jegyzékes kézikönyv Rendszerváltozók

13.6 Generátoros csatolás (CP...)

A "Generátoros csatolás" egy általános csatolás-funkció, amelyben a létező csatolási módok (vontatás, vezető-érték csatolás, elektronikus hajtómű és szinkron-orsók) összes csatolási tulajdonságai össze vannak foglalva.

Ez a funkció rugalmas programozást tesz lehetővé.

- A felhasználó az alkalmazásához szükséges csatolási tulajdonságokat szabadon kiválaszthatja (építőkocka el).
- Minden csatolási tulajdonság egyenként programozható.
- Egy definiált csatolás csatolási tulajdonságai (pl. csatolási tényező) változtathatóak.
- További csatolási tulajdonságok későbbi használata lehetséges.
- A követő-tengely koordináta vonatkoztatási rendszere (bázis koordinátarendszer vagy gép koordinátarendszer) programozható.
- Bizonyos csatolási módokat lehet szinkronakciókban is programozni.
  Irodalom: Működési kézikönyv szinkronakciók

Megjegyzés

Az eddigi vontatás (TRAIL*), vezető-érték csatolás (LEAD*), elektronikus hajtómű (EG*) és szinkronorsó (COUP*) funkciók illesztő ciklusokkal továbbra is támogatottak.

Összes kulcsszó és csatolási tulajdonság áttekintése

A következő táblázat a generátoros csatolás összes kulcsszavának és az azzal programozható csatolási tulajdonságoknak áttekintése.

Kulcsszó	Csatolási tulajdonság / Jelentés	Szintaxis
CPDEF	egy csatolási mód létrehozása	CPDEF=(<FAx>)
CPDEL	egy csatolási mód törlése	CPDEL=(<FAx>)
CPLA	egy vezető tengely definíciója	CPLA[<FAx>]=(<LAx>)
CPLDEF	egy vezető tengely definíciója egy csatolási módus létrehozásához (lehetséges CPDEF + CPLA-val is)	CPLDEF[<FAx>]=(<LAx>) vagy CPDEF=(<FAx>) CPLA[<FAx>]=(<LAx>)
CPLDEL	egy csatolási módus egy vezető tengelyének törlése (lehetséges CPDEL + CPLA-val is)	CPLDEL[<FAx>]=(<LAx>) vagy CPDEL=(<FAx>) CPLA[<FAx>]=(<LAx>)
CPON	egy csatolási mód bekapcsolása	CPON=(<FAx>)
CPOF	egy csatolási mód kikapcsolása	CPOF=(<FAx>)
CPLON	egy csatolási módus egy vezető tengelyének bekapcsolása	CPLON[<FAx>]=<LAx>

Kulcsszó	Csatolási tulajdonság / Jelentés	Szintaxis		
CPLOF	egy csatolási módus egy vezető tengelyének kikapcsolása	CPLOF[<FAx>]=<LAx>		
CPLNUM	csatolási tényező számlálója	CPLNUM[FAx,LAx]=<érték>		
CPLDEN	csatolási tényező nevezője	CPLDEN[FAx,LAx]=<érték>		
CPLCTID	görbe-táblázatok száma	CPLCTID[FAx,LAx]=<érték>		
CPLSETVAL	csatolás vonatkoztatás	CPLSETVAL[FAx,LAx]="<csatolási vonatkozás>"		
		"<csatolás vonatkoztatás>":	"CMDPOS"	parancsérték csatolás
			"CMDVEL"	sebesség-csatolás
			"ACTPOS"	valósérték csatolás
CPFRS	koordináta vonatkoztatási rendszer	CPFRS[FAx]="<koordináta vonatkozás>"		
		"<koordináta vonatkozás>":	"BCS"	Alap-koordinátarendszer
			"MCS"	Gép-koordinátarendszer
CPBC	mondatváltás kritérium	CPBC[FAx]="<mondatváltás kritérium>"		
		"<mondatváltás kritérium>":	"NOC"	A mondatváltás a csatolás állapotától függetlenül történik.
			"IPOSTOP"	A mondatváltás parancsérték oldali szinkronfutásnál történik.
			"COARSE"	A mondatváltás valósérték oldali "durva" szinkronfutásnál történik.
			"FINE"	A mondatváltás valósérték oldali "finom" szinkronfutásnál történik.
CPFPOS + CPON	követő tengely pozíciója bekapcsolásnál	CPON=FAx CPFPOS[FAx]=<érték>		
CPLPOS + CPON	vezető tengely szinkron-pozíciója bekapcsolásnál	CPLPOS[FAx,LAx]=<érték>		

Kulcsszó	Csatolási tulajdonság / Jelentés	Szintaxis		
CPFMSON	szinkronizációs módus	CPFMSON[FAx]="<szinkronizáció módus>"		
		"<szinkronizáció módus>":	"CFAST"	A csatolás idő-optimáltan lesz lezárva.
			"CCOARSE"	A csatolás csak akkor lesz bekapcsolva, ha a csatolás összefüggés által igényelt követő-tengely pozíció az aktuális követő-tengely pozíció tartományban van.
			"NTGT"	A következő fog-hézag idő-optimalizáltan lesz felvéve.
			"NTGP"	A következő fog-hézag út-optimalizáltan lesz felvéve.
			"NRGT"	A következő szegmens a menetszám és a fogszám arányának megfelelően idő-optimalizáltan lesz felvéve.
			"NRGP"	A következő szegmens a menetszám és a fogszám arányának megfelelően út-optimalizáltan lesz felvéve.
			"ACN"	Csak körtengelyeknél! A körtengely negatív tengelyirányban megy a szinkronpozícióba. A szinkronizáció azonnal következik.
			"ACP"	Csak körtengelyeknél! A körtengely pozitív tengelyirányban megy a szinkronpozícióba. A szinkronizáció azonnal következik.
			"DCT"	Csak körtengelyeknél! A körtengely negatív idő-optimalizáltan megy a szinkronpozícióba. A szinkronizáció azonnal következik.
			"DCP"	Csak körtengelyeknél! A körtengely negatív út-optimalizáltan megy a szinkronpozícióba. A szinkronizáció azonnal következik.

Kulcsszó	Csatolási tulajdonság / Jelentés	Szintaxis		
CPFMON	követő tengely viselkedése teljes bekapcsolásnál	CPFMON[FAx]="<bekapcsolás viselkedés>"		
		"<bekapcsolás viselkedés>":	"STOP"	Csak orsóknál! A követő-orsó aktív mozgását a bekapcsolás megállítja.
			"CONT"	Csak orsóknál és fő-futam tengelyeknél! A követő-tengely/orsó aktuális mozgása kezdő mozgásként átvételre kerül a csatolásba.
			"ADD"	Csak orsóknál! A csatolás mozgás-részei az aktuális átlapolt mozgáshoz kiegészítőleg hatnak, vagyis a követő-tengely/orsó aktuális mozgása átlapolt mozgásként lesz megtartva.
CPFMOF	követő tengely viselkedése teljes kikapcsolásnál	CPFMOF[FAx]="<kikapcsolás viselkedés>"		
		"<kikapcsolás viselkedés>":	"STOP"	megállítja a követő-tengelyt/orsót Az aktív átlapolt mozgás álló állapotig lesz fékezve. Ezután a csatolás bontva lesz.
			"CONT"	Csak orsóknál és fő-futam tengelyeknél! A követő orsó a kikapcsolási időpontban aktuális fordulatszámmal/sebességgel fut tovább.
CPFPOS + CPOF	követő-tengely kikapcsolási pozíciója kikapcsolásnál	CPOF=(FAx) CPFPOS[FAx]=<érték>		

Kulcsszó	Csatolási tulajdonság / Jelentés	Szintaxis		
CPMRESET	csatolás viselkedése RESET-nél	CPMRESET[FAx]="<Reset viselkedés>"		
		"<Reset viselkedés>":	"NONE"	A csatolás aktuális állapota lesz megtartva.
			"ON"	Ha a megfelelő csatolási módus be van állítva, a csatolás bekapcsol. Az összes definiált vezető-tengely kapcsolat aktiválva lesz. Ez akkor is megtörténik, ha ezen vezető-tengely kapcsolatok mind vagy részben már aktívak, vagyis egy teljesen aktivált csatolásnál is újra szinkronizálás történik.
			"OF"	Az aktív átlapolt mozgás álló állapotig lesz fékezve. Ezután a csatolás ki lesz kapcsolva. Ha a megfelelő csatolás modul definíció (CPDEF) lett létrehozva, a csatolás modul törlésre kerül. Egyéb esetekben továbbra is megmarad, vagyis továbbra is lehet használni.
			"OFC"	Csak orsóknál lehetséges! A követő orsó a kikapcsolási időpontban aktuális fordulatszámmal/sebességgel fut tovább. A csatolás ki lesz kapcsolva. Ha a megfelelő csatolás modul definíció (CPDEF) lett létrehozva, a csatolás modul törlésre kerül. Egyéb esetekben továbbra is megmarad, vagyis továbbra is lehet használni.
			"DEL"	Az aktív átlapolt mozgás álló állapotig lesz fékezve. Ezután a csatolás deaktiválva és ezt követően törölve lesz.
			"DELC"	Csak orsóknál lehetséges! A követő orsó a kikapcsolási időpontban aktuális fordulatszámmal/sebességgel fut tovább. A csatolás deaktiválva és ezt követően törölve lesz.

Kulcsszó	Csatolási tulajdonság / Jelentés	Szintaxis		
CPMSTART	csatolás viselkedése munkadarabprogram indításánál	CPMRESET[FAx]="<Start viselkedés>"		
		"<Start viselkedés>":	"NONE"	A csatolás aktuális állapota lesz megtartva.
			"ON"	Csatolás bekapcsolva. Az összes definiált vezető-tengely kapcsolat aktiválva lesz. Ez akkor is megtörténik, ha ezen vezető-tengely kapcsolatok mind vagy részben már aktívak, vagyis egy teljesen aktivált csatolásnál is újra szinkronizálás történik.
			"OF"	A csatolás ki lesz kapcsolva. Ha a megfelelő csatolás modul definíció (CPDEF) lett létrehozva, a csatolás modul törlésre kerül. Egyéb esetekben továbbra is megmarad, vagyis továbbra is lehet használni.
			"DEL"	A csatolás deaktiválva és ezt követően törölve lesz.
CPMPRT	csatolás viselkedése munkadarabprogram indításánál programteszt általi mondatkeresés alatt	CPMPRT[FAx]="<Start viselkedés>"		
		"<Start viselkedés>":	lásd CPMSTART	
CPLINTR	eltolási érték egy vezető tengely bemenet értékéhez	CPLINTR[FAx,LAx]=<érték>		
CPLINSC	skálázási tényező egy vezető tengely bemenet értékéhez	CPLINSC[FAx,LAx]=<érték>		
CPLOUTTR	eltolási érték egy csatolás kimenet értékéhez	CPLOUTTR[FAx,LAx]=<érték>		
CPLOUTSC	skálázási tényező egy vezető tengely kimenet értékéhez	CPLOUTSC[FAx,LAx]=<érték>		
CPSYNCOP	küszöbérték a "durva" pozíció-szinkron futásra	CPSYNCOP[FAx]=<érték>		
CPSYNFIP	küszöbérték a "finom" pozíció-szinkron futásra	CPSYNFIP[FAx]=<érték>		
CPSYNCOP2	második küszöbérték a "durva" pozíció-szinkron futásra	CPSYNCOP2[FAx]=<érték>		
CPSYNFIP2	második küszöbérték a "finom" pozíció-szinkron futásra	CPSYNFIP2[FAx]=<érték>		
CPSYNCOV	küszöbérték a "durva" sebesség-szinkron futásra	CPSYNCOV[FAx]=<érték>		
CPSYNFIV	küszöbérték a "finom" sebesség-szinkron futásra	CPSYNFIV[FAx]=<érték>		

Kulcsszó	Csatolási tulajdonság / Jelentés	Szintaxis		
CPMBRAKE	követő tengely viselkedése adott állj-jeleknél és parancsoknál	CPMBRAKE[FAx]=<bit-kódolt érték>		
CPMVDI	követő tengely viselkedése adott NC/PLC interfész jelekre	CPMVDI[FAx]=<bit-kódolt érték>		
CPMALARM	speciális csatolás-vonatkozású vészjelzések kiadásának elnyomása	CPMALARM[FAx]=<bit-kódolt érték>		
CPSETTYPE	csatolás típus	CPSETTYPE[FAx]="<csatolás típus>"		
		"<csatolás típus>":	"CP"	szabad programozhatóság
			"TRAIL"	"vontatás" csatolás típus
			"LEAD"	"vezető-érték csatolás" csatolás típus
			"EG"	"elektronikus hajtómű" csatolás típus
			"COUP"	"szinkron-orsó" csatolás típus

FAx: követő-tengely/-orsó

LAx: vezető-tengely/-orsó

Megjegyzés

A közvetlen módon (munkadarabprogramban vagy szinkronakciókban) nem programozott csatolási tulajdonságok az alap-beállítási értékeikkel lesznek hatásosak.

A CPSETTYPE kulcsszó beállításától függően az alap-beállítások (CPSETTYPE="CP") helyett előre beállított csatolási tulajdonságok is lehetnek hatásosak..

Irodalom

A generátoros csatoláshoz részletes információk találhatók:

- Működési kézikönyv Különleges funkciók; M3: Tengely csatolások, fejezet: "Generátoros csatolás"

13.7 Master/Slave csatolás (MASLDEF, MASLDEL, MASLON, MASLOF, MASLOFS)

A "Master/Slave csatolás" lehetővé teszi:

- a Slave tengely becsatolását a Master tengelyéhez az érintett tengelyek nyugalmi állapotában.
- forgó, fordulatszám-vezérelt orsók csatolását és leválasztását.
- dinamikus beállítás

Megjegyzés

Pozicionáló üzem

A pozícionáló üzemben levő tengelyeknél és orsóknál a csatolás csak nyugalmi helyzetben kapcsolható be és bontható.

Szintaxis

```
MASLON(<Slave_1>,<Slave_2>,...)
MASLOF(<Slave_1>,<Slave_2>,...)
MASLOFS(<Slave_1>,<Slave_2>,...)
```

dinamikus beállítás:

```
MASLDEF(<Slave_1>,<Slave_2>,...,<Master>)
MASLDEL(<Slave_1>,<Slave_2>,...)
```

Jelentés

MASLON:	Egy ideiglenes Master/Slave csatolást bekapcsolni	
	<Slave_x>,...:	Slave-tengely 1 … n
MASLOF:	Egy aktív Master/Slave csatolást kikapcsolni	
	<Slave_1>,...:	Slave-tengely 1 … n
MASLOFS:	Master/Slave csatolást leválasztani és Slave orsókat automatikusan lefékezni (lásd "Orsó csatolási viselkedése fordulatszám vezérelt üzemben" utalást!)	
	<Slave_1>,...:	Slave-tengely 1 … n
MASLDEF:	Master/Slave egyesülést a munkadarabprogramból létrehozni/változtatni	
	<Slave_1>,...:	Slave-tengely 1 … n
	<Master>:	Master tengely
MASLDEL:	Master/Slave csatolás felbontása és az egyesülés definíciójának törlése	
	<Slave_1>,...:	Slave-tengely 1 … n
	Utalás: A gépadatokban megadott Master/Slave definíciók megmaradnak.	

Megjegyzés

Csatolási viselkedés orsóknál a fordulatszám-vezérelt üzemben

A fordulatszám-vezérelt üzemben levő orsóknál a MASLON, MASLOF, MASLOFS és MASLDEL csatolási viselkedését a következő gépadat adja meg:

MD37263 $MA_MS_SPIND_COUPLING_MODE

Az MD 37263 = 0 alap-beállításnál a Slave-tengelyek becsatolása és leválasztása kizárólag az érintett tengelyek nyugalmi állapotában történik. MASLOFS megfelel MASLOF-nak.

Az MD 37263 = 1 esetén a csatolási utasítás közvetlenül és ezzel már mozgásban is végre lesz hajtva. A csatolás a MASLON-nál azonnal létrejön és MASLOFS vagy MASLOF-nál azonnal megszűnik. Az adott időpontban forgó Slave-orsók megtartják a MASLOF-nál a fordulatszámukat az új fordulatszám programozásáig. Ezzel szemben a MASLOFS-nál automatikusan le lesznek fékezve.

Megjegyzés

MASLOF/MASLOFS esetén nincs közvetett előrefutás-állj. A hiányzó előrefutás-állj miatt a $P-rendszerváltozók a Slave-tengelyekre az új programozás időpontjáig nem adnak aktualizált értékeket.

Megjegyzés

A Slave-tengelyre a valósértéket PRESETON-nal a Master-tengellyel azonos értékre lehet szinkronizálni. Ehhez a tartós Master-/Slave-csatolást rövid időre ki kell kapcsolni a nem referált Slave-tengelyek valósértékének POWER ON-nal a Master-tengely értékére beállításához. Azután a tartós csatolás ismét helyre lesz állítva.

A tartós Master-/Slave-csatolás a következő gépadattal lesz aktiválva:
MD37262 $MA_MS_COUPLING_ALWAYS_ACTIVE = 1
Ennek nincs hatása az ideiglenes csatolás utasításaira.

Példák

Példa 1: Valósérték beállítása Master/Slave csatolás Slave-tengelyeinél

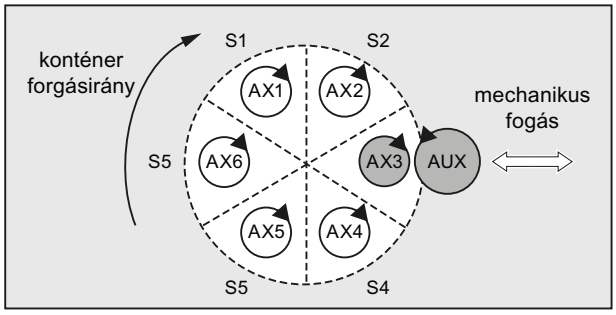
Egy állandó Master/Slave csatolásnál a PRESETON-nal a Slave-tengely valósértéke a Master-tengely értékére lesz beállítva.

Programkód	Kommentár
$MA_MS_COUPLING_ALWAYS_ACTIVE[AX2]=0	; Slave-tengely állandó csatolását kikapcsolni
NEWCONF	; gépadat változásokat aktiválni
STOPRE	
MASLOF(Y1)	; ideiglenes csatolást kikapcsolni
PRESETON(AX2,$VA_IM(M_AX))	; Slave-tengely valósértéke = Master-tengely valósértéke

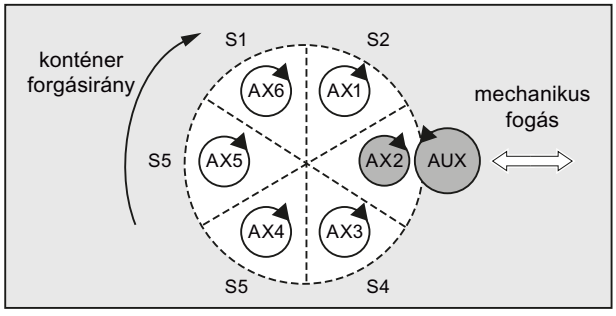
Programkód	Kommentár
$MA_MS_COUPLING_ALWAYS_ACTIVE[AX2]=1	; Slave-tengely állandó csatolását bekapcsolni
NEWCONF	; gépadat változásokat aktiválni

Példa 2: Master/Slave csatolás dinamikus beállítása

A csatolás létrehozásához egy másik orsóval a tengely-konténerforgatás után a régi csatolást előbb oldani, a beállítást törölni és az új csatolást beállítani kell.



Kép 13-1 Tengely-konténer forgatás előtt



Kép 13-2 Tengely-konténer forgatás egy osztással után

Programkód	Kommentár
MASLDEF(AUX,S3)	; AUX: Slave, S3: Master = AX3
MASLON(AUX)	; csatolás be
M3=3 S3=4000	; Master forgatása
MASLDEL(AUX)	; csatolás oldása és törlése
AXCTSWE(CT1)	; tengely-konténer forgatás engedélyezése
MASLDEF(AUX,S3)	; AUX: Slave, S3: Master = AX2

Irodalom

- Különleges funkciók működési kézikönyv, "TE3: Fordulatszám/forgatónyomaték csatolás, Master-Slave" fejezet
- Bővítő funkciók működési kézikönyv; "B3 Decentralizált rendszer - csak 840D sl" > "NCU-Link" > "Tengely-konténer" fejezet

Szinkron-akciók 14

14.1 Egy szinkronakció definíciója

Egy szinkronakció definíciója egy munkadarabprogram egy mondatában történik. Ezen mondaton belül nem szabad további urasításokat programozni, amelyek nem részei a szinkronakciónak.

Egy szinkronakció a következő komponensekből áll:

érvényesség, azonosító szám (opcionális)	utasítás rész (opcionális)			akció rész		
	gyakoriság	G utasítás (opcionális)	feltétel	kulcsszó	G utasítás (opcionális)	akciók
--- [1] ID=<Nr> IDS=<Nr>	--- [1] WHENEVER FROM WHEN EVERY	G...	logikai kifejezés	DO	G...	akció 1 ... akció n

[1] nincs programozva

Szintaxis

```
DO <akció 1> ... <akció n>
<gyakoriság> [<G-funkció>] <feltétel> DO <akció 1> ... <akció n>
ID=<sz><gyakoriság> [<G-funkció>] <feltétel> DO <akció 1> ... <akció
n>
IDS=<sz><gyakoriság> [<G-funkció>] <feltétel> DO <akció 1> ...
<akció n>
```

Irodalom

A szinkronakciók működésének részletes leírás található:

Szinkronakciók működési kézikönyv

Ingázás 15

15.1 Aszinkron ingázás (OS, OSP1, OSP2, OST1, OST2, OSCTRL, OSNSC, OSE, OSB)

Egy inga-tengely az 1 és 2 fordulópontok között a megadott előtolással oda és vissza jár, amíg az ingamozgás ki nem lesz kapcsolva.

Más tengelyek az ingamozgás közben tetszőlegesen interpolálhatnak. Egy pályamozgással vagy egy pozícionáló-tengellyel el lehet érni egy folyamatos fogásvételt. Ennek során **nincs összefüggés** az inga- és a fogásvételi mozgás között.

Aszinkron ingázás tulajdonságai

- Az aszinkron ingázás tengely-specifikusan a mondathatárokon át hatásos.
- A munkadarabprogramban az ingamozgás mondat-szinkron bekapcsolása biztosított.
- Több tengely közös interpolációja és az ingaszakaszok átlapolása nem lehetséges.

Programozás

A következő címekkel lehetséges a munkadarab-programból az aszinkron ingázásnak az NC programnak megfelelő bekapcsolása és befolyásolása.

A programozott értékek a főfutásban mondat-szinkron lesznek a beállítási adatokba beírva és a következő változásig hatásosak maradnak.

Szintaxis

```
OSP1[<tengely>]=<érték> OSP2[<tengely>]=<érték>
OST1[<tengely>]=<érték> OST2[<tengely>]=<érték>
FA[<tengely>]=<érték>
OSCTRL[<tengely>]=(<be opció>,<ki opció>)
OSNSC[<tengely>]=<érték>
OSE[<tengely>]=<érték>
OSB[<tengely>]=<érték>
OS[<tengely>]=1
OS[<tengely>]=0
```

Jelentés

`<tengely>`:	inga-tengely neve		
`OS`:	ingázást be-/kikapcsolni		
	Érték:	1	ingázást **be**kapcsolni
		0	ingázást **ki**kapcsolni
`OSP1`:	fordulópont 1 pozíció megadása		

OSP2:	fordulópont 2 pozíció megadása **Utalás:** Ha egy növekményes mozgás aktív, akkor a pozíció az utolsó NC programban programozott megfelelő fordulás-pozícióhoz növekményesen lesz beszámítva.		
OST1:	állásidőt a fordulópont 1-en [s]-ban megadni		
OST2:	állásidőt a fordulópont 2-n [s]-ban megadni		
	<érték>:	-2	interpoláció folytatása pontos állj-ra várakozás nélkül
		-1	várakozás pontos állj durvára
		0	várakozás pontos állj finomra
		>0	várakozás pontos állj finomra és ezután állásidő kivárása **Utalás:** Az állásidő egysége azonos a G4-gyel programozott állásidőével.
FA:	Előtolás sebesség megadása Előtolás sebességként a pozícionáló-tengely definiált előtolás sebessége érvényes. Ha nincs előtolás sebesség programozva, a gépadatban beállított érték érvényes.		
OSCTRL:	Beállítási és törlési opciók megadása A 0 - 3 opció értékek a fordulópontokon viselkedést adják meg kikapcsolásnál. A 0- 3 variációk egyikét lehet választani. A többi beállítások igény szerint kombinálhatók a kiválasztott variációval. Több opció plusz jellel (+) fűzhető össze.		
	<érték>:	0	Az ingamozgás kikapcsolásánál a következő fordulóponton megállni **Utalás:** csak az 1 és 2 értékek törlésével lehetséges
		1	Az ingamozgás kikapcsolásánál az 1 fordulópontnál megállni
		2	Az ingamozgás kikapcsolásánál az 2 fordulópontnál megállni
		3	Az ingamozgás kikapcsolásánál nem kell a fordulópontra menni, ha nincsenek kiszikráztatási löketek programozva
		4	A kiszikráztatás után egy végpozícióra menni
		8	Az ingamozgás maradékút törléssel lesz megszakítva, ezután kiszikráztatási löketeket megtenni és esetleg végpozícióra menni
		16	Az ingamozgás maradékút törléssel lesz megszakítva, ezután mint a lekapcsolásnál, megfelelő fordulópontra menni
		32	A megváltozott előtolás csak a következő fordulóponttól aktív
		64	FA egyenlő 0, FA = 0: út-rátevődés aktív FA nem egyenlő 0, FA <> 0: sebesség-rátevődés aktív
		128	Körtengelynél DC (legrövidebb út)
		256	Kiszikráztatás dupla löketként végrehajtva. (alaphelyzet) 1 = kiszikráztatási löket egyes löketként lesz végrehajtva.
		512	kezdő pozícióra menni
OSNSC:	kiszikráztatási löketek számát megadni		

OSE:	végpozíciót (MKR-ben) megadni, amire az ingázás kikapcsolása után rá kell menni **Utalás:** Az "OSE" programozásánál az "OSCTRL"-re implicit az opció 4 lesz hatásos.
OSB:	Kezdő pozíciót (MKR-ben) megadni, amire az ingázás bekapcsolása előtt rá kell menni A kezdő pozíció a fordulópont 1 előtt lesz felvéve. Ha a kezdő pozíció egyezik a fordulópont 1-gyel, akkor következőként a fordulópont 2-re lesz rámenet. A kezdő pozícióban nem hat az állásidő, még ha a kezdő pozíció egybe is esik a fordulópont 1-gyel, ehelyett pontos-állj finomra lesz várás. A beállított pontos-állj feltétel be lesz tartva. **Utalás:** A kezdő pozícióba menethez a SD43770 $SA_OSCILL_CTRL_MASK bit 9 beállítási adat be kell legyen állítva.

Példák

Példa 1: Inga-tengely két fordulópont között kell ingázzon

A Z inga-tengely Z 10 és 100 között kell ingázzon. A fordulópont 1-et pontos-állj finommal, a fordulópont 2-öt pontos-állj durvával kell elérni. Az inga-tengely 250-es előtolással működjön. A megmunkálás végén 3 kiszikráztatási löket legyen és az inga-tengely vegye fel a 200-as véghelyzetet. Az előtolás a fogásvételi tengelyre legyen 1, a fogásvétel vége X-irányban 15-nél legyen elérve.

Programkód	Kommentár
WAITP(X,Y,Z)	; kiinduló állapot
G0 X100 Y100 Z100	; átkapcsolás pozícionáló üzembe
WAITP(X,Z)	
OSP1[Z]=10 OSP2[Z]=100	; fordulópont 1, fordulópont 2
OSE[Z]=200	; végpozíció
OST1[Z]=0 OST2[Z]=-1	; állásidő U1-nél: pontos-állj finom
	; állásidő U2-nél: pontos-állj durva
FA[Z]=250 FA[X]=1	; inga-tengely előtolás, fogásvételi-tengely előtolás
OSCTRL[Z]=(4,0)	; beállító opciók
OSNSC[Z]=3	; 3 kiszikráztatási löket
OS[Z]=1	; ingázás indítása
WHEN $A_IN[3]==TRUE DO DELDTG(X)	; maradékút törlés
POS[X]=15	; X tengely kiindulási helyzet
POS[X]=50	; X tengely véghelyzet
OS[Z]=0	; ingázást megállítani
M30	

Megjegyzés

Az "OSP1[Z]=..." ... "OSNCS[Z]=..." utasítás sort lehet egy mondatban is programozni.

Példa 2: Ingázás a forduló-pozíció online változtatásával

Az aszinkron ingázáshoz szükséges beállítási adatokat a munkadarabprogramban be lehet állítani.

Ha a munkadarab-programban a beállítási adatok közvetlenül vannak írva, a változás már az előrefutás időpontjában hatásos. Szinkron viselkedést előrefutás állj-jal (STOPRE) lehet elérni.

Programkód	Kommentár
$SA_OSCILL_REVERSE_POS1[Z]=-10	
$SA_OSCILL_REVERSE_POS2[Z]=10	
G0 X0 Z0	
WAITP(Z)	
ID=1 WHENEVER $AA_IM[Z] < $$AA_OSCILL_REVERSE_POS1[Z] DO $AA_OVR[X]=0	; Ha az inga-tengelyek valósértéke a fordulópontot meghaladta, a fogásvétel tengely meg lesz állítva.
ID=2 WHENEVER $AA_IM[Z] < $$AA_OSCILL_REVERSE_POS2[Z] DO $AA_OVR[X]=0	
OS[Z]=1 FA[X]=1000 POS[X]=40	; ingázást bekapcsolni
OS[Z]=0	; ingázást kikapcsolni
M30	

További információk

Inga-tengely

Az inga-tengelyre érvényes:

- Minden tengelyt lehet inga-tengelyként használni.
- Egyidőben lehet több inga-tengely aktív (maximum: a pozícionáló-tengelyek száma).
- Az inga-tengelyre mindig a programban aktuálisan érvényes G-utasítástól függetlenül a G1 egyenes-interpoláció aktív.

Az inga-tengely lehet:

- bemenő-tengely dinamikus transzformációhoz
- vezető-tengely Gantry- és vontató-tengelyeknél
- mozgatható:
  - "BRISK" rándulás-határolás nélkül
    vagy
  - "SOFT" rándulás-határolással
    vagy
  - megtört gyorsulási jelleggörbével (mint a pozícionáló-tengelyek).

Inga fordulópontok

Az inga-pozíciók megadásánál az aktuális eltolásokat figyelembe kell venni:

- abszolút megadás
  "OSP1[Z]=<érték>"
  fordulópont pozíció = eltolások összege + programozott érték
- relatív megadás
  "OSP1[Z]=IC(<érték>)"
  fordulópont pozíció = fordulópont 1 + programozott érték

Példa:

Programkód

```
N10 OSP1[Z]=100 OSP2[Z]=110
...
N40 OSP1[Z]=IC(3)
```

WAITP

Ha egy geometria-tengelyt akarunk ingázni, ezt a "WAITP"-vel az ingázásra engedélyezni kell.

Az ingázás befejezése után "WAITP" utasítással az inga-tengely ismét pozícionáló-tengelyként lesz megadva és újra normálisan lehet használni.

Ingázás mozgás-szinkron akciókkal és állásidőkkel

A beállított állásidők lefutása után történik az ingázásnál a belső mondatváltás (a tengelyek új maradékútjaiból látható). A mondatváltáskor van a kikapcsolási funkció megvizsgálva. Ennél a mozgás lefutásának vezérlési beállításától függően (`OSCTRL`) lesz a kikapcsolási funkció beállítva. *Ez a időbeli viselkedés az előtolás override-dal befolyásolható.*

Adott körülmények között még egy ingalöket lesz végrehajtva a kiszikráztatási löketek indítása vagy a végpozíció felvétele előtt. *Ennek során az a benyomás keletkezik, mintha a kikapcsolási viselkedés megváltozna. Ez azonban nem így van.*

15.2 Szinkron-akciókkal vezérelt ingázás (OSCILL)

Ennél a fajta ingázásnál csak a fordulópontokon ill. a definiált fordulás-tartományban megengedett a fogásvételi mozgás.

A követelmények szerint az ingamozgást a fogásvétel közben lehet

- folytatni vagy
- megállítani, amíg a fogásvétel teljesen megtörténik.

Szintaxis

1. **Ingázás paraméterek megadása**
2. **Mozgás-szinkron akciók definiálása**
3. **Tengelyek hozzárendelése, fogásvétel megadása**

Jelentés

`OSP1[<inga-tengely>]=`	fordulópont 1 pozíció
`OSP2[<nga-tengely>]=`	fordulópont 2 pozíció
`OST1[<inga-tengely>]=`	állásidő a fordulópont 1-nél mp-ben
`OST2[<inga-tengely>]=`	állásidő a fordulópont 2-nél mp-ben
`FA[<inga-tengely>]=`	inga-tengely előtolása
`OSCTRL[<inga-tengely>]=`	be- ill. visszaállító opciók
`OSNSC[<inga-tengely>]=`	kiszikráztatási löketek száma
`OSE[<inga-tengely>]=`	végpozíció
`WAITP(<inga-tengely>)`	tengely engedélyezése ingázásra

Tengely hozzárendelés, fogásvétel

```
OSCILL[<inga-tengely>]=(<fogásvétel-tengely 1>,<fogásvétel-tengely
2>,<fogásvétel-tengely 3>)
POSP[<fogásvétel-tengely>]=(<vég-pozíció>,<rész-hossz>,<modus>)
```

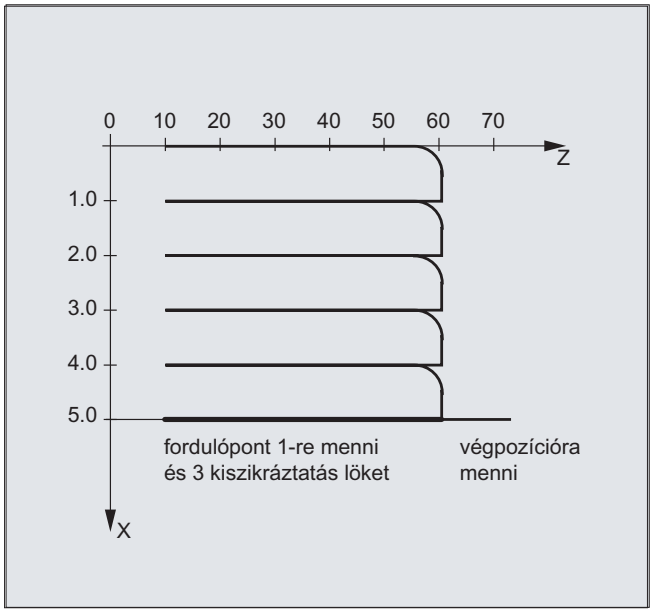
`OSCILL`:	fogásvétel-tengely(ek) hozzárendelése az ingatengelyhez
`POSP`:	össz- és rész-fogásvétel megadása (lásd Fájl és programkezelés fejezet)
`vég-pozíció`:	fogásvétel-tengely végpozíciója, miután az összes rész-fogásvétel megtörtént
`részhossz`:	rész-fogásvétel nagysága a fordulóponton/fordulás-tartományban
`modus`:	össz-fogásvétel felosztása rész-fogásvételekre = két azonos nagyságú maradéklépés (előbeállítás); = minden rész-fogásvétel azonos nagyságú

Mozgásszinkron akciók

`WHEN… … DO`	ha…, akkor…
`WHENEVER … DO`	mindig ha…, akkor…

Példa

A fordulópont 1-nél ne történjen fogásvétel. A fordulópont 2-nél a fogásvétel már ii2 távolságra fordulópont 2-től történjen és az inga-tengely a fordulóponton ne várjon a rész-fogásvétel befejezésére. A Z tengely az inga-tengely és az X tengely a fogásvétel-tengely.



1. paraméterek az ingázásra

Programkód	Kommentár
DEF INT ii2	; változót a fordulási-tartomány 2-re definiálni
OSP1[Z]=10 OSP2[Z]=60	; fordulópont 1-et és 2-t definiálni
OST1[Z]=0 OST2[Z]=0	; fordulópont 1: pontos-állj finom
	fordulópont 2: pontos-állj finom
FA[Z]=150 FA[X]=0.5	; előtolás inga-tengely Z, előtolás fogásvétel-tengely X
OSCTRL[Z]=(2+8+16,1)	; ingamozgás lekapcsolása a fordulópont 2-nél; RWL után kiszikráztatás és vég-pozícióra menet; RWL után megfelelő forduló-pozícióra menni
OSNC[Z]=3	; kiszikráztatás löketek
OSE[Z]=70	; végpozíció = 70
ii2=2	; fordulási-tartományt beállítani
WAITP(Z)	; ingázás engedélyezés a Z tengelyre

2. mozgás-szinkron akciók

Programkód	Kommentár
WHENEVER $AA_IM[Z]<$SA_OSCILL_REVERSE_POS2[Z] DO -> $AA_OVR[X]=0 $AC_MARKER[0]=0	; Mindig, ha kisebb a Z inga-tengely aktuális pozíciója GKR-ben mint a fordulási-tartomány 2 kezdete, akkor állítsa be az X fogásvétel-tengely override-ot 0%-ra és a 0 indexű jelölőt 0 értékre.

Programkód	Kommentár
WHENEVER $AA_IM[Z]>=$SA_OSCILL_REVERSE_POS2[Z] DO $AA_OVR[Z]=0	; Mindig, ha a Z inga-tengely aktuális pozíciója GKR-ben nagyobb egyenlő a fordulópont 2-vel, akkor állítsa be a Z inga-tengely override-ot 0%-ra.
WHENEVER $AA_DTEPW[X]==0 DO $AC_MARKER[0]=1	; Mindig, ha a rész-fogásvétel maradékút 0-val egyenlő, akkor állítsa be a 0 indexű jelölőt 1 értékre.
WHENEVER $AC_MARKER[0]==1 DO $AA_OVR[X]=0 $AA_OVR[Z]=100	; Mindig, ha a 0 indexű jelölő 1-gyel egyenlő, akkor állítsa be az X fogásvétel-tengely override-ot 0%-ra. Ezzel egy túl korai fogásvétel meg lesz akadályozva (a Z ingatengely a fordulási-tartomány 2-t még nem hagyta el újra, de az X fogásvétel-tengely X kész egy új fogásvételre). Állítsa be a Z inga-tengely override-ot 0%-ról (2. szinkronakció akciója) ismét 100 %-ra.

-> egy mondatban kell programozni

3. Ingázást indítani

Programkód	Kommentár
OSCILL[Z]=(X) POSP[X]=(5,1,1)	; A tengelyek indítása A Z inga-tengelyhez az X tengely hozzárendelése fogásvétel-tengelyként. Az X tengely a végpozíció 5-ig 1-es lépésekben menjen.
M30	; Programvég

További információ

1. **Ingázás paraméterek megadása**
   Az ingázás paramétereit meg kell adni a mozgási mondat előtt, amelyik a fogásvételi- és inga-tengely hozzárendeléseket, valamint a fogásvétel megadását tartalmazza (lásd "Aszinkron ingázás").
2. **Mozgás-szinkron akciók megadása**
   A szinkron feltételekkel történik:
   Fogásvétel elnyomása, amíg az inga-tengely egy fordulási tartományon belül (ii1, ii2) vagy egy fordulóponton (U1, U2) található.
   Ingamozgást a fogásvétel alatt a fordulóponton **megállítani**.
   Ingamozgást a rész-fogásvétel befejezése után újra **indítani**.
   Következő rész-fogásvétel indítását megadni.
3. **Inga- és fogásvétel-tengely hozzárendelés** ill. **össz- és rész-fogásvétel** megadása.

Ingázás paraméterek megadása

Inga- és fogásvétel-tengely hozzárendelés: OSCILL

```
OSCILL[<inga-tengely>]=(<fogásvétel-tengely1>,<fogásvétel-
tengely2>,<fogásvétel-tengely3>)
```

Az "OSCILL" utasítással történnek a tengely hozzárendelések és az ingamozgás indítása.

Maximum egy inga-tengelyhez 3 fogásvétel-tengelyt lehet hozzárendelni.

Megjegyzés

Az ingázás indítása előtt a tengelyek viselkedésének szinkron-feltételeit meg kell adni.

Fogásvételek megadása: POSP

```
POSP[<fogásvétel-tengely>]=(<vég-pozíció>,<osztáshossz>,<módus>)
```

A "POSP" utasítással közöljük a vezérléssel:

- az össz-fogásvételt (a végpozícióval)
- a fogásvétel nagyságát a fordulóponton ill. a fordulás-tartományban
- a rész-fogásvétel viselkedését a végpozíció elérésénél (módus)

módus = 0	Az utolsó két rész-fogásvételhez a célpontig megmaradt út felosztása történik 2 egyenlő nagyságú maradék lépésre (előbeállítás).
módus = 1	Minden rész-fogásvétel azonos méretű. Kiszámításuk az össz- fogásvételből történik.

Mozgás-szinkron akciók megadása

A következőkben leírásra kerülő mozgás-szinkron akciók általában használatosak az ingázáshoz.

Az egyes igények kielégítésére példa megoldások találhatók, amelyek az alkalmazó-specifikus ingamozgásokhoz modulokként alkalmazhatók.

Megjegyzés

Az egyes estekben a szinkron-feltételek másképp is programozhatók.

Kulcsszavak

`WHEN … DO …`	ha…, akkor…
`WHENEVER … DO`	mindig ha…, akkor…

Funkciók

A következőkben részletesen leírásra kerülő nyelvi eszközökkel a következő funkciókat lehet megvalósítani:

1. fogásvétel a fordulóponton
2. fogásvétel a fordulás-tartományban

3. fogásvétel mindkét fordulóponton
4. ingamozgás megállítása a fordulóponton
5. ingamozgás újra indítása
6. rész-fogásvételt nem túl korán indítani

Az összes, itt példaként leírt szinkronakcióra érvényesek a következő feltevések:

- fordulópont 1 < fordulópont 2
- Z = inga-tengely
- X = fogásvétel-tengely

Megjegyzés

Közelebbi magyarázat a Mozgás-szinkron akciók fejezetben.

Inga- és fogásvétel-tengely hozzárendelés ill. össz- és rész-fogásvétel megadása

Fogásvétel a fordulás-tartományban

A fogásvételi mozgás egy fordulási-tartományon belül kell kezdődjön, a fordulópont elérése előtt.

Ezek a szinkronakciók megakadályozzák a fogásvételi mozgás, amíg az inga-tengely egy fordulási-tartományban található.

Az adott feltételezések mellett (lásd fent) a következő utasítások adódnak:

Fordulási-tartomány 1:

```
WHENEVER $AA_IM[Z]>$SA_OSCILL_RESERVE_POS1[Z]+ii1 DO $AA_OVR[X]=0
```

Mindig, ha az aktuális inga-tengely aktuális pozíciója GKR-ben nagyobb egyenlő a fordulópont 1-vel, akkor állítsa be a inga-tengely override-ot 0%-ra.

Fordulási-tartomány 2:

```
WHENEVER $AA_IM[Z]<$SA_OSCILL_RESERVE_POS2[Z]+ii2 DO $AA_OVR[X]=0
```

Mindig, ha az aktuális inga-tengely aktuális pozíciója KR-ben kisebb a fordulópont 2-nél, akkor állítsa be a inga-tengely override-ot 0%-ra.

Fogásvétel a fordulóponton

Amíg az inga-tengely a fordulópontot nem érte el, nem történik a fogásvétel-tengelyen mozgás.

Az adott feltételezések mellett (lásd fent) a következő utasítások adódnak:

Fordulási-tartomány 1:

```
WHENEVER $AA_IM[Z]<>$SA_OSCILL_RESERVE_POS1[Z] DO $AA_OVR[X]=0
$AA_OVR[Z]=100
```

Mindig, ha a Z inga-tengely aktuális pozíciója GKR-ben nagyobb vagy kisebb mint a fordulási-tartomány 1 kezdete, akkor állítsa be az X fogásvétel-tengely override-ot 0%-ra és a Z inga-tengely override-ot 100%-ra

Fordulási-tartomány 2:

fordulópont 2-höz:

```
WHENEVER $AA_IM[Z]<>$SA_OSCILL_RESERVE_POS2[Z] DO $AA_OVR[X]=0
$AA_OVR[Z]=100
```

Mindig, ha a Z inga-tengely aktuális pozíciója GKR-ben nagyobb vagy kisebb mint a fordulópont 2 pozíciója, akkor állítsa be az X fogásvétel-tengely override-ot 0%-ra és a Z inga-tengely override-ot 100%-ra

Az ingamozgás megállítása a fordulóponton

Az inga-tengely a fordulóponton megáll, egyidejűleg kezdődik a fogásvételi mozgás. Az ingamozgás folytatódik, ha a fogásvételi mozgás teljesen végre van hajtva.

Ezt a szinkronakciót egyidejűleg lehet a fogásvételi mozgás indítására is használni, ha ez egy előző szinkronakcióval, amelyik még hatásos, meg lett állítva.

Az adott feltételezések mellett (lásd fent) a következő utasítások adódnak:

Fordulási-tartomány 1:

```
WHENEVER $SA_IM[Z]==$SA_OSCILL_RESERVE_POS1[Z] DO $AA_OVR[X]=0
$AA_OVR[Z]=100
```

Mindig, ha az inga-tengely aktuális pozíciója GKR-ben egyenlő a fordulópont 1-vel, akkor állítsa be az inga-tengely override-ot 0%-ra és a fogásvétel-tengely override-ot 100%-ra.

Fordulási-tartomány 2:

```
WHENEVER $SA_IM[Z]==$SA_OSCILL_RESERVE_POS2[Z] DO $AA_OVR[X]=0
$AA_OVR[Z]=100
```

Mindig, ha a Z inga-tengely aktuális pozíciója GKR-ben egyenlő a fordulópont 2-vel, akkor állítsa be az X inga-tengely override-ot 0%-ra és a fogásvétel-tengely override-ot 100%-ra.

A fordulópont online-kiértékelése

Ha az összehasonlítás jobb oldalán egy `$$` -ral jelölt főfutás-változó áll, akkor mindkét változó az IPO-ütemben folyamatosan kiértékelésre és összehasonlításra kerül.

Megjegyzés

Több információt ehhez lásd a "Mozgás-szinkron akciók" fejezetben".

Ingamozgást újra indítani

Ez a szinkronakció az inga-tengely mozgásának folytatásához, ha a rész-fogásvétel mozgás lezárult.

Az adott feltételezések mellett (lásd fent) a következő utasítások adódnak:

```
WHENEVER $AA_DTEPW[X]==0 DO $AA_OVR[Z]=100
```

Mindig, ha az X fogásvétel-tengely maradékútja a rész-fogásvételhez GKR-ben egyenlő nullával, akkor állítsa be az inga-tengely override-ot 100%-ra.

Következő rész-fogásvétel

Egy fogásvétel után a következő rész-fogásvétel túl korai indítását meg kell akadályozni.

Ehhez egy csatorna-specifikus jelzőt (`$AC_MARKER[Index]`) használunk, ami a rész-fogásvétel végén (rész-maradékút ≡ 0) lesz beállítva és a fordulási-tartomány elhagyásakor lesz törölve. Ekkor egy szinkronakcióval a következő fogásvétel mozgás meg lehet akadályozni.

Az adott feltételek mellett (lásd fent) pl. a fordulópont 1-ra a következő utasítások adódnak:

1. jelölő beállítása:

```
WHENEVER $AA_DTEPW[X]==0 DO $AC_MARKER[1]=1
```

Mindig, ha az X fogásvétel-tengely maradékútja a rész-fogásvételhez GKR-ben egyenlő nullával, akkor állítsa be az 1-es indexű jelölőt 1-re.

2. jelölő törlése

```
WHENEVER $AA_IM[Z]<> $SA_OSCILL_RESERVE_POS1[Z] DO $AC_MARKER[1]=0
```

Mindig, ha a Z inga-tengely aktuális pozíciója GKR-ben nagyobb vagy kisebb a fordulópont 1 pozíciójával, akkor állítsa be az 1-es jelölőt 0-ra.

3. fogásvétel megakadályozása

```
WHENEVER $AC_MARKER[1]==1 DO $AA_OVR[X]=0
```

Mindig, ha a jelölő 1-gyel egyenlő, akkor állítsa be az X fogásvétel-tengely override-ot 0%-ra.

Lyukasztás és sapkázás 16

16.1 Aktiválás / deaktiválás

16.1.1 Lyukasztás és sapkázás be-/kikapcsolás (SPOF, SON, PON, SONS, PONS, PDELAYON, PDELAYOF, PUNCHACC)

Lyukasztás ill. sapkázás aktiválás/deaktiválás

A PON és SON aktiválja a lyukasztás ill. sapkázás funkciót. A SPOF befejezi az összes lyukasztás- és sapkázás-specifikus funkciót. A PON és SON modálisan hatásos utasítások kölcsönösen kizárják egymást, vagyis a PON deaktiválja a SON-t és fordítva.

Lyukasztás és sapkázás előfeszítéssel

A SONS és PONS funkciók szintén a lyukasztó ill. sapkázó funkciókat kapcsolják be.

A SON/PON-nál hatásos löketvezérléssel az interpolációs szinten ellenében ezeknél a funkcióknál a löket-kioldás jeltechnikai vezérlése a szervó szinten történik. Ezáltal nagyobb löket-frekvenciával és nagyobb lyukasztási teljesítménnyel lehet dolgozni.

Az előfeszítés jelkiértékelése közben minden funkció tiltva van, amelyek a lyukasztó vagy sapkázó tengelyek pozíció-változását okozhatják (pl. kézi mozgatás kézikerékkel, frame-k változtatása PLC-ből, mérés funkciók).

Lyukasztás késleltetéssel

A PDELAYON a lyukasztó löket késleltetett kiadását okozza. Ennek a modálisan hatásos utasításnak előkészítő funkciója van és általában a PON előtt áll. A PDELAYOF után a lyukasztás normálisan folytatódik tovább.

Megjegyzés

A késleltetési idő az SD42400 $SC_PUNCH_DWELLTIME beállítási adatban van beállítva.

Útfüggő gyorsulás

A PUNCHACC utasítás megad egy gyorsulási jelleggörbét, ami a lyuktávolságtól függően különféle gyorsulásokat definiál.

Második lyukasztás interfész

Gépek, amelyek váltakozva egy második lyukasztás interfészt (második lyukasztó-egység vagy egy hasonló eszköz) használnak, átkapcsolhatóak a vezérlés gyors digitális be- és kimeneteinél egy második (I/O) párra. Mindkét interfész használható a teljes lyukasztás/

sapkázás funkcionalitásra. Az első és a második interfész között az átkapcsolás a `SPIF1` és `SPIF2` utasításokkal történik

Megjegyzés

Előfeltétel: egy második I/O-pár gépadatokkal definiálva kell legyen a lyukasztás funkciókhoz (→ lásd a gépgyártó tájékoztatásait!).

Szintaxis

```
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
```

Jelentés

`PON`:	lyukasztást aktiválni	
`SON`:	sapkázást aktiválni	
`PONS`:	lyukasztást előfeszítéssel aktiválni	
`SONS`:	sapkázást előfeszítéssel aktiválni	
`SPOF`:	lyukasztást /sapkázást deaktiválni	
`PDELAYON`:	lyukasztást késleltetéssel aktiválni	
`PDELAYOF`:	lyukasztást késleltetéssel deaktiválni	
`PUNCHACC`:	útfüggő gyorsulást aktiválni paraméter:	
	`<Smin>`	legkisebb lyuktávolság
	`<Amin>`	kezdeti gyorsulás `<Amin>` nagyobb kell legyen, mint az `<Amax>`!
	`<Smax>`	legnagyobb lyuktávolság
	`<Amax>`	végső gyorsulás `<Amax>` nagyobb lehet, mint az `<Amin>`!
`SPIF1`:	**első** lyukasztás interfészt aktiválni löketvezérlés az első pár gyors I/O-val történik	
`SPIF2`:	**második** lyukasztás interfészt aktiválni löketvezérlés a második pár gyors I/O-val történik	
	Utalás: RESET vagy a vezérlés felfutása után mindig az első interfész aktív. Ha csak egy interfész van használva, ezt nem kell programozni.	

Példák

Példa 1: Sapkázást aktiválni

Programkód	Kommentár
...	
N70 X50 SPOF	; Pozícionálás lyukasztás nélkül.
N80 X100 SON	; Sapkázást aktiválni, egy löket kiváltása a mozgás előtt (X=50) és a programozott mozgás végén (X=100).
...	

Példa 2: Lyukasztás késleltetéssel

Programkód	Kommentár
...	
N170 PDELAYON X100 SPOF	; Pozícionálás lyukasztás nélkül, késleltetett lyukasztás aktiválása.
N180 X800 PON	; Lyukasztást aktiválni. A vég-pozíció elérése utána lyukasztás löket késleltetve lesz kiadva.
N190 PDELAYOF X700	; Lyukasztást késleltetéssel deaktiválni, normális lyukasztás a programozott mozgás végén.
...	

Példa 3: Lyukasztás két interfésszel

Programkód	Kommentár
...	
N170 SPIF1 X100 PON	; A mondat végén történik egy löket kiadása az első gyors kimeneten. A "Löket aktív" jel az első bemeneten lesz felügyelve.
N180 X800 SPIF2	; A második löket kiadása a második gyors kimeneten történik. A "Löket aktív" jel a második bemeneten lesz felügyelve.
N190 SPIF1 X700	; Az összes további löketek vezérlése az első interfészen történik.
...	

További információk

Lyukasztás és sapkázás előfeszítéssel (PONS/SONS)

Lyukasztás és sapkázás előfeszítéssel egyidejűleg több csatornában nem lehetséges. `PONS` ill. `SONS` csak egy csatornában lehet aktiválva.

Útfüggő gyorsulás (PUNCHACC)

Példa:
`PUNCHACC(2,50,10,100)`

Lyuktávolság 2mm alatt:

A mozgás gyorsulása a maximális gyorsulás 50%-a.

Lyuktávolság 2mm...10mm:

A gyorsulás arányosan növekszik 100%-ra.

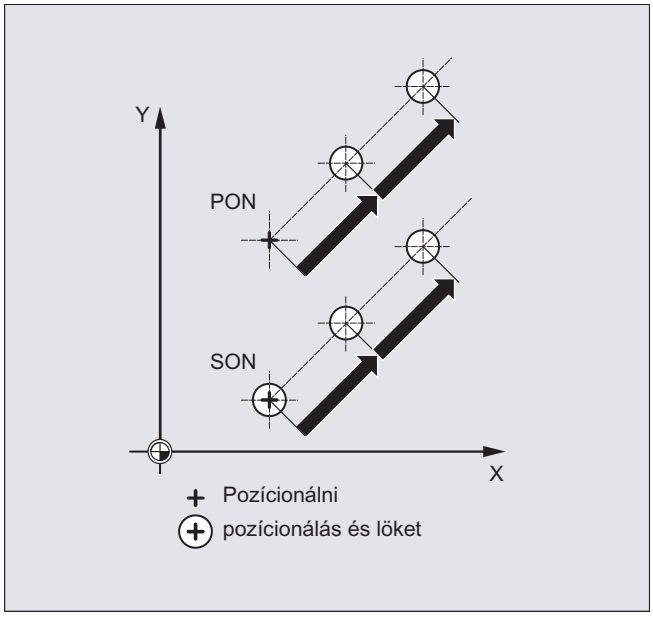
Lyuktávolság nagyobb 10mm-nél:

Mozgás 100% gyorsulással

Az első löket kioldása

A funkció aktiválása után az első löket kioldása a sapkázásnál és a lyukasztásnál időben különbözően történik:

- `PON/PONS`:
  - Minden löket - az első mondaté is az aktiválás után - a mondat végén történik.
- `SON/SONS`:
  - Az első löket a sapkázás aktiválása után már a mondat elején megtörténik.
  - Minden további löket a mondat végén történik.



Lyukasztás és sapkázás helyben

A löket kioldása csak akkor történik meg, ha a mondat a lyukasztó vagy sapkázó tengelyre mozgás (az aktív sík tengelyei) információt tartalmaz.

Egy löket kioldásához azonos helyen a lyukasztó vagy sapkázó tengelyt 0 mozgásúttal kell programozni.

Munka forgatható szerszámokkal

Megjegyzés

A forgatható szerszámoknak a programozott pályához érintőleges beállítására használjuk az érintő-vezérlést.

M utasítások használata

A makrótechnika felhasználásával a nyelvi utasítások helyett M utasításokat is használhatunk (kompatibilitás): Ennél a következő megfelelések érvényesek a régi rendszerrel:

```
M20, M23   ≙  SPOF
M22        ≙  SON
M25        ≙  PON
M26        ≙  PDELAYON
```

Példa makrófájlra:

Programkód	Kommentár
DEFINE M25 AS PON	; lyukasztás BE
DEFINE M125 AS PONS	; sapkázás előfeszítéssel be
DEFINE M22 AS SON	; sapkázás be
DEFINE M122 AS SONS	; sapkázás előfeszítéssel be
DEFINE M26 AS PDELAYON	; lyukasztás késleltetéssel be
DEFINE M20 AS SPOF	; lyukasztást, sapkázást ki
DEFINE M23 AS SPOF	; lyukasztást, sapkázást ki

Programozási példa:

Programkód	Kommentár
...	
N100 X100 M20	; Pozícionálás lyukasztás nélkül.
N110 X120 M22	; Sapkázást aktiválni, mozgás előtt és után löket.
N120 X150 Y150 M25	; Lyukasztást aktiválni, löket a mozgás végén.
...	

16.2 Automatikus út-felosztás

Felosztás rész-szakaszokra

Az aktív lyukasztásnál ill. sapkázásnál úgy az SPP, mint az SPN a pályatengelyekre programozott össz-elmozdulás felosztását eredményezi azonos hosszúságú rész-szakaszokra (azonos távolságú út-felosztás). Belül minden rész-szakasz egy mondatnak felel meg.

Löketek száma

A lyukasztásnál az első löket az első rész-szakasz végén történik, sapkázásnál ezzel szemben az első rész-szakasz kezdetén. A teljes mozgási szakaszra ezzel a következő számok adódnak:

lyukasztás: löketek száma = rész-szakaszok száma

sapkázás: löketek száma = rész-szakaszok száma + 1

Segédfunkciók

A segédfunkciók az első létrehozott mondatban lesznek végrehajtva.

Szintaxis

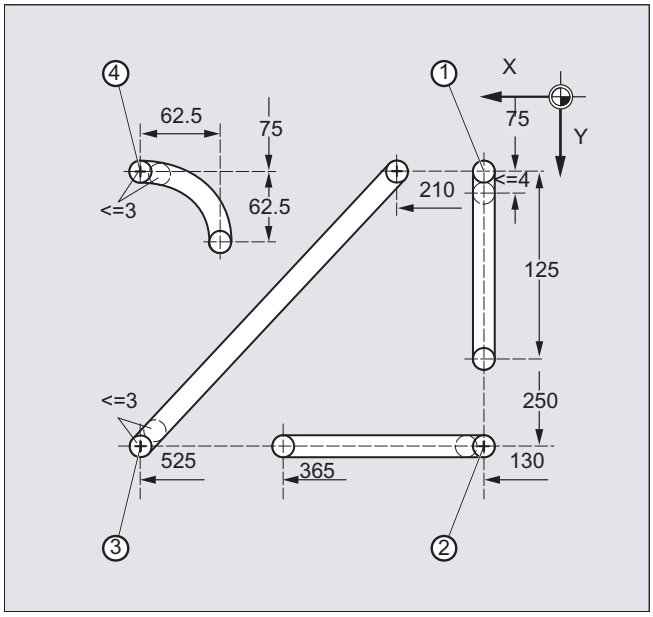
```
SPP=
SPN=
```

Jelentés

SPP:	rész-szakasz nagyság (maximális löket-távolság); modálisan hat
SPN:	rész-szakaszok száma mondatonként; mondatonként hat

Példa 1

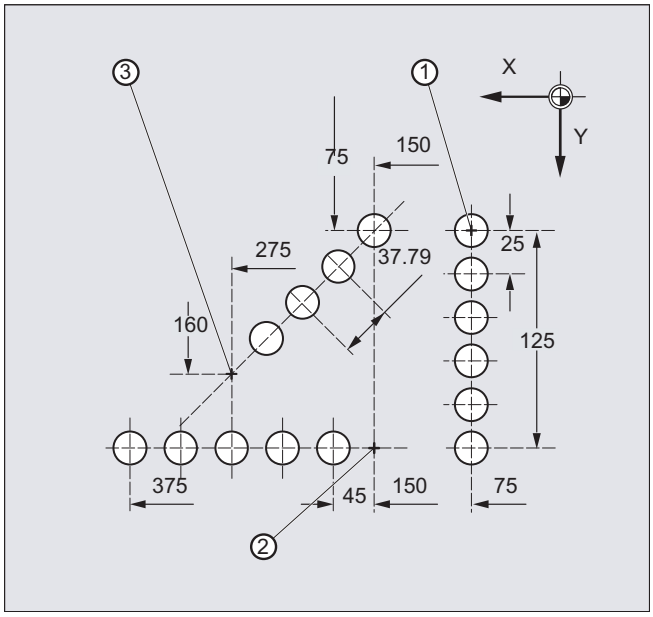
A programozott sapkázó szakaszokat automatikusan azonos nagyságú rész-szakaszokra kell felosztani.



Programkód	Kommentár
N100 G90 X130 Y75 F60 SPOF	; pozicionálás kezdőpont 1-re
N110 G91 Y125 SPP=4 SON	; sapkázás be; maximális rész-szakasz hossz automatikus út felosztásra: 4 mm
N120 G90 Y250 SPOF	; sapkázás ki; pozicionálás kezdőpont 2-re
N130 X365 SON	; sapkázás be; maximális rész-szakasz hossz automatikus út felosztásra: 4 mm
N140 X525 SPOF	; sapkázás ki; pozicionálás kezdőpont 3-ra
N150 X210 Y75 SPP=3 SON	; sapkázás be; maximális rész-szakasz hossz automatikus út felosztásra: 3 mm
N160 X525 SPOF	; sapkázás ki; pozicionálás kezdőpont 4-re
N170 G02 X-62.5 Y62.5 I J62.5 SPP=3 SON	; sapkázás be; maximális rész-szakasz hossz automatikus út felosztásra: 3 mm
N180 G00 G90 Y300 SPOF	; sapkázás ki

Példa 2

Az egyes lyuksorokra automatikus út felosztás szükséges. A felosztásra mindig meg van adva a maximális rész-szakasz hossz (SPP-érték).



Programkód	Kommentár
N100 G90 X75 Y75 F60 PON	; pozicionálás kezdőpont 1-re; lyukasztás be egyes lyuk lyukasztás
N110 G91 Y125 SPP=25	; maximális rész-szakasz hossz automatikus út felosztásra: 25 mm
N120 G90 X150 SPOF	; lyukasztás ki; pozicionálás kezdőpont 2-re
N130 X375 SPP=45 PON	; lyukasztás be; maximális rész-szakasz hossz automatikus út felosztásra: 45 mm
N140 X275 Y160 SPOF	; lyukasztás ki; pozicionálás kezdőpont 3-ra
N150 X150 Y75 SPP=40 PON	; lyukasztás be; a programozott 40 mm-es rész-szakasz hossz helyett; a kiszámított 37,79 mm rész-szakasz hossz lesz használva
N160 G00 Y300 SPOF	; lyukasztás ki; pozicionálás

16.2.1 Út felosztás pályatengelyeknél

SPP rész-szakasz hossza

Az `SPP`-vel megadjuk a maximális lökettávolságot és ezzel a rész-szakaszok maximális hosszát, amelyekre az össz-hosszat fel kell osztani. Az utasítás kikapcsolása `SPOF` -fal vagy `SPP=0`-val történik.

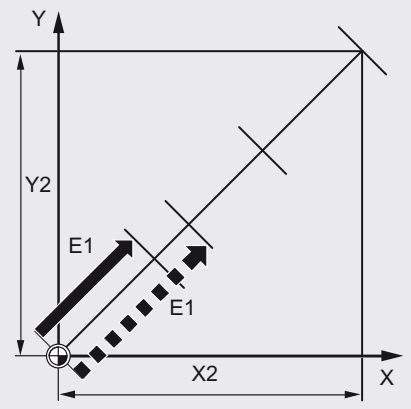
Példa:

```
N10 SON X0 Y0
N20 SPP=2 X10
```

A 10 mm-es össz-hossz 5 db 2 mm-es rész-szakaszra (`SPP=2`) lesz felosztva.

Megjegyzés

Az út felosztása `SPP` -vel mindig azonos távolságokkal történik: minden rész-szakasz azonos hosszú. Ez azt jelenti, hogy a programozott rész-szakasz méret (`SPP`értéke) csak akkor érvényes, ha az össz-hossz és az SPP-érték hányadosa egészszám. Ha ez nem így van, akkor a rész-szakaszok mérete belül úgy lesz csökkentve, hogy egészszámú hányados adódjon.



X2/Y2 programozott mozgási út (sapkázó vagy lyukasztó mondat)
E1 programozott rész-szakaszok
E1' automatikusak lekerekített rész-szakaszok

Példa:

```
N10 G1 G91 SON X10 Y10
N20 SPP=3.5 X15 Y15
```

A 15 mm-es össz-hossznál és egy 3,5 mm-es rész-szakasz méretnél nem adódik egészszámú hányados (4.28). Ezért az SPP-érték csökken a következő lehetséges egészszámú hányadosig. Ebben az esetben egy 3 mm-es rész-szakasz hossz adódik.

Rész-szakaszok száma SPN

Az `SPN` -nel adjuk meg a rész-szakaszok számát, amelyeket az össz-hosszból kell létrehozni. A rész-szakaszok hossza automatikusan lesz kiszámítva. Mivel az `SPN` mondatonként hatásos, előbb a lyukasztást vagy sapkázást `PON` -nal vagy `SON` -nal aktiválni kell.

16.2.2 Út felosztás egyes tengelyeknél

Ha a pályatengelyek mellett egyes tengelyek is vannak lyukasztó-sapkázó tengelyként definiálva, akkor ezek is érintettek lehetnek az automatikus útfelosztásban.

Egyes tengelyek viselkedése SPP-nél

A rész-szakaszok programozott hossza (`SPP`) alapvetően a pályatengelyekre vonatkozik. Ezért egy mondatban, amelyben az egyes tengely mozgás és az SPP-érték mellett nincs pályatengely programozva, az SPP-érték nem számít.

Ha egyes és pályatengelyek is vannak a mondatban programozva, az egyes tengely viselkedése a megfelelő gépadat beállításhoz igazodik.

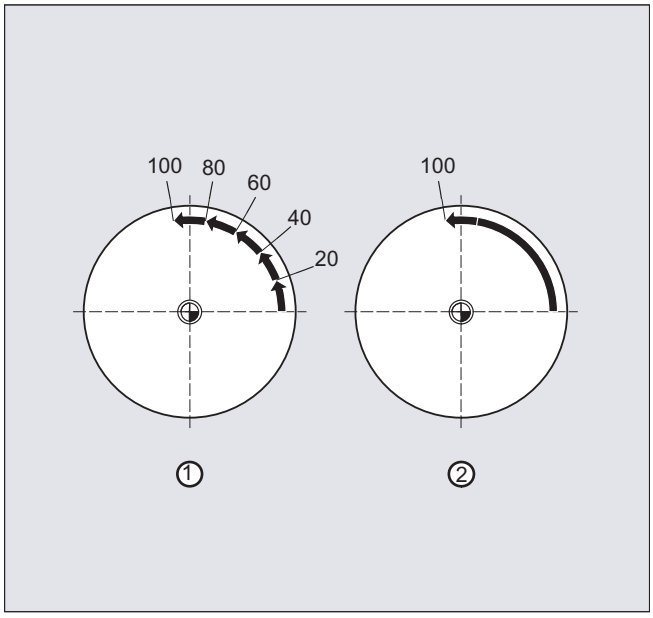
1. alapbeállítás
   Az egyes tengely útja egyenletesen lesz az `SPP` által létrehozott közbenső mondatokra elosztva.

Példa:

```
N10 G1 SON X10 A0
N20 SPP=3 X25 A100
```

A 3 mm-es löketszakasz miatt az X tengely (pályatengely) teljes 15 mm-es mozgási útjából 5 mondat lesz létrehozva.

Az A tengely ezzel minden mondatban 20°-ot fordul el.



1. Egyes tengely útfelosztás nélkül
   Az egyes tengely a teljes utat az első létrehozott mondatban teszi meg.
2. Különböző útfelosztás
   Az egyes tengely viselkedése a pályatengelyek interpolációjától függ:

- kör-interpoláció: út-felosztás
- egyenes interpoláció: nincs út felosztás

Viselkedés SPN-nél

A rész-szakaszok programozott száma akkor is érvényes, ha nincs egyidejűleg egy pályatengely is programozva.

Előfeltétel: az egyes tengely lyukasztó-sapkázó tengelyként van definiálva.

Köszörülés 17

17.1 Köszörű-specifikus szerszám-felügyelet be-/kikapcsolása (TMON, TMOF)

Az előre definiált TMON(...) és TMOF(...) eljárásokkal lehet a köszörű-specifikus szerszámfelügyeletet (geometria- és fordulatszám-felügyelet) be- ill. kikapcsolni.

Előfeltétel

A $TC_TPG1 ... $TC_TPG9 szerszám-specifikus paraméterek be kell legyenek állítva.

Szintaxis

```
TMON(<TNr>)
...
TMOF(<TNr>)
```

Jelentés

`TMON(...)`:	köszörű-specifikus szerszámfelügyelet bekapcsolása Az utasítást abban a csatornában kell programozni, amelyben a köszörű-specifikus szerszámfelügyeletet be kell kapcsolni.
`TMOF(...)`:	köszörű-specifikus szerszámfelügyelet kikapcsolása Az utasítást abban a csatornában kell programozni, amelyben a köszörű-specifikus szerszámfelügyeletet ki kell kapcsolni.
`<TNr>`:	T-szám **Utalás:** Csak akkor szükséges, ha a felügyeletet a használatban levő szerszám helyett egy nem aktív köszörű-tárcsára kell be- ill. kikapcsolni.
`TMOF(0)`:	Felügyelet kikapcsolása az összes szerszámra

További funkciók 18

18.1 Tengely funkciók (AXNAME, AX, SPI, AXTOSPI, ISAXIS, AXSTRING, MODAXVAL)

Az "AXNAME" pl. általánosan érvényes ciklusok készítésénél használható, ha a tengelyek neve nem ismert

Az "AX"-t a geometria- és szinkrontengelyek közvetett programozására használják. A tengely-jelölő egy AXIS típusú változóban lesz megadva vagy egy utasítás adja, mint az "AXNAME" vagy "SPI".

Az "SPI"-t a tengelyfunkcióknak orsóra, pl. szinkron-orsóra alkalmazásánál használják.

Az "AXTOSPI"-t egy tengely-jelölőnek egy orsó indexre átalakításához használják (az "SPI" fordított funkciója).

Az "AXSTRING"-t egy tengely-jelölőnek (AXIS adattípus) string-re átalakításához használják (az "AXNAME" fordított funkciója).

Az "ISAXIS"-t általánosan érvényes ciklusokban használják annak a megállapítására, hogy egy megadott geometria-tengely létezik-e és ezzel egy következő $P_AXNX-hívás nem lesz-e hibával megszakítva.

A "MODAXVAL"-t a modulo-körtengelyeknél a modulo-pozíció megállapítására használják.

Szintaxis

```
AXNAME("string")
AX[AXNAME("string")]
SPI(n)
```

`AXTOSPI(A)` vagy `AXTOSPI(B)` vagy `AXTOSPI(C)`

```
AXSTRING(SPI(n))
ISAXIS(<geometria-tengely szám>)
<modulo pozíció>=MODAXVAL(<tengely>,<tengelypozíció>)
```

Jelentés

AXNAME:	Egy bemeneti stringet alakít át tengely-jelölőre; bemeneti string kell tartalmazzon érvényes tengelynevet.
AX:	változó tengely-jelölő
SPI:	Orsószámot alakít át tengely-jelölőre; az átadása paraméter kell egy érvényes orsószámot tartalmazzon.
n:	orsószám
AXTOSPI:	Egy tengely-jelölőt alakít át Integer típusú orsó-indexre. Az "AXTOSPI" az "SPI" funkció fordítottja.
X, Y, Z:	AXIS típusú tengely-jelölők változóként vagy állandóként
AXSTRING:	String kiadása a hozzárendelt orsószámmal.

`ISAXIS`:	Megvizsgálja, hogy a geometria-tengely létezik-e.
`MODAXVAL`:	Megállapítja egy modulo-körtengely modulo-pozícióját; ez megfelel a modulo maradék vonatkoztatva a modulo-tartományra (alap beállításban 0 ... 360 fok; a MD30340 MODULO_RANGE_START és MD30330 $MA_MODULO_RANGE által lehet a modulo-tartomány kezdetét és méretét változtatni).

Megjegyzés

SPI bővítések

Az SPI(n) tengelyfunkció használható frame-komponensek olvasására és írására is. Ezzel a frame-k pl.`$P_PFRAME[SPI(1),TR]=2.22` szintaxissal írhatók.

A tengely-pozíciók kiegészítő programozásával az `AX[SPI(1)]=<tengelypozíció>` címmel lehet egy tengelyt mozgatni. Ennek előfeltétele, hogy az orsó pozícionáló vagy tengely-üzemben van.

Példák

Példa 1: AXNAME, AX, ISAXIS

Programkód	Kommentár
`OVRA[AXNAME("Planachse")]=10`	`; síktengely override`
`AX[AXNAME("Planachse")]=50.2`	`; síktengely végpozíció`
`OVRA[SPI(1)]=70`	`; orsó 1 override`
`AX[SPI(1)]=180`	`; orsó 1 végpozíció`
`IF ISAXIS(1)==FALSE GOTOF WEITER`	`; abszcissza létezik?`
`AX[$P_AXN1]=100`	`; abszcissza mozgás`
`WEITER:`	

Példa 2: AXSTRING

Az AXSTRING[SPI(n)] programozásánál nem az orsóhoz hozzárendelt tengely tengely-indexe kiadva orsó-számként, hanem az "Sn" string lesz kiadva.

Programkód	Kommentár
`AXSTRING[SPI(2)]`	`; Az "S2" stringet adja ki.`

Példa 3: MODAXVAL

Az A modulo-körtengely modulo-pozícióját kell megállapítani.

A számítás kimeneti értéke a 372.55 tengely-pozíció.

A paraméterezett modulo-tartomány 0 ... 360 fok:

MD30340 MODULO_RANGE_START = 0

MD30330 $MA_MODULO_RANGE = 360

Programkód	Kommentár
`R10=MODAXVAL(A,372.55)`	`; Kiszámított modulo pozíció R10 = 12.55.`

Példa 4: MODAXVAL

Ha a programozott tengely-jelölő nem egy modulo-körtengelyre vonatkozik, akkor az átalakítandó érték (<tengelypozíció>) változatlanul lesz visszaadva.

Programkód	Kommentár
`R11=MODAXVAL(X,372.55)`	`; X lineáris tengely; R11 = 372.55.`

18.2 Átkapcsolható geometria-tengelyek (GEOAX)

Az "Átkapcsolható geometria-tengelyek" funkcióval helyettesíteni lehet a gépadatokban paraméterezett geometria-tengelyeket más csatornatengelyekkel.

Szintaxis

```
GEOAX(<n>,<csatornatengely>,<n>,<csatornatengely>,<n>,<csatornatenge
ly>)
GEOAX()
```

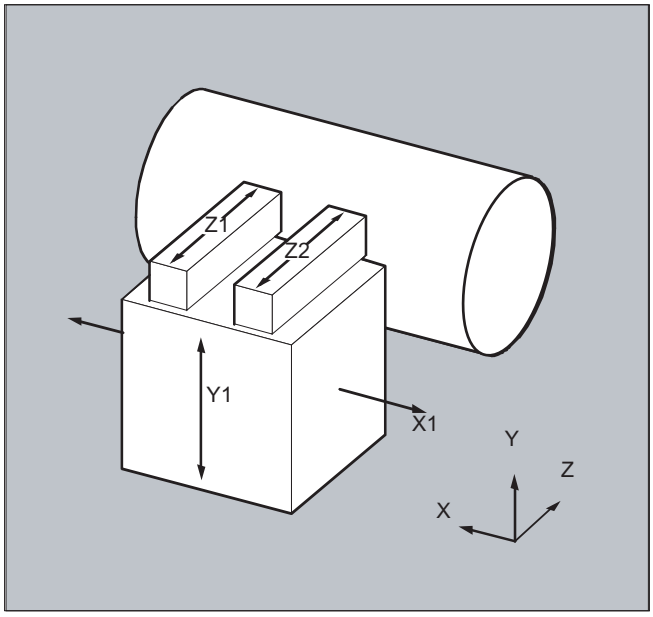
Jelentés

`GEOAX(...)`	Funkció a geometria tengelyek átkapcsolásához. **Utalás:** `GEOAX()` paraméter megadása nélkül ismét a gépadatokban paraméterezett geometria-tengelyek alap-konfigurációját aktiválja.
`<n>`	A geometria-tengely száma, amelyik a megadott csatornatengellyel helyettesítve lesz. értéktartomány: 0, 1, 2, 3 **Utalás:** 0: A megadott csatornatengely pótlás nélkül lesz eltávolítva a geometria-egyesülésből. 1: 1. geometria-tengely ≙ MKR X koordináta-tengelye (abszcissza) 2: 2. geometria-tengely ≙ MKR Y koordináta-tengelye (ordináta) 3: 3. geometria-tengely ≙ MKR Z koordináta-tengelye (aplikáta)
`<csatornatengely>`	A csatorna-tengely neve, amelyiket a fel kell venni a geometria-tengely egyesülésbe.

Példák

Példa 1: Két tengelyt váltakozva geometriai tengelyként átkapcsolni

Egy szerszámszán mozgatható az X1, Y1, Z1, Z2 tengelyekkel.



A geometria-tengelyek úgy vannak megtervezve, hogy a bekapcsolás után először a Z1 mint 3. geometria-tengely a "Z" geometria-tengely névvel hatásos és az X1-gyel és Y1-gyel egy geometria-tengely egyesülést alkotnak.

A munkadarab-programban a Z1 és Z2 tengelyek felváltva Z geometria-tengelyként kerülnek alkalmazásra.

Programkód	Kommentár
...	
N100 GEOAX(3,Z2)	; 3. geometria-tengelyként (Z) a Z2 csatornatengely szerepel.
N110 G1 ...	
N120 GEOAX(3,Z1)	; 3. geometria-tengelyként (Z) a Z1 csatornatengely szerepel.
...	

Példa 2: Geometria-tengelyek átkapcsolása 6 csatornatengely esetén

Egy gépnek 6 csatornatengelye van, XX, YY, ZZ, U ,V ,W nevekkel ellátva.

A geometria-tengely konfiguráció alapbeállítása gépadatokkal a következő:

XX csatornatengely = 1. geometria-tengely (X tengely)

YY csatornatengely = 2. geometria-tengely (Y tengely)

ZZ csatornatengely = 3. geometria-tengely (Z tengely)

Programkód	Kommentár
N10 GEOAX()	; A geometria-tengelyek alapkonfigurációja hatásos.
N20 G0 X0 Y0 Z0 U0 V0 W0	; Összes tengely gyorsmenetben 0 pozícióra.
N30 GEOAX(1,U,2,V,3,W)	; U csatornatengely lesz az első (X), V a második (Y)
	; W a harmadik geometria-tengely (Z).

Programkód	Kommentár
N40 GEOAX(1,XX,3,ZZ)	; XX csatornatengely lesz az első (X), ZZ a harmadik
	; geometria-tengely (Z). V csatornatengely marad a második
	; geometria-tengely(Y).
N50 G17 G2 X20 I10 F1000	; Teljes kör az X/Y-síkban. Az XX és V
	; csatornatengelyek mozognak.
N60 GEOAX(2,W)	; W csatornatengely lesz a második geometria-tengely (Y).
N80 G17 G2 X20 I10 F1000	; Teljes kör az X/Y-síkban. Az XX és W
	; csatornatengelyek mozognak.
N90 GEOAX()	; Visszaállás az alapállapotra.
N100 GEOAX(1,U,2,V,3,W)	; U csatornatengely lesz az első (X), V a második
	; (Y) és W a harmadik geometria-tengely (Z).
N110 G1 X10 Y10 Z10 XX=25	; U, V, W csatornatengelyek
	; a 10-es pozícióra mennek. XX mint póttengely a 25-ös pozícióra megy.
N120 GEOAX(0,V)	; V ki lesz véve a geometria-tengely egyesülésből.
	; U és W továbbra is első (X) és harmadik
	; geometria-tengely (Z).
	; A második geometria-tengely (Y) szabadon marad.
N130 GEOAX(1,U,2,V,3,W)	; U csatornatengely marad az első (X), V lesz a
	; második (Y), W marad a harmadik geometria-tengely (Z).
N140 GEOAX(3,V)	; V lesz a harmadik geometria-tengely (Z), ezáltal W
	; felül lesz írva és ki lesz véve a geometria-tengely
	; egyesülésből. ; A második geometria-tengely (Y)
	; továbbra is szabadon marad.

Gépadatok

Tengely-konfiguráció

Geometria-, kiegészítő- és géptengelyek hozzárendelése a csatornatengelyekhez:

- MD10000 $MN_AXCONF_MACHAX_NAME_TAB
- MD20050 $MC_AXCONF_GEOAX_ASIGN_TAB
- MD20060 $MC_AXCONF_GEOAX_NAME_TAB
- MD20070 $MC_AXCONF_MACHAX_USED
- MD20080 $MC_AXCONF_CHANAX_NAME_TAB
- MD35000 $MA_SPIND_ASSIGN_TO_MACHAX

Reset viselkedés

A megváltoztatott geometria hozzárendelések Reset viselkedése:

- MD20110 $MC_RESET_MODE_MASK, bit 12
- MD20118 $MC_GEOAX_CHANGE_RESET

NC-Start viselkedés

- MD20112 $MC_START_MODE_MASK, bit 12

Közlés a PLC alkalmazói programmal

Az M utasítás paraméterezési lehetősége, amelyik egy geometria-tengely átkapcsolásnál az NC/PLC interfészre ki lesz adva:

- MD22532 $MC_GEOAX_CHANGE_M_CODE

Peremfeltételek

Nincs geomeria-tengely átkapcsolás

- Ha a következő funkciók egyike aktív, a geomeria-tengely átkapcsolás nem lehetséges.
  - Transzformáció
  - Spline interpoláció
  - szerszámsugár-korrekció
  - szerszám finom-korrekció
- A geometria-tengelynek és egy másik csatornatengelynek azonos a neve.
- A geomeria-tengely átkapcsolásban résztvevő tengelyek egyike egy olyan akcióban érintett, amely a mondathatárokon át tart. Pl. Mondatokat átfogó pozícionáló tengely vagy egy tengely-csatolás követő-tengelye.

Körtengelyek

Körtengelyeket nem lehet geometria-tengelyekké tenni.

Tengelyállapot a helyettesítés után

Egy, a geometria-tengely egyesülésben való átkapcsolással helyettesített tengely az átkapcsolási eljárás után a csatornatengely-névvel kiegészítő tengelyként programozható.

Frame-k, védelmi tartományok, munkatér-határolások

A geometria-tengelyek átkapcsolásával minden frame, védelmi tartomány és munkatér-határolás ki lesz törölve.

Polár-koordináták

Egy geometria-tengely csere `GEOAX`X-szal a `G17`-`G19` síkváltással analóg módon a modális polár-koordinátákat 0 értékre állítja.

DRF, NPE

Egy esetleges kézikerék-eltolás (DRF) vagy egy külső nullapont-eltolás (NPE) az átkapcsolás után hatásos marad.

A geometria-tengelyek alap-konfigurációja

A `GEOAX()` utasítás a geometria-tengely egyesülés alap-konfigurációját hívja fel.

POWER ON után és "Referenciapontra menet" üzemmódra való átkapcsolásnál automatikusan vissza lesz kapcsolva az alap-konfigurációra.

Szerszámhossz-korrekció

Egy aktív szerszámhossz-korrekció az átkapcsolási eljárás után is hatásos. Azonban az újként felvett ill. pozícióban cserélt geometria-tengelyekre még nem megtettnek számít. Ennek megfelelően az első mozgásparancsnál ezen geometria-tengelyekre az eredő mozgásút a szerszámhossz-korrekció és a programozott mozgásút összegéből adódik.

Geometria-tengelyek, amelyek egy átkapcsolásnál megtartják a tengely-egyesülésbeli pozíciójukat, a szerszámhossz-korrekcióra vonatkozó állapotukat is megtartják.

Geometria-tengely konfiguráció aktív transzformációnál

- Egy aktív transzformációra transzformáció-gépadattal paraméterezett geometria-tengely konfiguráció az "Átkapcsolható geometria-tengelyek" funkcióval nem változtatható meg.
- Egy transzformáció különböző geometria-tengely konfigurációihoz a transzformációs gépadatokban különböző adatkészleteket kell paraméterezni.
- Egy `GEOAX`-szal megváltoztatott geometria-tengely konfiguráció egy transzformáció aktiválásával törölve lesz.
- A geometria-tengelyek vonatkozásában az aktív transzformációk transzformáció-specifikus geometria-tengely paraméterezéseinek elsőbbsége van a geometria-tengelyek átkapcsolását érintő paraméterezések előtt.
  Példa: Egy transzformáció aktív A gépadatok szerint a transzformáció csatorna Reset-nél meg kell maradjon. Azonban ezzel egyidőben egy csatorna Reset-nél a geometria-tengelyek alapkonfigurációját elő kell állítani. A geometria-tengely konfiguráció megmarad, ahogy a transzformációra meg van adva.
- A transzformáció kikapcsolásával ismét a geometria-tengely konfiguráció paraméterezett alaphelyzete lesz hatásos.

JOG üzemmód, REF gépfunkció

JOG üzemmód, REF gépfunkcióra (referenciapontra menet) átkapcsolásnál a gépadatokban paraméterezett geometria-tengely konfiguráció lesz hatásos.

18.3 Tengely-konténer (AXCTSWE, AXCTSWED, AXCTSWEC)

Az "AXCTSWE" ill. "AXCTSWED" utasításokkal engedélyezve lesz a megadott tengely-konténer forgatása.

Az "AXCTSWEC" utasítással egy már megadott tengely-konténer forgatás engedélyezését vissza lehet venni.

Szintaxis

```
AXCTSWE(<ID>)
AXCTSWED(<ID>)
AXCTSWEC(<ID>)
```

Jelentés

AXCTSWE:	engedély a tengely-konténer forgatására A program feldolgozása a "AXCTSWE"-vel nem áll meg. A forgatás megtörténik, mihelyt a tengely-konténerben résztvevő csatornák mindegyike megkapta az engedélyt.	
AXCTSWED:	engedély a tengely-konténer forgatására a tengely-konténerben résztvevő többi csatorna figyelembe vétele nélkül **Utalás** • Utasítás változat a munkadarabprogram ill. a szinkronakciók üzembehelyezésének egyszerűsítésére.. • A tengely-konténerben résztvevő többi csatornával kapcsolatos viselkedést meg lehet adni: MD12760 $MN_ AXCT_FUNCTION_MASK, bit 0	
AXCTSWEC:	engedély visszavétele a tengely-konténer forgatására **Utalás** Egy tengely-konténer forgatás engedélyezését csak akkor lehet visszavenni, ha a forgatás még nem kezdődött el: $AN_AXCTSWA[<tengely-konténer>] == 0 Rendszerváltozóhoz lásd "Tengely-konténer (AXCTSWE, AXCTSWED, AXCTSWEC) (Oldal 637)"	
<ID>:	tengely-konténer jelölő vagy egy konténer-tengely:	
	CT<szám>:	egy tengely-konténer jelölő alapesete: MD12750 $MN_AXCT_NAME_TAB Példa: "CT1"
	<konténer>:	egy tengely-konténer felhasználó-specifikus jelölője: MD12750 $MN_AXCT_NAME_TAB Példa: "CONTAINER_1"
	<tengely>:	egy, a csatornában ismert konténer-tengely jelölője

Megjegyzés

Lépéshossz

Egy tengely-konténer forgatás lépéshosszának megadása beállítási adattal:

SD41700 $SN_AXCT_SWWIDTH

További információk

Diagnózis

Egy tengely-konténer aktuális csatolási állapotát a következő rendszerváltozóval lehet olvasni:

Rendszerváltozó	típus	leírás
$AC_AXCTSWA[<név>]	BOOL	tengely-konténer csatorna-specifikus állapota
$AN_AXCTSWA[<tengely-konténer>]	BOOL	tengely-konténer NCU-specifikus állapota
$AN_AXCTSWE[<tengely-konténer>]	INT	tengely-konténer forgatás slot-specifikus állapota A rendszerváltozó **bitenként** adja meg a tengely-konténer slot állapotát. Minden bit egy slot-nak felel meg.
$AN_AXCTAS[<tengely-konténer>]	INT	Helyek (slotok) száma, amivel a tengely-konténer aktuálisan tovább lesz kapcsolva.

Tengely-konténer forgatása implicit GET/GETD-vel

A következő gépadattal be lehet állítani, hogy az "AXCTSWE" utasítással a csatorna összes konténer-tengelye közvetve "GET/GETD" által a csatornába lesz hozva. A tengelycsere csak a konténer forgatás megtörténte után lehetséges ismét.

MD10722 $MN_AXCHANGE_MASK, bit 1 = 1

Megjegyzés

A tengely-konténer forgatás implicit "GET/GETD"-vel egy "főfutás-tengely" állapotban levő tengelyre (pl. PLC-tengely) **nem** alkalmazható, mivel ezt az állapotot a tengelyt a tengely-konténer forgatáshoz el kellene hagyja.

18.4 Várni érvényes tengelypozícióra (WAITENC)

A "WAITENC" utasítással az NC programban lehet várni, amíg az MD34800 $MA_WAIT_ENC_VALID = 1-gyel beállított tengelyekre a szinkronizált ill. helyreállított tengelypozíciók rendelkezésre állnak.

A várakozási állapotban lehet egy megszakítás, pl. egy ASUP indításával vagy üzemmód váltással JOG-ba. A program folytatásával a várakozási állapot adott esetben újra fel lesz véve.

Megjegyzés

A várakozási állapot a kezelőfelületen a "Várakozás mérőrendszerre" állapottal ki lesz jelezve.

Szintaxis

A "WAITENC"-t egy tetszőleges NC program programrészében lehet programozni.

A programozás egy önálló mondatban kell legyen.

```
...
WAITENC
...
```

Példa

A "WAITENC" pl. az .../_N_CMA_DIR/_N_PROG_EVENT_SPF esemény-vezérelt felhasználói programban lesz használva a következő alkalmazási példa szerint.

Alkalmazási példa: Szerszám visszahúzás POWER OFF után tájolási transzformációval

Egy megmunkálás szerszámtájolással feszültség kimaradás miatt megszakadt.
Az ezt követő felfutásnál az .../_N_CMA_DIR/_N_PROG_EVENT_SPF esemény-vezérelt felhasználói program lesz felhívva.

Az esemény-vezérelt felhasználói programban "WAITENC"-vel várakozás van a szinkronizált ill helyreállított tengelypozíciókra, ami után egy frame lesz kiszámítva, ami az MKR-t a szerszámirányba állítja be.

Programkód	Kommentár
...	
IF $P_PROG_EVENT == 4	; Felfutás.
IF $P_TRAFO <> 0	; A transzformáció ki lett kapcsolva.
WAITENC	; Várakozás a tájolási tengelyek érvényes tengelypozícióra.
TOROTZ	; MKR Z tengelyét a szerszámtengely irányába forgatni.
ENDIF	
M17	
ENDIF	
...	

Ezután lehet a szerszámot JOG üzemmódban egy visszahúzási mozgással a szerszámtengely irányában szabadra vinni.

18.5 Programozható paraméterkészlet-átkapcsolás (SCPARA)

A "SCPARA" utasítással lehet igényelni egy tengelyre az átkapcsolást egy adott paraméterkészletre.

Megjegyzés

Nincs paraméterkészlet-átkapcsolás menet megmunkálás közben

A G33 menetvágásnál és a G331 / G332 menetfúrásnál a paraméterkészletet a vezérlés választja ki és azt nem lehet megváltoztatni.

Zárolt paraméterkészlet-átkapcsolás

A paraméterkészlet-átkapcsolást az NC/PLC interfészen keresztül is lehet igényelni. Az átkapcsolási konfliktus elkerülésére az NC/PLC interfészen keresztül az NC paraméterkészlet-átkapcsolását (`SCPARA`) lehet zárolni:

DB31, ... DBX9.3 (paraméterkészlet megadása az NC-ből zárolva)

Megjegyzés

Ha "SCPARA"-val egy paraméterkészlet-átkapcsolás van igényelve, miközben a paraméterkészlet-átkapcsolás az NC/PLC interfészen tiltva van, az átkapcsolás hibajelzés nélkül lesz elutasítva.

Szintaxis

```
SCPARA[<tengely>]=<érték>
```

Jelentés

`SCPARA`:	Utasítás: paraméterkészlet-átkapcsolás	
`<tengely>`:	tengely jelölő (csatornatengely)	
	típus:	AXIS
`<érték>`:	paraméterkészlet-szám 1, 2, 3, ... max. paraméterkészlet-szám	

Példa

Programkód	**Kommentár**
...	
N110 SCPARA[X]= 3	; Kiválasztás: X tengely, 3. paraméterkészlet
...	

További információk

Paraméterkészlet-átkapcsolás engedélyezése

A tengely paraméterkészlet-átkapcsolását kifejezetten engedélyezni kell:

MD35590 $MA_PARAMSET_CHANGE_ENABLE[<tengely>]

Paraméterkészlet-számot olvasni

A kiválasztott paraméterkészlet számát (parancs paraméterkészlet-szám) a $AA_SCPAR rendszerváltozóból lehet kiolvasni.

Irodalom

A paraméterkészletekhez részletes információk találhatók:

Alapfunkciók működési kézikönyv; "Sebességek, parancs-/valósérték rendszer, Szabályozás (G2)" >, "Helyzetszabályzó paraméterkészletek" fejezet

18.6 NC-nyelv meglevő terjedelmének vizsgálata (STRINGIS)

A "STRINGIS(...)" funkcióval meg lehet vizsgálni, hogy a megadott string az NC programnyelv aktuális nyelvi terjedelmében rendelkezésre áll-e.

Definíció

```
INT STRINGIS(STRING <név>)
```

Szintaxis

```
STRINGIS(<név>)
```

Jelentés

STRINGIS:	funkció visszaadási értékkel
<név>:	NC programnyelv megvizsgálandó elemének a neve
visszaadási érték	A visszaadási érték formátuma yxxx (decimális).

Magas szintű NC nyelvek elemei

Az NC programnyelv következő elemeit lehet megvizsgálni:

- az összes létező G-csoport G utasításai, pl. "G0", "INVCW", "POLY", "ROT", "KONT", "SOFT", "CUT2D", "CDON", "RMBBL", "SPATH"
- DIN vagy NC címek mint pl. ADIS", "RNDM", "SPN", "SR", "MEAS"
- funkciók pl. "TANG(...)" vagy "GETMDACT"
- eljárások pl. "SBLOF".
- kulcsszavak pl. "ACN", "DEFINE" vagy "SETMS"
- rendszeradatok pl. $M... gépadatok, $S... beállítási adatok vagy $O... opciós adatok
- rendszerváltozók $A... , $V... , $P...
- számítási paraméterek R...
- aktív ciklusok ciklus-nevei
- GUD vagy LUD változók
- makró-nevek
- címke-nevek

Visszaadási érték

A visszaadási értéknek csak az első 3 decimális értéke számít. A visszaadási érték formátuma yxxx, ahol y = alap információ és xx = részletes információ

Visszaadási érték	Jelentés
000	A 'név' stringet a rendszer nem ismeri [1)]
100	A 'név' string eleme az NC programnyelvnek, de aktuálisan nem programozható (opció / nem aktív funkció)

Visszaadási érték	Jelentés	
2xx	A 'név' string egy programozható eleme az NC programnyelvnek (opció / aktív funkció) Az xx részletes információk további információkat tartalmaznak az elem fajtájáról:	
	xx	Jelentés
	01	DIN cím vagy NC cím [2)]
	02	G utasítás (pl. G04, INVCW)
	03	funkció visszaadási értékkel
	04	funkció visszaadási érték nélkül
	05	kulcsszó (pl. DEFINE)
	06	gépadatok ($M...), beállítási adatok ($S...) vagy opciós adatok ($O...)
	07	rendszer-paraméterek, pl. rendszerváltozók ($...) vagy számítási paraméterek (R...)
	08	ciklus (A ciklus az NC-be betöltve és a ciklus-programok aktiválva kell legyenek[3)])
	09	GUD-változó (a GUD változó a GUD definíciós fájlban definiálva és a GUD-változók aktiválva kell legyenek)
	10	makró-név (a makró a makró definíciós fájlban definiálva és a makrók aktiválva kell legyenek) [4)]
	11	LUD-változója az aktuális munkadarabprogramnak
	12	ISO G utasítás (ISO nyelvi modus aktív kell legyen)
400	A 'név' string egy NC cím, amely nem xx == 01 vagy xx == 10 -ként lett felismerve és nem G vagy R [2)]	
y00	nem lehetséges specifikus hozzárendelés	

1) A vezérléstől függően adott körülmények között a Siemens NC nyelv utasításoknak csak egy része ismert, pl. SINUMERIK 802D sl. Ezeken a vezérléseken azokra a stringekre, amelyek elvileg Siemens NC nyelvi utasítások, a 0 érték lesz visszaadva.. Ezt a viselkedést az MD10711 $MN_NC_LANGUAGE_CONFIGURATION által meg lehet változtatni. Az MD10711 = 1 esetén a Siemens NC nyelvi utasításokra mindig a 100-as érték lesz visszaadva.

2) az NC címek a következő betűk: A, B, C, E, I, J, K, Q, U, V, W, X, Y, Z. Ezeket az NC címeket lehet cím-bővítéssel is programozni. A cím-bővítést a STRINGIS vizsgálatnál meg lehet adni.. Példa: 201 == STRINGIS("A1").
A: D, F, H, L, M, N, O, P, S, T betűk NC címek vagy segédfunkciók, amelyeket felhasználó által definiáltan lehet használni. Ezekre mindig a 400-as érték lesz visszaadva. Példa: 400 == STRINGIS("D"). Ezeket az NC címeket a STRINGIS vizsgálatnál nem lehet cím-bővítéssel megadni..
Példa: 000 == STRINGIS("M02"), de 400 == STRINGIS("M").

3) Ciklus-paraméter neveket nem lehet STRINGIS-szel vizsgálni.
4) A makróként definiált G, H, L, M címek makróként lesznek azonosítva

Példák

A következő példákban feltételezzük, hogy a stringként megadott NC nyelvi elemek, hacsak nincs külön jelezve, a vezérlésben elvileg programozhatóak.

1. A "T" string segédfunkcióként definiálva:
```
400 == STRINGIS("T")
000 == STRINGIS ("T3")
```
2. Az "X" string tengelyként van definiálva:
```
201 == STRINGIS("X")
201 == STRINGIS("X1")
```
3. Az "A2" string NC címként van definiálva bővítéssel:
```
201 == STRINGIS("A")
201 == STRINGIS("A2")
```
4. Az "INVCW" string elnevezett G utasításként van definiálva:
```
202 == STRINGIS("INVCW")
```

5. Az "$MC_GCODE_RESET_VALUES" string gépadatként van definiálva:
   `206 == STRINGIS("$MC_GCODE_RESET_VALUES")`
6. A "GETMDACT" string egy NC nyelvi funkció:
   `203 == STRINGIS("GETMDACT ")`
7. A "DEFINE" string egy kulcsszó:
   `205 == STRINGIS("DEFINE")`
8. A "$TC_DP3" string egy rendszerparaméter (szerszámhossz-komponens):
   `207 == STRINGIS("$TC_DP3")`
9. A "$TC_TP4" string egy rendszerparaméter (szerszámméret):
   `207 == STRINGIS("$TC_TP4")`
10. A "$TC_MPP4" string egy rendszerparaméter (tárhely állapot):
    - szerszámtár kezelés aktív `207 == STRINGIS("$TC_MPP4") ;`
    - szerszámtár kezelés nem aktív: `000 == STRINGIS("$TC_MPP4")`

    lásd : szerszámtár kezelés
11. A "MACHINERY_NAME" string GUD-változóként van definiálva:
    `209 == STRINGIS("MACHINERY_NAME")`
12. A "LONGMACRO" string makróként van definiálva:
    `210 == STRINGIS("LONGMACRO")`
13. A "MYVAR" string LUD-változóként van definiálva:
    `211 == STRINGIS("MYVAR")`
14. Az "XYZ" az NC-ben ismeretlen utasítás, GUD-változó, makró vagy ciklus-név:
    `000 == STRINGIS("XYZ")`

Szerszámtár kezelés

Ha a szerszámtár kezelés funkció nem aktív, a STRINGIS a szerszámtár kezelés rendszerparamétereire, függetlenül a

- MD10711 $MN_NC_LANGUAGE_CONFIGURATION

gépadattól mindig 000 értéket ad.

ISO modus

Ha az "ISO modus" funkció aktív:

- MD18800 $MN_MM_EXTERN_LANGUAGE (külső NC nyelvek aktiválása)
- MD10880 $MN_ MM_EXTERN_CNC_SYSTEM (adaptálandó vezérlőrendszer)

STRINGIS a megadott stringet először SINUMERIK G utasításként vizsgálja meg. Ha a string nem SINUMERIK G utasítás, utána ISO G utasításként lesz megvizsgálva.

A programozott átkapcsolásoknak (`G290` (SINUMERIK modus), `G291` (ISO modus)) a STRINGIS-re nincs hatásuk.

Példa

A STRINGIS(...) funkció szempontjából számító gépadatok értékei a következők:

- MD10711 $MN_NC_LANGUAGE_CONFIGURATION = 2 (csak azok az NC nyelvi elemek lesznek ismertnek tekintve, amelyek opciója be van állítva)
- MD19410 $ON_TRAFO_TYPE_MASK = 'H0' (opció: transzformációk)
- MD10700 $MN_PREPROCESSING_LEVEL='H43' (ciklusok elő-feldolgozása aktív)

A következő példaprogram hibajelzés nélkül lesz végrehajtva:

Programkód	**Kommentár**
`N1 R1=STRINGIS("TRACYL")`	`; R1 == 0, mivel a TRACYL a hiányzó transzformáció opció miatt "nem ismert"-ként lesz azonosítva.`
`N2 IF STRINGIS("TRACYL") == 204`	
`N3  TRACYL(1,2,3)`	`; N3 át lesz ugorva`
`N4 ELSE`	
`N5  G00`	`; és helyette N5 végrehajtva`
`N6 ENDIF`	
`N7 M30`	

18.7 Ablakot munkadarabprogramból interaktívan felhívni (MMC)

Az előre definiált MMC(...) alprogrammal egy NC programból lehet felhasználó-specifikus dialógusokat a kezelőfelületen pl. SINUMERIK Operate.

A dialógusok létrehozása a következő dialógus formákra történhet:

- Run MyScreens
- Easy XML
- felhasználói XML

Irodalom

- Run MyScreens programozási kézikönyv
- Easy XML programozási kézikönyv

Szintaxis

```
MMC("<ADDRESS>,<COMMAND>,<FILE>,<DIALOG>","<QUIT>")
```

Jelentés

MMC(...):	alprogram jelölő A paraméterek hely-kódolva és vesszővel elválasztva lesznek megadva két stringen, a parancs stringen és a nyugtázási stringen belül.	
Paraméterek a parancs stringen belül:		
<ADDRESS>:	kezelési-tartomány, amelyikben a megadott felhasználói dialógusok végre lesznek hajtva	
	Funkció	**Kezelési tartományok**
	"Run MyScreens" felhasználói dialógus	CYCLES
	"Easy XML" felhasználói dialógus	CYCLES
	felhasználói XML	XML
	Popup "Run MyScreens"	POPUPDLG
	Popup "Easy XML"	POPUPDLG
<COMMAND>:	végrehajtandó parancs	
	Funkció	**Parancsok**
	"Run MyScreens" felhasználói dialógus	PICTURE_ON, PICTURE_OFF
	"Easy XML" felhasználói dialógus	PICTURE_ON, PICTURE_OFF
	felhasználói XML	XML_ON, XML_OFF
	Popup "Run MyScreens"	PICTURE_ON, PICTURE_OFF
	Popup "Easy XML"	PICTURE_ON, PICTURE_OFF

<FILE>:	A fájl neve, amelyben a kijelzendő dialógus programozva van.	
	Funkció	**Fájlok**
	"Run MyScreens" felhasználói dialógus	<name>.com
	"Easy XML" felhasználói dialógus	<name>.xml
	felhasználói XML	<name>.xml
	Popup "Run MyScreens"	<name>.com
	Popup "Easy XML"	<name>.xml
	Popup "Easy XML" beállítással közvetlenül az NC programban (lásd példa 2)	xmldial_emb.xml
<DIALOG>:	kijelzendő dialógus neve	
	Funkció	**Dialógus név**
	Összes funkció a PopUp "Easy XML" kivételével beállítással közvetlenül az NC programban	dialógusnak a <FILE>-ban beállított neve
	Popup "Easy XML" beállítással közvetlenül az NC programban (lásd példa 3)	main
Paraméterek a nyugtázás stringen belül:		
<QUIT>:	nyugtázás típus	
	N:	Nincs nyugtázás. A programmegmunkálás az utasítás kiküldése után folytatódik.. Nincs visszajelzés akkor, ha az utasítást nem lehetett sikeresen végrehajtani. **Utalás** Nyugtázás típus "N"-t kell használni, ha az NC program egy kijelzési idő (várakozási idő) van programozva (lásd lent példa 2-ben)
	A:	Aszinkron nyugtázás A program feldolgozása az utasítás kiküldése után folytatódik.. A visszaadási érték egy, a dialógus létrehozás kertében definiált felhasználó-specifikus nyugtázási változóban (GUD változó) lesz eltárolva és az NC programban olvasható.

Példa

Példa 1

Egy dialógus kijelzése és reakció a felhasználói kezelésre egy NC programban

```
Programkód                                  Kommentár
; A QUIT nyugtázási változó már globális felhasználói változóként (GUD)
; STRING típussal a dialógus beállításánál létre lett hozva:
; DEF NCK STRING[20] QUIT
QUIT = "XXX"                                ; nyugtázási változót inicializálni
G4 F5
```

```
Programkód                                  Kommentár
MMC("CYCLES,PICTU-                          ; dialógust kijelezni
RE_ON,test.com,test1","A")                  ;  - kezelési tartomány CYCLES
                                            ;  - kép állapot: PICTURE_ON (kijelezni)
                                            ;  - fájl dialógus kép: test.com
                                            ;  - dialógus kép: test1
INPUT:                                      ; várni Compile ciklusra
   STOPRE                                   ;    előrefutás-állj
   IF MATCH (QUIT,"RUN") >= 0 GOTOF WORK    ;    "RUN" softkey
   IF MATCH (QUIT,"CHK") >= 0 GOTOF CHECK   ;    "CHK" softkey
GOTOB INPUT                                 ; => várni

WORK:                                       ; "RUN" softkey megnyomva
MSG("tovább megmunkálással -> NC-Start")    ; jelentést kiadni
MMC("CYCLES,PICTURE_OFF","N")               ; dialógust bezárni
M0                                          ; várni NC-Start-ra
GOTOF END                                   ; => programvég

CHECK:                                      ; "CHK" softkey megnyomva
MSG("pozíciót felvenni -> NC-Start")        ; jelentést kiadni
MMC("CYCLES,PICTURE_OFF","N")               ; dialógust bezárni
M0                                          ; várni NC-Start-ra
GOTOF END                                   ; => programvég

END:
...
```

Példa 2

Egy dialógus kijelzési ideje az NC programban, pl. egy várakozási idővel lesz megadva.

```
Programkód                                          Kommentár
F1000 G94
...
MMC("POPUPDLG,PICTURE_ON,xmldial_emb.xml,main","N")   ; dialógust kijelezni
X200
Z40
MMC("POPUPDLG,PICTURE_OFF","N")                       ; dialógust bezárni
```

Példa 3

Egy PopUp-Script beágyazása egy NC programba és annak alkalmazása.

Programkód

```
PROC POPUP_TEST
; ----------------------------- Script -----------------------------
; <main_dialog entry="rpara_main">
;    <let name="xpos" />
```

Programkód

```
;    <let name="ypos" />
;    <let name="field_name" type="string" />
;    <let name="num" />
;    <menu name="rpara_main">
;       <open_form name="rpara_form"/>
;       <softkey_back>
;          <close_form />
;       </softkey_back>
;    </menu>
;    <form name="rpara_form">
;       <init>
;          <caption>mask from NC part program</caption>
;          <let name="count" >0</let>
;          <op>
;             xpos = 120;
;             ypos = 34;
;             "nck/Channel/Parameter/R[10]" = 10;
;          </op>
;          <!-- load the number of controls -->
;          <op>
;             num = "nck/Channel/Parameter/R[10]";
;          </op>
;          <while>
;             <condition> count < num</condition>
;             <print name="field_name" text="edit%d">count</print>
;             <op>
;                ypos = ypos + 24;
;                count = count + 1;
;             </op>
;          </while>
;       </init>
;       <paint>
;          <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;
```

Programkód

```
;             </op>
;          </while>
;       </paint>
;    </form>
; </main_dialog>
; ========================= programrész ==========================
...
G94 F100
MMC("POPUPDLG,PICTURE_ON,xmldial_emb.xml,main","N")
G4 F4
X200
MMC("POPUPDLG,PICTURE_OFF","N")
G4 F2
X0
...
```

Peremfeltételek

- A dialógus *.com definíciós fájljai a "proj" könyvtárban kell legyenek.
- A dialógus Easy XML *.xml definíciós fájljai az "appl" könyvtárban kell legyenek.
  Ha a definíciós fájlok egy másik könyvtárban vannak tárolva, az ág-megadás közvetetten, az "appl" könyvtárból kiindulva kell történjen.
- A felhasználó által definiált dialógusokat nem lehet egyidejűleg különböző csatornákból kijelezni.

18.8 Program futásidő / munkadarab számláló

A szerszámgépek kezelőinek támogatására információk kaphatók a program futásidőről és a munkadarabok számáról.

Ezek az információk rendszerváltozókként az NC és/vagy PLC programban feldolgozhatóak. Egyidejűleg ezek az információk a kezelőfelület számára rendelkezésre állnak.

18.8.1 Program futásidő

A "Program futásidő" funkció NC belső időzítést bocsájt rendelkezésre a technológiai folyamatok felügyeletére, amelyet NC- és csatorna-specifikus rendszerváltozóként a munkadarabprogramban és a szinkron-akciókban olvasni és írni lehet.

A futásidő mérés trigger ($AC_PROG_NET_TIME_TRIGGER) a funkció egyetlen írható rendszerváltozója és a programszakaszok szelektív mérését szolgálja. Azaz a trigger írásával az NC programban az időmérést lehet be- és újra kikapcsolni.

Rendszerváltozó	Jelentés	Aktivitás
NC-specifikus		
$AN_SETUP_TIME	az utolsó vezérlés felfutás alapértékekkel ("hideg indítás") óta eltelt idő percekben Minden vezérlés felfutás alapértékekkel alkalmával automatikusan "0"-ra lesz visszaállítva.	• mindig aktív
$AN_POWERON_TIME	az utolsó normális vezérlés felfutás ("meleg indítás") óta eltelt idő percekben Minden normális vezérlés felfutás alkalmával automatikusan "0"-ra lesz visszaállítva.	
csatorna-specifikus		
$AC_OPERATING_TIME	NC-programok össz-futásideje Automatika üzemmódban másodpercekben Az érték minden vezérlés felfutás alkalmával automatikusan "0"-ra lesz visszaállítva.	• aktiválás MD27860-nal • csak AUTOMATIK üzemmód
$AC_CYCLE_TIME	kiválasztott NC-program futásideje másodpercekben Az érték egy új NC program indításával automatikusan "0"-ra lesz visszaállítva.	
$AC_CUTTING_TIME	megmunkálási idő másodpercekben A pályatengelyek (legalább egy aktív) futásideje lesz mérve aktív gyorsmenet nélkül az összes NC programban az NC-Start és a program vége / NC-Reset között. A mérés az aktív várakozási időnél meg lesz szakítva. Az érték minden vezérlés felfutás alapértékekkel alkalmával automatikusan "0"-ra lesz visszaállítva.	

Rendszerváltozó	Jelentés		Aktivitás
$AC_ACT_PROG_NET_TIME	kiválasztott NC-program aktuális nettó futásideje másodpercekben Egy új NC program indításával automatikusan "0"-ra lesz visszaállítva.		• mindig aktív • csak AUTOMATIK üzemmód
$AC_OLD_PROG_NET_TIME	az éppen helyesen M30-cal befejezett program nettó futásideje másodpercekben		
$AC_OLD_PROG_NET_TIME_COUNT	változások az $AC_OLD_PROG_NET_TIME-ra POWER ON után az $AC_OLD_PROG_NET_TIME_COUNT értéke "0". A $AC_OLD_PROG_NET_TIME_COUNT mindig akkor lesz növelve, amikor a vezérlés a $AC_OLD_PROG_NET_TIME-t újra írta.		
$AC_PROG_NET_TIME_TRIGGER	trigger futásidő méréshez		• csak AUTOMATIK üzemmód
	0	semleges állapot A trigger nem aktív.	
	1	befejezni Befejezi a mérést és másolja az értéket az $AC_ACT_PROG_NET_TIME-ból az $AC_OLD_PROG_NET_TIME-ba. Az $AC_ACT_PROG_NET_TIME "0"-ra lesz beállítva és azután újra fut.	
	2	indítani Elindítja a mérést és ennél a $AC_ACT_PROG_NET_TIME-t "0"-ra állítja. Az $AC_OLD_PROG_NET_TIME nem változik.	
	3	megállítani Megállítja a mérést. Nem változtatja meg az $AC_OLD_PROG_NET_TIME-t és a $AC_ACT_PROG_NET_TIME-t a folytatásig állandóan tartja.	
	4	folytatni Folytatni a mérést, vagyis egy korábban megállított mérés újra fel lesz véve. Az $AC_ACT_PROG_NET_TIME tovább fut. Az $AC_OLD_PROG_NET_TIME nem változik.	
A POWER ON által az összes rendszerváltozó "0'-ra lesz visszaállítva!			

Megjegyzés

Gépgyártó

Az aktiválható időzítés bekapcsolása az MD27860 $MC_PROCESSTIMER_MODE gépadattal történik.

Az aktív időmérés viselkedése bizonyos funkcióknál (pl. GOTOS, override = 0%, aktív próbafutás-előtolás, programteszt, ASUP, PROG_EVENT, …) az MD27850 $MC_PROG_NET_TIMER_MODE és MD27860 $MC_PROCESSTIMER_MODE gépadatokkal lesz konfigurálva.

Irodalom:
Funktionshandbuch Grundfunktionen; BAG, Kanal, Programmbetrieb, Reset-Verhalten (K1), Kapitel: Program futásidő

Megjegyzés

Maradékidő egy munkadarabra

Ha egymás után ugyanaz a munkadarab lesz gyártva, a következő időzítés értékekből meg lehet állapítani a hátralevő maradékidőt egy munkadarabra:

- megmunkálási idő az utoljára gyártott munkadarabra (lásd $AC_OLD_PROG_NET_TIME)
- aktuális megmunkálási idő (lásd $AC_ACT_PROG_NET_TIME)

A maradékidő az aktuális megmunkálási idő mellett a kezelőfelületen ki lesz jelezve.

Megjegyzés

STPRE használata

A $AC_OLD_PROG_NET_TIME és $AC_OLD_PROG_NET_TIME_COUNT rendszerváltozók nem okoznak közvetett előrefutás-álljt. A munkadarabprogramban való alkalmazásnál ez nem kritikus, mert a rendszerváltozók értéke az előző programfutásból származik. Ha azonban a futásidő mérés trigger ($AC_PROG_NET_TIME_TRIGGER) magas frekvenciával lesz írva és ezáltal a $AC_OLD_PROG_NET_TIME nagyon gyakran változik, akkor a munkadarabprogramban egy közvetlen `STOPRE`-t kell használni.

Peremfeltételek

- **Mondatkeresés**
  A mondatkeresésénél nem lesz program futásidő megállapítva.
- **REPOS**
  Egy REPOS folyamat ideje az aktuális megmunkálási időbe ($AC_ACT_PROG_NET_TIME) be lesz számítva.

Példák

Példa 1: A "mySubProgrammA" időtartamát megmérni

Programkód
`...`

Programkód

```
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
```

Miután a program az N80 sort feldolgozta, a $AC_OLD_PROG_NET_TIME-ban "mySubProgrammA" nettó futásideje van.

A $AC_OLD_PROG_NET_TIME értéke:

- az M30 után is megmarad.
- minden köszörülés átfutás után aktualizálva lesz.

Példa 2: A "mySubProgrammA" és a "mySubProgrammC" időtartamát megmérni

Programkód

```
...
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 Munkadarab számláló

A "Munkadarab számláló" funkció számlálókat bocsát rendelkezésre, amelyeket pl. a munkadarabok vezérlésen belüli számlálására lehet használni.

Ezek számlálók csatorna-specifikus rendszerváltozók írás és olvasás hozzáféréssel, 0 ... 999 999 999 értéktartománnyal.

Rendszerváltozó	Jelentés
$AC_REQUIRED_PARTS	igényelt munkadarabok száma (munkadarab-igény) Ebben a számlálóban lehet a munkadarabok számát megadni, ami elérésénél az aktuális munkadarabszám ($AC_ACTUAL_PARTS) "0"-ra lesz állítva.
$AC_TOTAL_PARTS	összesen előállított munkadarabok száma (tényleges össz-munkadarabszám) Ez a számláló megadja a kezdés időponttól előállított összes munkadarab számát. Az érték csak vezérlés felfutás alapértékekkel alkalmával automatikusan "0"-ra lesz visszaállítva.

Rendszerváltozó	Jelentés
$AC_ACTUAL_PARTS	előállított munkadarabok száma (tényleges munkadarabszám) Ez a számláló megadja a kezdés időponttól előállított munkadarabok számát. A kívánt munkadarabszám ($AC_REQUIRED_PARTS) eléréskor a számláló automatikusan "0"-ra lesz állítva ($AC_REQUIRED_PARTS > 0 feltétellel).
$AC_SPECIAL_PARTS	felhasználó által számolt munkadarabok száma Ez a számláló lehetővé tesz a felhasználónak egy munkadarab-számlálást saját definíció szerint. Definiálni lehet egy vészjelzés kiadást a kívánt munkadarabszám ($AC_REQUIRED_PARTS) elérésekor. A számláló nullázásáról a felhasználónak kell gondoskodni.

Megjegyzés

Minden számláló a vezérlés felfutásakor "0"-ra lesz állítva és aktiválásuktól függetlenül olvashatók és írhatók.

Megjegyzés

Csatorna-specifikus gépadatokkal lehet befolyásolni a számlálók aktiválását, a nullázás időpontját és a számláló algoritmust.

Megjegyzés

Munkadarab számlálás a felhasználó által definiált M-utasításokkal

Gépadatokkal be lehet állítani, hogy a különféle munkadarab számlálók számláló impulzusait az `M2`/`M30` programvég helyett a felhasználó által definiált M-utasítások váltsák ki.

18.9 Process DataShare - Kiadás egy külső készülékre/fájlba (EXTOPEN, WRITE, EXTCLOSE)

Az adatok írása egy munkadarabprogramból egy külső készülékre/fájlba három lépésben történik:

1. Külső készüléket/fájlt megnyitni
   Az EXTOPEN utasítással a külső készülék/fájl a csatornának írásra meg lesz nyitva.
2. Adatok írása
   A kiadási dátumot az NC nyelv string-funkcióival, pl. SPRINT, lehet feldolgozni. Az írás a WRITE utasítással történik.
3. Külső készüléket/fájlt bezárni
   Az EXTCLOSE utasítással vagy a programvég elérésével (M30) ill. csatorna Reset-tel a csatornában lefoglalt külső készülék/fájl ismét engedélyezve lesz.

Szintaxis

```
DEF INT <Result>
DEF STRING[<n>] <Output>
…
EXTOPEN(<Result>,<ExtDev>,<SyncMode>,<AccessMode>,<WriteMode>)
…
<Output>="Adatok kiadása"
WRITE(<Result>,<ExtDev>,<Output>)
…
EXTCLOSE(<Result>,<ExtDev>)
```

Jelentés

EXTOPEN:	előre definiált előre definiált eljárás egy külső készülék/fájl megnyitásához		
<Result>:	**Paraméter 1:** eredmény-változó Az eredmény-változó értéke alapján a programban a művelet sikerességét ki lehet értékelni és annak megfelelően folytatni.		
	típus:	INT	
	Értékek:	0	nincs hiba
		1	külső készüléket nem lehet megnyitni
		2	külső készülék nem állítható be
		3	külső készülék érvénytelen ággal beállítva
		4	nincs hozzáférési jog külső készülékhez
		5	felhasználási módus: külső készülék már "kizárólagosan" foglalt
		6	felhasználási módus: külső készülék már "shared" foglalt
		7	Fájl-hossz nagyobb mint LOCAL_DRIVE_MAX_FILESIZE
		8	külső készülékek maximális száma túllépve
		9	LOCAL_DRIVE opció nincs beállítva
		11	V.24 funkciót már az Easy-Message funkció foglalja (csak 828D)
		12	írási módus megadás ellentmond extdev.ini-nek
		16	érvénytelen külső ág van programozva
		22	külső készülék nincs mount-olva

<ExtDev>:	**Paraméter 2:** szimbolikus jelölő a megnyitandó külső készülékre/fájlra		
	típus:	STRING	
	A szimbolikus jelölő a következőkből áll: 1. logikai készüléknév 2. esetleg követi egy ág-megadás ("/"-vel hozzáfűzve) A következő **logikai készüléknevek** vannak definiálva:		
	"LOCAL_DRIVE":		helyi CompactFlash Card (előre definiált)
	"CYC_DRIVE":		foglalt meghajtók a SIEMENS ciklusok alkalmazásához (előre definiált)
	"/dev/ext/1", ... "/dev/ext/9":		Rendelkezésre álló meghajtók **Utalás:** Beállítás a extdev.ini fájlban szükséges!
	"/dev/cyc/1", "/dev/cyc/2":		foglalt meghajtók a SIEMENS ciklusok alkalmazásához **Utalás:** Beállítás a extdev.ini fájlban szükséges!
	"/dev/v24":		V.24 interfész **Utalás:** Beállítás a extdev.ini fájlban szükséges!
	Fájl-ág: • A "LOCAL_DRIVE" és "CYC_DRIVE" esetén meg kell adni egy fájl-ágat, pl.: "LOCAL_DRIVE/my_dir/my_file.txt" • A "/dev/ext/1...9" és "/dev/cyc/1...2" logikai készülék nevek a beállítással: – már egy fájlra utalhatnak, akkor csak a logikai készüléknevet szabad megadni, pl.: "/dev/ext/4" – vagy egy könyvtárra, akkor meg kell adni egy fájl-ágat, pl.: "/dev/ext/5/my_dir/my_file.txt" • A "/dev/v24"-hoz nem szabad egy fájl-ágat hozzátenni.		
	Utalás: A "/dev/ext/1...9", "/dev/v24" és "/dev/cyc/1...2" készülékneveknél nem számít a nagy-/kisbetű, egy fájl fájl-ág megadásánál számít a nagy-/kisbetű. A "LOCAL_DRIVE" és "CYC_DRIVE" esetén csak a nagybetű megengedett.		
<SyncMode>:	**Paraméter 3:** megmunkálás módus a WRITE utasításhoz erre a készülékre/fájlra		
	típus:	STRING	
	Értékek:	"SYN":	Szinkron írás A program végrehajtása meg lesz állítva, amíg az írási eljárás lezárul. A szinkron írás sikeres befejezését a WRITE utasítás hibaváltozójának kiértékelésével lehet megvizsgálni.
		"ASYN":	Aszinkron írás A program végrehajtását a WRITE utasítás nem szakítja meg. **Utalás:** A WRITE utasítás eredmény-változója ebben a módusban nem érdekes és az értéke mindig 0 (nincs hiba). Ebben a módusban nincs bizonyosság, hogy a WRITE utasítás sikeres volt-e.

<AccessMode>:	**Paraméter 4:** felhasználási módus erre a készülékre/fájra		
	típus:	STRING	
	Értékek:	"SHARED":	A készülék/fájl "Shared" módusban lesz igényelve. Más csatornák is használhatják a készüléket, vagyis szintén ebben a módusban megnyithatják.
		"EXCL":	A készülék/fájl ebben a csatornában kizárólagosan van használva, más csatorna nem használhatja a készüléket.
<WriteMode>:	**Paraméter 5:** írási módus a WRITE utasításhoz erre a készülékre/fájlra (opcionális)		
	típus:	STRING	
	Értékek:	"APP":	Hozzáillesztés A fájl tartalma megmarad, az írás a végéhez hozzáillesztve történik.
		"OVR":	Átírás A fájl tartalma törülve lesz és a következő írásokkal újra lesz létrehozva.
	Utalás: Ezzel a paraméterrel nem lehet az extdev.ini fájlban megadott írás módust átírni. Konfliktus esetben az EXTOPEN hívás hibával lesz nyugtázva.		

WRITE:	előre definiált eljárás a kiadási adatok írására

EXTCLOSE:	előre definiált eljárás egy megnyitott külső készülék/fájl lezárására		
<Result>:	**Paraméter 1:** eredmény-változó		
	típus:	INT	
	értékek:	0	nincs hiba
		16	érvénytelen külső ág van programozva
		21	hiba a külső készülék lazárásánál
<ExtDev>:	**paraméter 2:** szimbolikus jelölő a lezárandó külső készülékre/fájlra, leírást lásd az EXTOPEN-nél is! **Utalás:** A jelölő azonos kell legyen az EXTOPEN felhívásban megadott jelölővel!		

Példa

Programkód

```
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
```

Programkód

```
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
…
```

Lásd még

String műveletek (Oldal 84)
Fájl írás (WRITE) (Oldal 145)

18.10 Vészjelzések (SETAL)

Egy NC programban be lehet állítani vészjelzéseket. Ezek a kezelőfelületen egy külön mezőben lesznek ábrázolva. Egy vészjelzéshez mindig kötődik a vezérlésnek a vészjelzés kategóriának megfelelő reakciója.

Irodalom:
További információk a vészjelzés reakciókhoz, lásd az Üzembehelyezési kézikönyvben.

Szintaxis

```
SETAL(<vészjelzés-szám>[,<karakterlánc>])
```

Jelentés

`SETAL`:	kulcsszó egy vészjelzés programozásához A `SETAL`-t külön NC-mondatban kell programozni.	
`<vészjelzés-szám>`:	INT típusú érték. Tartalmazza a vészjelzés-számot. A vészjelzés-számok érvényes tartománya 60000 ... 69999, amiből 60000 ... 64999 a SIEMENS ciklusok számára foglalt és 65000 ... 69999 a felhasználó rendelkezésére áll.	
`<karakterlánc>`:	A felhasználó ciklus vészjelzések programozásánál továbbá rendelkezésre áll egy karakterlánc max.. 4 paraméterrel. Ezekben a paraméterekben lehet változó felhasználói szövegeket definiálni. Az alábbi előre definiált paraméterek is rendelkezésre állnak:	
	Paraméter	**Jelentés**
	%1	csatornaszám
	%2	mondatszám, címke
	%3	szöveg-index ciklus vészjelzésekhez
	%4	további vészjelzés paraméter

Megjegyzés

A vészjelzés szövegek a kezelőfelületen be kell legyenek állítva.

Megjegyzés

Ha a vészjelzés kiadása a kezelőfelületen aktív nyelven kell történjen, a felhasználónak információra van szüksége a HMI-n aktuálisan beállított nyelvről. Ezt az információt a munkadarabprogramban és a szinkronakciókban a $AN_LANGUAGE_ON_HMI rendszerváltozóval lehet lekérdezni (lásd "Aktuális nyelv a HMI-ben (Oldal 914)").

Példa

```
Programkód                          Kommentár
...
N100 SETAL(65000)                   ; vészjelzés-szám 65000 beállítása
...
```

18.11 Kibővített leállítás és visszahúzás (ESR)

A "Kibővített leállítás és visszahúzás" funkció, a továbbiakban ESR-rel jelölve, lehetőséget hiba esetén a folyamattól függetlenül a rugalmas reagálásra:

- Kibővített leállítás
  Amennyiben a specifikus hiba helyzet megengedi, a Kibővített leállításra engedélyezett tengelyek rendezetten lesznek leállítva.
- Visszahúzás
  A használatban lévő szerszám a lehető leggyorsabban vissza lesz húzva a munkadarabtól.
- Generátoros üzem (SINAMICS hajtás funkció "Vdc szabályozás")
  A közbensőköri feszültségnek egy paraméterezhető érték alá csökkenésénél, pl. a hálózati feszültség kiesésénél, a visszahúzáshoz szükséges elektromos energia az erre megadott hajtások fékezési energiájának visszatáplálásával (generátoros üzem) lesz biztosítva.

Kiváltó források

Általános források (NC külső/globális vagy BAG-/csatorna-specifikus)

- digitális bemenetek (pl. az NCU modulon) ill. a digitális kimenetek vezérlésen belüli, visszaolvasható leképezése ($A_IN, $A_OUT)
- csatorna állapot ($AC_STAT)
- VDI jelek ($A_DBB)
- vészjelzések egy csoportjának gyűjtő jelzése ($AC_ALARM_STAT)

Tengely források

- követő tengely vész-visszahúzási küszöb (elektronikus csatolás szinkronfutás, $VC_EG_SYNCDIFF[<követő tengely>])
- Hajtás: közbensőkör figyelmeztetési küszöb (alacsony feszültség veszélye), $AA_ESR_STAT[<tengely>]
- Hajtás: generátor minimális fordulatszám küszöb (nincs több visszatáplálható forgási energia), $AA_ESR_STAT[<tengely>].

Statikus szinkronakciók logikai kapcsolása Forrás/reakció kapcsolat

A statikus szinkronakciók rugalmas kapcsolati lehetőségeit lehet használni a források alapján időben viszonylag közel megadott reakciók kiváltására.

Az összes érintett forrás kapcsolata a statikus szinkronakciók segítségével a felhasználó kezében van. A forrás rendszerváltozókat lehet egészében vagy egy bit maszk segítségével szelektíven kiértékelni és ahhoz kapcsolni a kívánt reakciókat. A statikus szinkron-akciók minden üzemmódban hatnak.

irodalom:
Szinkronakciók működési kézikönyv

Aktiválás

Funkció engedélyezés

A generátor üzem, visszahúzás, leállítás a hozzájuk tartozó $AA_ESR_ENABLE vezérlő jelek beállításával lesznek engedélyezve. Ezt a vezérlő jelet meg lehet változtatni szinkronakciókkal.

Funkció kiváltása

Az ESR az összes engedélyezett tengelyre közösen a $AC_ESR_TRIGGER rendszerváltozó beállításával lesz kiváltva.

A generátor üzem a hajtásban a veszélyesen alacsony közbensőköri feszültség felismerésekor "automatikusan" aktív lesz.

A hajtás alapú leállítás és/vagy visszahúzás aktív lesz a kommunikáció (NC és hajtás között) kiesésének felismerésénél és a hajtásban az alacsony közbensőköri feszültség felismerésekor (konfigurálás és engedélyezés feltételével).

A hajtás alapú leállítást és/vagy visszahúzást kiváltható még az NC oldalról is megfelelő $AN_ESR_TRIGGER (Broadcast utasítás minden hajtásnak) vezérlőjel beállításával is.

Irodalom

Részletes információk az ESR-hez lásd:

Működési kézikönyv Különleges funkciók; Bővített leállás és visszahúzás (R3)

18.11.1 NC-vezérelt ESR

18.11.1.1 NC vezetésű visszahúzás (POLF, POLFA, POLFMASK, POLFMLIN)

Az NC vezetésű visszahúzáshoz bizonyos kiinduló feltételek szükségesek (lásd "NC vezetésű visszahúzás (POLF, POLFA, POLFMASK, POLFMLIN) (Oldal 664)"). Ha ezek a feltételek teljesültek, akkor a csatornában konfigurált visszahúzási tengely(ek)re a $AC_ESR_TRIGGER (ill. $AA_ESR_TRIGGER egyes tengelyekre) rendszerváltozó beállításával a gyors leemelés (LIFTFAST) lesz aktiválva.

Szintaxis

```
POLF[<tengely>]=<pozíció>
POLFA(<tengely>,<típus>,<pozíció>)
POLFMASK(<tengely_1>,<tengely_2>,…)
POLFMLIN(<tengely_1>,<tengely_2>,…)
```

A `POLFA`-ra a következő rövid formák megengedettek:

```
POLFA(<tengely>,<típus>) ; rövid forma egyes tengely visszahúzásra
POLFA(tengely,0/1/2) ; gyors deaktiválás vagy aktiválás
POLFA(tengely,0,$AA_POLFA[tengely]) ; egy előrefutás-állj-t okoz
POLFA(tengely,0) ; nem okoz előrefutás-állj-t
```

Jelentés

POLF:	cím a visszahúzás tengely célpozíciójának megadásához POLF modálisan hatásos.		
	<tengely>:	geometriai vagy csatorna-/géptengely neve, amelyik visszahúz	
	<pozíció>:	visszahúzási pozíció	
		típus:	REAL
		A geometriai tengelyekre az MKR érvényes, egyébként a GKR. Azonos jelölőknél geometriai vagy csatorna-/géptengelyre a visszahúzás MKR-ben történik.	
POLFA:	előre definiált alprogramhívás az egyes tengelyek visszahúzási pozíciójának megadására		
	<tengely>:	csatornajelölő	
	<típus>:	pozíció megadási módus	
		típus:	INT
		érték:	0: pozíció értéket érvénytelennek megjelölni
			1: pozíció érték abszolút
			2: pozíció érték növekményes (távolság)
		Utalás: Ha egy tengely nem egyes tengely vagy hiányzik a típus ill. típus=0, akkor egy megfelelő vészjelzés lesz kiadva.	
	<pozíció>:	visszahúzási pozíció (lásd fent)	
		Utalás: A pozíció érték a típus=0 esetén is át lesz véve. Csak ekkor ez az érték érvénytelennek van megjelölve és a visszahúzáshoz újra kell programozni.	
POLFMASK:	Előre definiált alprogramhívás, amelyek a gyors leemelés kiváltása után **egymástól függetlenül** vissza lesznek húzva.		
	<tengely_1>:	A tengelyek nevei, amelyek a gyors leemelésnél a POLF-fal definiált pozícióikra kell menjenek. Az összes megadott tengely azonos koordinátarendszerben kell legyen.	
	POLFMASK() tengely megadása nélkül deaktiválja a gyors leemelést az összes tengelyre, amelyek egymástól függetlenül vissza lettek húzva.		
POLFMLIN:	Előre definiált alprogramhívás a tengelyek kiválasztására, amelyek a gyors leemelés kiváltása után **lineáris kapcsolatban** lesznek visszahúzva.		
	<tengely_1>,...:	lásd fent	
	POLFMLIN() tengely megadása nélkül deaktiválja a gyors leemelést az összes tengelyre, amelyek lineáris kapcsolatban lettek visszahúzva.		

Megjegyzés

Mielőtt POLFMASK vagy POLFMLIN által a gyors visszahúzást egy fix pozícióra engedélyezni lehet, a kiválasztott tengelyekre kell POLF-fal programozni egy pozíciót.

Megjegyzés

ha a tengelyek egymás után POLFMASK, POLFMLIN vagy POLFMLIN, POLFMASK által engedélyezve lettek, a mindenkori tengelyre mindig az utolsó megadás érvényes.

Megjegyzés

A POLF-fal programozott pozíciók és az aktiválás POLFMASK vagy POLFMLIN által a munkadarabprogramban törölve lesz. Ez azt jelenti, hogy a felhasználónak minden munkadarabprogramban a POLF értékét és a POLFMASK ill. POLFMLIN által kiválasztott tengelyeket újra kell programoznia.

Megjegyzés

Ha a POLFA rövid forma használatával csak a típus lesz megváltoztatva, akkor a felhasználónak biztosítania kell, hogy vagy a visszahúzási pozíció vagy a visszahúzási érték értelmes. A visszahúzási pozíciót és a visszahúzási értéket különösen POWER ON után kell újra beállítani.

Példa

Egy egyes tengely visszahúzása:

Programkód	Kommentár
MD37500 $MA_ESR_REACTION[AX1]=21	; NC vezetésű visszahúzás
...	
$AA_ESR_ENABLE[AX1]=1	
POLFA(AX1,1,20.0)	; AX1 a 20.0 (abszolút) tengely visszahúzási pozíciót kapja.
$AA_ESR_ENABLE[AX1]=1	; Innen kezdődik a visszahúzás.

További információk

NC vezetésű visszahúzás előfeltételei

- A csatornában be van állítva egy visszahúzási tengely az NC vezetésű visszahúzáshoz:
  MD37500 $MA_ESR_REACTION = 21
- ESR engedélyezve kell legyen erre a tengelyre:
  $AA_ESR_ENABLE = 1
- A késleltetési idők definiálva vannak:
  MD21380 $MC_ESR_DELAY_TIME1
  MD21381 $MC_ESR_DELAY_TIME2
- A munkadarabprogramban tengely-specifikus visszahúzási pozíciók vannak programozva POLF-fal.
- A tengelyek POLFMASK/POLFMLIN által ki vannak választva az NC vezetésű visszahúzásra.
- A visszahúzási mozgásra az engedélyező jelek be kell legyenek állítva és beállítva kell maradjanak.

NC vezetésű visszahúzást engedélyezni és indítani

Ha a $AC_ESR_TRIGGER = 1 rendszerváltozó be van állítva, és ha ebben a csatornában van konfigurálva egy visszahúzási tengely (azaz MD37500 $MA_ESR_REACTION = 21) és erre a tengelyre $AA_ESR_ENABLE = 1 be van állítva, akkor ebben a csatornában gyors leemelés (LIFTFAST) lesz aktiválva.

A `POLF` (ill. `LFPOS`) által konfigurált leemelési mozgás a `POLFMASK` vagy `POLFMLIN` által kiválasztott tengely(ek)re helyettesíti az ezekre a tengelyekre a munkadarabprogramban megadott pályamozgást.

A visszahúzásra maximum az MD21380 $MC_ESR_DELAY_TIME1 és MD21381 $MC_ESR_DELAY_TIME2 idők összege áll rendelkezésre. Ezen időtartam lefutása után a visszahúzási tengelyre is egy gyors fékezés lesz indítva azt követően utánvezetéssel.

Megjegyzés

A kibővített visszahúzási mozgás (azaz a $AC_ESR_TRIGGER által kiváltott LIFTFAST/LFPOS) **nem megszakítható**és csak Vész ki által fejezhető be idő előtt.

Megjegyzés

A $AC_ESR_TRIGGER által kiváltott visszahúzás többszöri visszahúzás ellen zárolva van.

Egyes tengely visszahúzás

Az egyes tengely visszahúzásnál a `POLFA` által programozott kell legyen az egyes tengely visszahúzási pozíciója és a következő feltételeket be kell tartani:

- $AA_ESR_ENABLE = 1
- <tengely> a trigger időpontban ($AA_ESR_TRIGGER = 1) egy egyes tengely kell legyen.
- <típus> 1 vagy 2 kell legyen.

Visszahúzási irány gyors leemelésnél

A gyors leemelés aktiválásnak időpontjában érvényes frame figyelembe lesz véve.

Megjegyzés

A frame-k forgatással befolyásolják `POLF`-fal az irányt is, amerre a leemelés történik.

Tengely-csere

A visszahúzási tengelyeket mindig pontosan egy NC csatornához kell hozzárendelni és nem cserélhetőek a csatornák között. Egy visszahúzási tengelynek egy másik csatornába cserlése kísérleténél egy vészjelzés lesz kiadva. Miután ez a tengely $AA_ESR_ENABLE[AX] = 0 által ismét deaktiválva lett, a tengely lehet cserélni egy új csatornába. A tengely csere megtörténte után a tengelyeket a $AA_ESR_ENABLE[AX] = 1 által ismét meg lehet fogni.

Semleges tengelyek

Semleges tengelyek nem tudnak NC vezetésű ESR végrehajtani.

18.11.1.2 NC vezetésű leállítás

A csatornában konfigurált leállítási tengely(ek)re a $AC_ESR_TRIGGER (ill. $AA_ESR_TRIGGER egyes tengelyekre) rendszerváltozó beállításával az NC vezetésű leállítás lesz aktiválva.

Előfeltételek

- A csatornában be van állítva egy leállítási tengely az NC vezetésű leállításhoz:
  MD37500 $MA_ESR_REACTION = 22
- ESR engedélyezve kell legyen erre a tengelyre:
  $AA_ESR_ENABLE = 1
- A késleltetési idők definiálva vannak:
  MD21380 $MC_ESR_DELAY_TIME1 (ESR tengelyek késleltetési ideje)
  MD21381 $MC_ESR_DELAY_TIME2 (ESR idő interpolátoros fékezéshez)

Lefutás

Az MD21380 időtartam alatt a tengely zavartalanul tovább interpolál, ahogyan programozva lett. Az MD21380 időtartam lefutása után az interpolálással vezetett fékezés (rámpa stop) fog indulni. Az interpolálással vezetett fékezésre maximum az MD21381 időtartam áll rendelkezésre. Ezen időtartam lefutása után gyors fékezés indul utánvezetéssel követve.

Példa

Egy egyes tengely leállítása:

Programkód	Kommentár
`MD37500 $MC_ESR_REACTION[AX1]=22`	`; NC vezetésű leállítás.`
`MD21380 $MC_ESR_DELAY_TIME1[AX1]=0.3`	
`MD21381 $MC_ESR_DELAY_TIME2[AX1]=0.06`	
`...`	
`$AA_ESR_ENABLE[AX1]=1`	
`$AA_ESR_ENABLE[AX1]=1`	`; Innen kezdődik a leállítás.`

18.11.2 Hajtás-alapú ESR

18.11.2.1 Hajtásalapú leállítás beállítása (ESRS)

Az `ESRS(...)` funkcióval a hajtásparaméterek a hajtásalapú ESR funkció "leállítás"-hoz lesznek beállítva.

Szintaxis

```
ESRS(<tengely_1>,<leállítási idő_1>[,...,<tengely_n>,<leállítási
idő_n>])
```

Jelentés

`ESRS(...)`:	Funkció a hajtásparaméterek írásához az ESR "leállítás" funkcióhoz A funkció: • egyedül kell álljon a mondatban. • előrefutás-álljt vált ki. • nem lehet használni szinkronakciókban.	
`<tengely_1>`, `...`, `<tengely_n>`:	tengely, amihez hajtásalapú leállítást kell beállítani A hajtásban ehhez a tengelyhez a p0888 hajtásparaméter (konfiguráció) lesz írva: p0888 = 1	
	típus:	AXIS
	értéktartomány:	csatornajelölő
`<leállítás idő_1>`, `...`, `<leállítás idő_n>`:	Időtartam, ameddig a hajtás egy hiba fellépése után az aktuális fordulatszám-parancsértékkel állandóan megy tovább A hajtásban ehhez a tengelyhez a p0892 hajtásparaméter (időtag) lesz írva: p0892 = <leállítási idő>	
	egység:	s
	típus:	REAL
	értéktartomány:	0.00 - 20.00
Egy funkció felhívásban maximum 5 tengely programozható; n = 5		

18.11.2.2 Hajtásalapú visszahúzás beállítása (ESRS)

Az `ESRR(...)` funkcióval a hajtásparaméterek a hajtásalapú ESR funkció "visszahúzás"-hoz lesznek beállítva.

Szintaxis

```
ESRR(<tengely_1>,<visszahúzási út_1>,<visszahúzási
sebesség_1>[,...,<tengely_n>,<visszahúzási út_n>,<visszahúzási
sebesség_n>])
```

Jelentés

`ESRR(...)`:	Funkció a hajtásparaméterek írásához az ESR "visszahúzás" funkcióhoz A funkció: • egyedül kell álljon a mondatban. • előrefutás-álljt vált ki. • nem lehet használni szinkronakciókban.	
`<tengely_1>`, `...`, `<tengely_n>`:	tengely, amit a hajtásalapú visszahúzáshoz be kell állítani A hajtásban ehhez a tengelyhez a p0888 hajtásparaméter (konfiguráció) lesz írva: p0888 = 2	
	típus:	AXIS
	értéktartomány:	csatornajelölő
`<visszahúzási út_1>`, `...`, `<visszahúzási út_n>`:	A visszahúzási út a hajtás számára át lesz számítva egy visszahúzási fordulatszámra. A hajtásban ehhez a tengelyhez a p0893 hajtásparaméter (fordulatszám) lesz írva: p0893 = (<visszahúzási út_n> átszámítva visszahúzási fordulatszámra)	
	egység:	mm/perc, hüv/perc, fok/perc (a tengely egységétől függően)
	típus:	REAL
	értéktartomány:	MIN - MAX
`<visszahúzási sebesség_1>`, `...`, `<visszahúzási sebesség_n>`:	A visszahúzási sebesség a hajtás számára át lesz számítva egy időtartamra. A hajtásban ehhez a tengelyhez a p0892 hajtásparaméter (időtag) [s] lesz írva: p0892 = <visszahúzási út_n> / <visszahúzási sebesség_n>	
	egység:	mm/perc, hüv/perc, fok/perc (a tengely egységétől függően)
	típus:	REAL
	értéktartomány:	0.00 - MAX
Egy funkció felhívásban maximum 5 tengely programozható; n = 5		

18.12 Nyersdarab definiálása (WORKPIECE)

A vezérlésnek kell ismernie egy nyersdarab formáját és nagyságát, hogy ki tudja jelezni a grafikus szimulációban. A felhasználónak lehetősége van a nyersdarabokat a kezelőfelületen vagy közvetlenül NC programban definiálni. A nyersdarab definíciók megmaradnak a (programvég-/csatorna-/BAG-)Reset után is. A vezérlés következő felfutásánál automatikus törölve lesznek.

Szintaxis

```
WORKPIECE("<WP>", "<RefP>", "<ZeroOffset>", "<Type>", <Par5>,
<Par6>, ..., <Par12>)
```

Jelentés

`WORKPIECE(...)`:		Előre definiált eljárás egy nyersdarab definiálásához	
		előrefutás-állj:	igen
		egyedül a mondatban:	igen
Paraméterek:			
1	`"<WP>"`:	munkadarab neve (opcionális)	
		adattípus:	STRING
		A megadás csak akkor szükséges, ha egy csatornában több munkadarab lehet. Megadás nélkül a "WORKP<n>" lesz feltételezve, ahol <n> a deklarálandó csatorna száma.	
2	`"<RefP>"`:	felfogás (opcionális; csak marógépeknél)	
		adattípus:	STRING
		értéktartomány:	"asztal" felfogás a fix asztalra
			"A" felfogás az A körtengelyre
			"B" felfogás a B körtengelyre
			"C" felfogás a C körtengelyre
		Előfeltétel: Az asztal ill. a körtengely a megfelelő gépadattal engedélyezve kell legyen a nyersdarab felfogásra (lásd SINUMERIK Operate üzembehelyezési kézikönyv).	
3	`"<ZeroOffset>"`:	beállítható nullaponteltolás a nyersdarab elhelyezéséhez (nem programozható) Egy beállítható nullaponteltolás kiválasztása a nyersdarab elhelyezéséhez csak a kezelőfelületen való nyersdarab beadásnál lesz felajánlva. A közvetlen nyersdarab definíciónál a munkadarab-programban a nyersdarab ezzel szemben mindig az aktuálisan érvényes nullaponteltolásra vonatkozik.	

4	"<Type>":	nyersdarab forma		
		adattípus:	STRING	
		értéktartomány:	"CYLINDER":	henger
			"PIPE":	cső
			"RECTANGLE":	Központos négyszög
			"BOX":	négyszög
			"N_CORNER":	N-sarok
5 ... 12	<Par5> ... <Par12>:	paraméterek a nyersdarab forma leírásához		
		adattípus:	REAL	
		A szükséges paraméterek száma és azok jelentése a mindenkori nyersdarab formájától és a bit paraméterek értékétől függ. Lásd: • "Paraméterek a nyersdarab forma leírásához" táblázat • "Bit paraméterek" táblázat		
WORPIECE():		Egy WORKPIECE felhívás paraméter nélkül az összes nyersdarab definíciót törli.		
WORPIECE(<WP>):		Egy WORKPIECE felhívás munkadarab nevekkel csak ezt a nyersdarab definíciót törli.		

Táblázat 18-1 paraméterek a nyersdarab forma leírásához

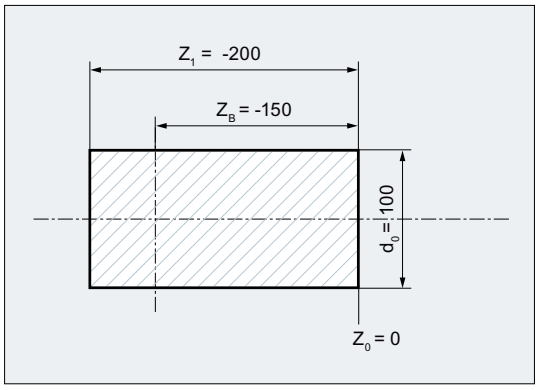
Nyersdarab forma	Paraméter							
	<Par5>	<Par6>	<Par7>	<Par8>	<Par9>	<Par10>	<Par11>	<Par12>
henger	bit paraméter Valós érték, ami bit kódolású egész számként lesz értelmezve. A bitek a következő paraméterek jelentését definiálják (lásd "bit paraméter" táblázat).	vonatkoztatási pont Z_0	Z_1 hossz	megmunkálási méret Z_B	külső átmérő d_0	-	forgatás körtengely körül	-
cső		vonatkoztatási pont Z_0	Z_1 hossz	megmunkálási méret Z_B	külső átmérő d_0	falvastagság (növ) / belső átmérő d_1(absz)	forgatás körtengely körül	-
Központos négyszög		vonatkoztatási pont Z_0	Z_1 hossz	megmunkálási méret Z_B	W szélesség	L hossz	forgatás körtengely körül	-
négyszög		vonatkoztatási pont Z_0	Z_1 hossz	megmunkálási méret Z_B	X_0	Y_0	X_1	Y_1
N-sarok		vonatkoztatási pont Z_0	Z_1 hossz	megmunkálási méret Z_B	sarkok száma	vállszélesség	forgatás körtengely körül	-

Táblázat 18-2 bit paraméter

bit	Jelentés	
4 (0x0010)	négyszög: X_1	
	= 0	növ
	= 1	absz
5 (0x0020)	négyszög: Y_1	
	= 0	növ
	= 1	absz
6 (0x0040)	Z_1 hossz (végső méret)	
	= 0	növ
	= 1	absz
Bit 7 (0x0080)	megmunkálási méret Z_B	
	= 0	növ
	= 1	absz
bit 8 (0x0100)	cső: falvastagság / belső átmérő	
	= 0	növ
	= 1	absz
9 (0x0200)	N-sarok	
	= 0	vállszélesség
	= 1	élhossz
12 (0x1000)	felfogás esztergagépekhez	
	= 0	főorsó
	= 1	ellenorsó
13 (0x2000)	ellenorsó	
	= 0	tükrözéssel
	= 1	tükrözés nélkül

Példák

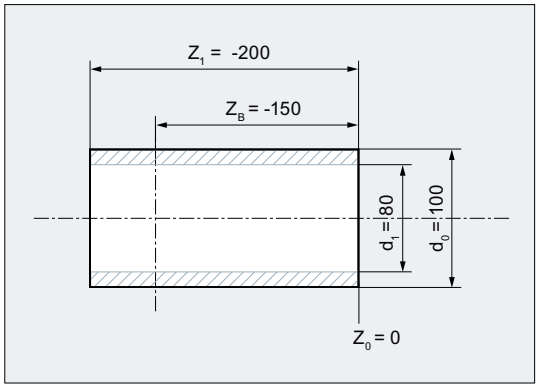
Példa 1: Csőformájú nyersdarab egy esztergagépen



Programkód	Kommentár
...	

Programkód	Kommentár
WORKPIECE(,,,"CYLINDER",0,0,-200,-150,100)	; **nyersdarab definíció:**
	; nyersdarab forma: henger
	; bit paraméter=0(nincs beállít-va bit) → hossz és megmunkálási méret értékek növekményesek, nyersdarab a főorsón
	; vonatkoztatási pont (Z0)=0
	; hossz (Z1)=-200
	; megmunkálási méret (ZB)=-150
	; külső átmérő (d0)=100
...	

Példa 2: Csőformájú nyersdarab egy esztergagépen



Programkód	Kommentár
...	
WORKPIECE(,,,"PIPE",256,0,-200,-150,100,80)	; **nyersdarab definíció:**
	; nyersdarab forma: cső
	; bit paraméter=256(bit8=1) → belső átmérő abszolút, hossz és megmunkálási méret értékek növek-mények, nyersdarab a főorsón
	; vonatkoztatási pont (Z0)=0
	; hossz (Z1)=-200
	; megmunkálási méret (ZB)=-150
	; külső átmérő (d0)=100
	; belső átmérő (d1)=80
...	

18.13 Nyelvi módus átkapcsolása (G290, G291)

A vezérlés lehetőséget munkadarabprogramoknak külső CNC rendszerekről beolvasására és feldolgozására Ennek előfeltétele, hogy a megfelelő nyelv módus (ISO dialektus) az üzembehelyezésnél be lett állítva.

Irodalom:
ISO dialektusok működési kézikönyv

Az ISO dialektus módust minden csatornára külön aktiválni lehet. Például a csatorna 1 futhat ISO dialektusban, miközben a csatorna 2 SINUMERIK módusban aktív.

Az átkapcsolás a SINUMERIK módus és az ISO dialektus módus között az NC programban történik a G-csoport 47 utasításaival. Az aktív szerszámokat, szerszámkorrekciókat és nullaponteltolásokat az átkapcsolás nem befolyásolja.

Szintaxis

```
G291
...
G290
```

Jelentés

G290:	SINUMERIK nyelvi módust aktiválni	
	egyedül a mondatban:	igen
	hatásosság:	modális
G291:	ISO nyelvi módust aktiválni	
	egyedül a mondatban:	igen
	hatásosság:	modális

Feltételek

SINUMERIK módus

- A G utasítások elő-beállítása minden csatornára egy gépadattal beállítható.
- A SINUMERIK módusban nem lehet nyelvi utasításokat az ISO dialektusból programozni.

ISO dialektus módus

- Az ISO dialektus módust a vezérlés alap-beállításaként gépadatokkal lehet beállítani. A vezérlés ezután alap-beállításként ISO dialektus módusban fut fel.
- Csak az ISO dialektus G utasításait lehet programozni. A SINUMERIK funkciók programozása az ISO dialektus módusban nem lehetséges.
- Az ISO dialektus és a SINUMERIK nyelv keverése azonos NC mondatban nem lehetséges.
- Az átkapcsolás G utasításokkal az ISO dialektus M (marás) és az ISO dialektus T (esztergálás) között nem lehetséges.

- Alprogramokat, amelyek SINUMERIK módusban vannak programozva, fel lehet hívni.
- Ha SINUMERIK funkciókat akarunk használni, először át kell kapcsolni a SINUMERIK módusba (lásd a példát).

Példa

Lineáris mondatok komprimálása ISO dialektus módusban

Programkód	**Kommentár**
`N5 G290`	`; SINUMERIK nyelvi módust aktiválni.`
`N10 COMPON`	`; A COMPON a Siemens nyelv egyik utasítása, amelyik egy kompresszor funkciót aktivál, amelyik az egymás utáni lineáris mondatokat meglehetősen nagy pálya-hosszúságú polinom mondatokkal helyettesíti.`
`N15 G291`	`; ISO nyelvi módust aktiválni.`
`N20 G01 X100 Y100 F1000`	`; Mivel a COMPON a SINUMERIK módusban aktiválva lett, ezzel a funkcióval a lineáris mondatokat az ISO dialektus módban is lehet komprimálni.`
`...`	

Saját leforgácsoló-program 19

19.1 A leforgácsolást támogató funkciók

A leforgácsoláshoz kész megmunkáló ciklusokat ajánlunk. Ezen túlmenően megvan a lehetőség a következőkben bemutatott funkciókkal saját leforgácsoló-programokat készíteni.

- Kontúr-táblázatot létrehozni (CONTPRON)
- Kódolt kontúr-táblázatot létrehozni (CONTPRON)
- Kontúr-feldolgozást kikapcsolni (EXECUTE)
- Két kontúrelem metszéspontját megállapítani (INTERSEC)
  (Csak CONTPRON-nal létrehozott táblázatokra.)
- Egy táblázat kontúrelemeinek mondatonkénti feldolgozása (EXECTAB)
  (Csak CONTPRON-nal létrehozott táblázatokra.)
- Kör-adatok kiszámítása (CALCDAT)

Megjegyzés

Ezek a funkciók nemcsak leforgácsoláshoz, hanem univerzálisan használhatók.

Előfeltételek

A CONTPRON vagy CONTDCON felhívása előtt:

- egy kezdőpontra kell menni, ami egy ütközésmentes megmunkálást tesz lehetővé,
- a vágóél-sugárkorrekciót `G40`-nel ki kell kapcsolni.

19.2 Kontúr-táblázatot létrehozni (CONTPRON)

A CONTPRON-nal kapcsoljuk be a kontúr-feldolgozást. Az ezután felhívott NC mondatok nem megmunkálva, hanem egyes mozgásokra felosztva és a kontúr-táblázatban elhelyezve lesznek. Minden kontúrelemnek megfelel egy táblázatsor a kontúr-táblázat kétdimenziós mezőjében. A megállapított hátravágások száma visszaadásra kerül.

Szintaxis

Kontúr-feldolgozást bekapcsolni:
CONTPRON(<kontúr-táblázat>,<megmunkálási mód>,<hátravágások>, <megmunkálási irány>)

Kontúr-feldolgozást kikapcsolni és a normális feldolgozási modusba visszatérni:
EXECUTE(<HIBA>)

Lásd "Kontúr-feldolgozást kikapcsolni (EXECUTE) (Oldal 693)"

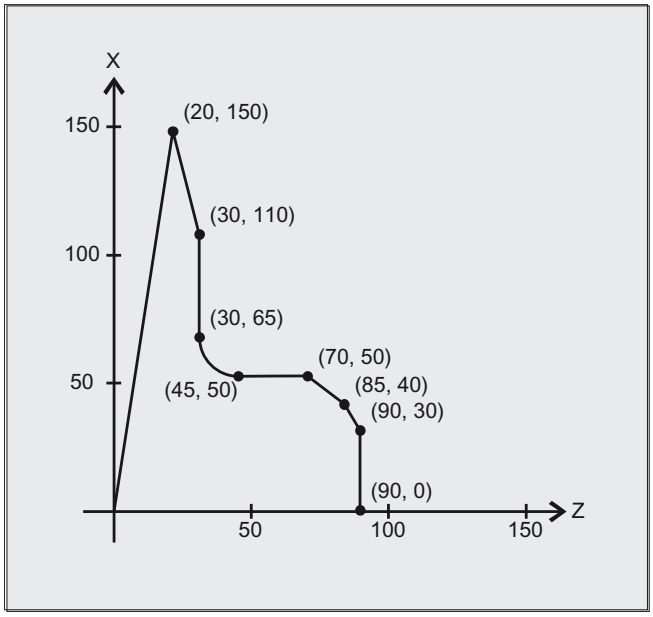
Jelentés

CONTPRON:	Előre definiált eljárás a kontúr-feldolgozás bekapcsolására egy kontúr-táblázat létrehozásához		
<kontúr-táblázat>:	kontúr-táblázat neve		
<megmunkálási mód>:	megmunkálási mód paraméter		
	típus:	CHAR	
	érték:	"G":	hossz-esztergálás: belső megmunkálás
		"L":	hossz-esztergálás: külső megmunkálás
		"N":	sík-esztergálás: belső megmunkálás
		"P":	sík-esztergálás: külső megmunkálás
<hátravágások>:	bemeneti változó a fellépő hátravágás-elemek számához		
	típus:	INT	
<megmunkálási irány>:	megmunkálási irány paraméter		
	típus:	INT	
	érték:	0	kontúr-feldolgozás előre (alap-érték)
		1	kontúr-feldolgozás mindkét irányban

Példa 1

Egy kontúr-táblázat létrehozása a következőkel:

- "KTAB" név
- max. 30 kontúrelem (körök, egyenesek)
- egy változó a fellépő hátravágás-elemek számához
- egy változó a hibajelzéseknek



NC program:

Programkód	Kommentár
N10 DEF REAL KTAB[30,11]	; Kontúr-táblázat KTAB névvel és max. 30 kontúrelemmel, paraméterérték 11 (táblázat oszlopszáma) egy fix érték.
N20 DEF INT ANZHINT	; Változó a hátravágás-elemek számához ANZHINT néven.
N30 DEF INT FEHLER	; Változó a hiba visszajelzésre (0=nincs hiba, 1=hiba).
N40 G18	
N50 CONTPRON(KTAB,"G",ANZHINT)	; Kontúr-feldolgozást bekapcsolni.
N60 G1 X150 Z20	; N60 ... N120: kontúrleírás
N70 X110 Z30	
N80 X50 RND=15	
N90 Z70	
N100 X40 Z85	
N110 X30 Z90	
N120 X0	
N130 EXECUTE(FEHLER)	; Kontúr-táblázat töltésének befejezése, átkapcsolás normál programüzemre.
N140 …	; A táblázat további feldolgozása.

KTAB kontúr-táblázat:

Index sor	oszlop									
(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

Oszloptartalmak magyarázata:

(0) mutató a következő kontúrelemre (magára a sor számára)

(1) mutató az előző kontúrelemre

(2) kontúr-módus kódolása a mozgásra

lehetséges értékek 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) kontúrelemek kezdőpontja

(3) = abszcissza, (4) = ordináta az aktuális síkban

(5), (6) kontúrelemek végpontja

(5) = abszcissza, (6) = ordináta az aktuális síkban

(7) max-/min-kijelző: jelöli a lokális maximumokat és minimumokat a kontúron

(8) maximális érték a kontúrelem és az abszcissza (hossz-megmunkálásnál) ill. ordináta (sík-megmunkálásnál) között A szög függ a programozott megmunkálási módtól.

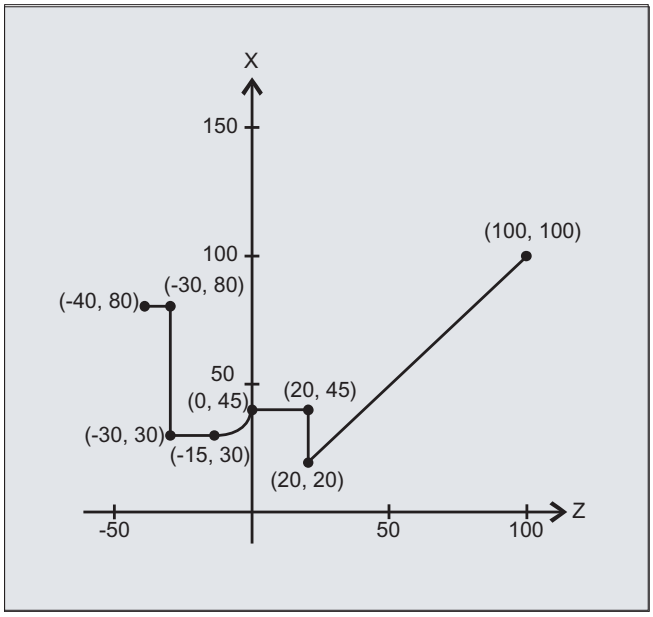
(9), (10) kontúrelem középpont koordinátái, ha az egy körmondat

(9) = abszcissza, (10) = ordináta

Példa 2

Egy kontúr-táblázat létrehozása a következőkel

- KTAB név
- max. 92 kontúrelem (körök, egyenesek)
- üzemmód: hossz-esztergálás, külső megmunkálás
- feldolgozás előre és vissza



NC program:

Programkód	Kommentár
N10 DEF REAL KTAB[92,11]	; Kontúr-táblázat KTAB névvel és max. 92 kontúrelemmel, paraméterérték 11 egy fix érték.
N20 DEF CHAR BT="L"	; CONTPRON üzemmód: hossz-esztergálás, külső megmunkálás
N30 DEF INT HE=0	; hátravágás-elemek száma=0
N40 DEF INT MODE=1	; feldolgozás előre és vissza
N50 DEF INT ERR=0	; hiba visszajelzés
...	
N100 G18 X100 Z100 F1000	
N105 CONTPRON(KTAB,BT,HE,MODE)	; Kontúr-feldolgozást bekapcsolni.
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)	; Kontúr-táblázat töltésének befejezése, átkapcsolás normál programüzemre.
...	

KTAB kontúr-táblázat:

A kontúr-feldolgozás vége után a kontúr mindkét irányban rendelkezésre áll.

Index	oszlop										
sor	(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

Oszloptartalmak magyarázata és észrevételek a 0, 1, 6, 8, 83, 85 és 91 sorokhoz

Az oszloptartalmakhoz a példa 1-ben megadott magyarázatok érvényesek.

Mindig a táblázat sor 0-ban:

1) előző: sor n tartalmazza a kontúr végét előre

2) következő: sor n tartalmazza a kontúr-táblázat végét előre

Mindig egyszer a kontúrelemen belül előre:

3) előző: kontúr kezdete (előre)

4) következő: kontúr vége (előre)

Mindig a kontúr-táblázat vége sorra (előre) +1:

5) előző: hátravágások száma előre

6) következő: hátravágások száma hátra

Mindig egyszer a kontúrelemen belül hátra:

7) következő: kontúr vége (hátra)

8) előző: kontúr kezdet (hátra)

Mindig az utolsó táblázat sorban:

9) előző: sor n a kontúr-táblázat kezdete (hátra)

10) következő: sor n tartalmazza a kontúr kezdetét (hátra)

További információk

Megengedett mozgás-utasítások, koordináta-rendszer

A kontúrprogramozáshoz a következő G-utasítások megengedettek:

- G-csoport 1: `G0`, `G1`, `G2`, `G3`

továbbá lehetségesek:

- lekerekítés és letörés
- kör-programozás `CIP` és `CT`-vel

A Spline-, polinom-, menetfunkciók hibát okoznak.

A koordináta-rendszer változásai egy frame bekapcsolás által a `CONTPRON` és `EXECUTE` között nem megengedett. Ugyanez érvényes a `G70` és `G71` ill. `G700` és `G710` közötti váltásra.

A geometria-tengelyek cseréje `GEOAX`-szal a kontúr-táblázat feldolgozása közben vészjelzést okoz.

Hátravágás-elemek

Az egyes hátravágás-elemek kontúrleírása választhatóan egy alprogramban vagy egyes mondatokban történhet.

Leforgácsolás függetlenül a programozott kontúriránytól

A `CONTPRON` kontúr-feldolgozás úgy lett bővítve, hogy a felhívása után a kontúr-táblázat a programozott iránytól függetlenül rendelkezésre áll.

19.3 Kódolt kontúr-táblázatot létrehozni (CONTPRON)

A CONTDCON-nal bekapcsolt kontúr-feldolgozásnál az utána felhívott NC mondatok egy 6-oszlopos kontúr-táblázat kerülnek elhelyezésre tároló-takarékos módon. Minden kontúrelemnek megfelel egy táblázatsor a kontúr-táblázatban. Az alábbiakban megadásra kerülő kódolási szabályok ismeretében lehet pl. a ciklusokra a táblázati sorokból DIN-kód programokat összeállítani. A 0 számú táblázat-sorban vannak a kezdőpont adatai tárolva.

Szintaxis

Kontúr-feldolgozást bekapcsolni:
CONTDCON(<kontúr-táblázat>,<megmunkálási irány>)

Kontúr-feldolgozást kikapcsolni és a normális feldolgozási modusba visszatérni:
EXECUTE(<HIBA>)

Lásd "Kontúr-feldolgozást kikapcsolni (EXECUTE) (Oldal 693)"

Jelentés

CONTDCON:	Előre definiált eljárás a kontúr-feldolgozás bekapcsolására egy kódolt kontúr-táblázat létrehozásához		
<kontúr-táblázat>:	kontúr-táblázat neve		
<megmunkálási irány>:	megmunkálási irány paraméter		
	típus:	INT	
	érték:	0	kontúr-feldolgozás a kontúrmondatok sora szerint (alapérték)
		1	nem megengedett

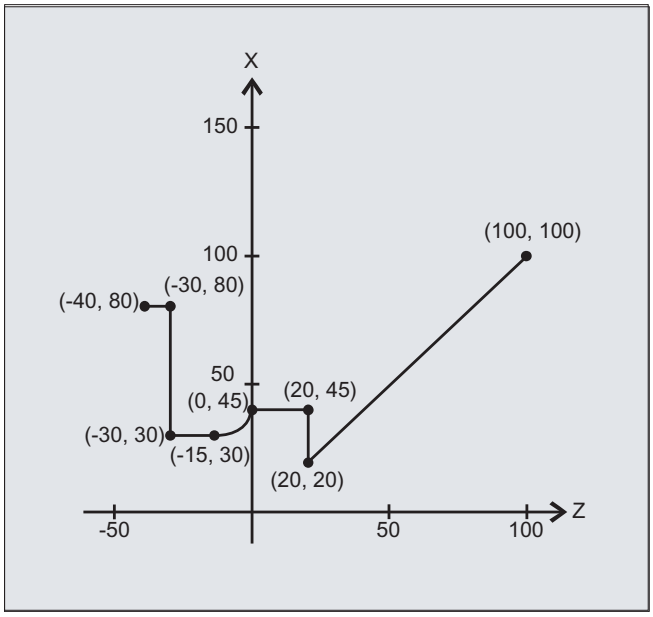
Megjegyzés

A CONTDCON-ra megengedett G utasítások a táblázatos programrészben terjedelmesebbek, mint a CONTPRON-nál. Ezen kívül előtolások és előtolás típusok kontúrdarabonként vannak eltárolva.

Példa

Egy kontúr-táblázat létrehozása a következőkel:

- "KTAB" név
- kontúrelemek (körök, egyenesek)
- üzemmód: Esztergálás
- megmunkálási irány: előre



NC program:

Programkód	Kommentár
N10 DEF REAL KTAB[9,6]	; Kontúr-táblázat KTAB névvel és 9 táblázatsorral. Ez 8 kontúrmondat enged meg. A paraméterérték 6 (táblázat oszlopszám) egy fix érték.
N20 DEF INT MODE = 0	; Változó a megmunkálási irányra. alapérték 0: csak a kontúr programozott irányába
N30 DEF INT ERROR = 0	; Változó a hiba visszajelzésre.
...	
N100 G18 G64 G90 G94 G710	
N101 G1 Z100 X100 F1000	
N105 CONTDCON (KTAB, MODE)	; Kontúr-feldolgozás felhívása (MODE elhagyható).
N110 G1 Z20 X20 F200	; Kontúrleírás.
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)	; Kontúr-táblázat töltésének befejezése, átkapcsolás normál programüzemre.
...	

KTAB kontúr-táblázat:

	oszlop-index					
	0	1	2	3	4	5
sor-index	kontúr-mo-dus	végpont abszcissza	végpont ordináta	középpont abszcissza	középpont ordináta	előtolás
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

Oszloptartalmak magyarázata:

sor 0: kódolások **kezdőpontra:**

oszlop 0: 10^0 (egyes-hely): G0 = 0
10^1 (tizes-hely): G70 = 0, G71 = 1, G700 = 2, G710 = 3
oszlop 1: kezdőpont abszcissza
oszlop 2: kezdőpont ordináta
oszlop 3-4: 0
oszlop 5: az utolsó kontúrdarab sor-indexe a táblázatban

1-n sor: bevitelek a **kontúr-darabnak**

oszlop 0: 10^0 (egyes-hely): G0 = 0, G1 = 1, G2 = 2, G3 = 3
10^1 (tizes-hely): G70 = 0, G71 = 1, G700 = 2, G710 = 3
10^2 (százas-hely): G90 = 0, G91 = 1
10^3 (ezres-hely): G93 = 0, G94 = 1, G95 = 2, G96 = 3
10^4 (tízezres-hely): G60 = 0, G44 = 1, G641 = 2, G642 = 3
10^5 (százezres-hely): G9 = 1
oszlop 1: végpont abszcissza
oszlop 2: végpont ordináta
oszlop 3: középpont abszcissza körinterpolációnál
oszlop 4: középpont ordináta körinterpolációnál
oszlop 5: előtolás

További információk

Megengedett mozgás-utasítások, koordináta-rendszer

A kontúrprogramozáshoz a következő G-csoportok és G-utasítások megengedettek:

G-csoport 1:	G0, G1, G2, G3
G-csoport 10:	G60, G64, G641, G642
G-csoport 11:	G9
G-csoport 13:	G70, G71, G700, G710
G-csoport 14:	G90, G91
G-csoport 15:	G93, G94, G95, G96, G961

továbbá lehetségesek:

- lekerekítés és letörés
- kör-programozás CIP és CT-vel

A Spline-, polinom-, menetfunkciók hibát okoznak.

A koordináta-rendszer változásai egy frame bekapcsolás által a CONTDCON és EXECUTE között nem megengedett. Ugyanez érvényes a G70 és G71 ill. G700 és G710 közötti váltásra.

A geometria-tengelyek cseréje GEOAX-szal a kontúr-táblázat feldolgozása közben vészjelzést okoz.

Megmunkálási irány

A CONTDCON-nal létrehozott kontúr-táblázat a kontúrnak a programozott irányba történő leforgácsolására alkalmas.

19.4 Két kontúrelem metszéspontját megállapítani (INTERSEC)

Az INTERSEC megállapítja a CONTPRON-nal létrehozott kontúr-táblázat két normált kontúrelemének metszéspontját.

Szintaxis

```
<állapot>=INTERSEC(<kontúr-táblázat_1>[<kontúrelem_1>],
<kontúr-táblázat_2>[<kontúrelem_2>],<metszéspont>,<megmunkálás mód>)
```

Jelentés

INTERSEC:	Előre definiált funkció két kontúrelem metszéspontjának megállapításához a CONTPRON-nal létrehozott kontúr-táblázatból		
<állapot>:	változó a metszéspont-állapotra		
	típus:	BOOL	
	érték:	TRUE	metszéspont van
		FALSE	metszéspont nincs
<kontúr-táblázat_1>:	első kontúr-táblázat neve		
<kontúrelem_1>:	első kontúr-táblázat kontúrelemének száma		
<kontúr-táblázat_2>:	második kontúr-táblázat neve		
<kontúrelem_2>:	második kontúr-táblázat kontúrelemének száma		
<metszéspont>:	metszéspont-koordináták az aktív síkban (G17 / G18 / G19)		
	típus:	REAL	
<megmunkálási mód>:	megmunkálási mód paraméter		
	típus:	INT	
	érték:	0	metszéspont számítása a paraméter 2-vel aktív síkban (alapérték)
		1	metszéspont számítása függetlenül az átadott síktól

Megjegyzés

Vegyük figyelembe, hogy a változókat az alkalmazásuk előtt kell definiálni.

A kontúrok átadása megköveteli a CONTPRON-nal definiált értékek betartását:

Paraméter	Jelentés
2	kontúr-mód kódolása a mozgásra
3	kontúr-kezdőpont abszcissza
4	kontúr-kezdőpont ordináta
5	kontúr-végpont abszcissza
6	kontúr-végpont ordináta
9	középpont-koordináta abszcisszához (csak kör kontúrnál)
10	középpont-koordináta ordinátához (csak kör kontúrnál)

Példa

A TABNAME1 táblázat kontúrelem 3 és a TABNAME2 táblázat kontúrelem 7 metszéspontját megállapítani. A metszéspont-koordináták az aktív síkban ISCOORD (1. elem = abszcissza, 2. elem = ordináta) lesznek letéve. Ha nincs metszéspont, ugrás KEINSCH-re (nincs metszéspont).

Programkód	Kommentár
DEF REAL TABNAME1[12,11]	; kontúr-táblázat 1
DEF REAL TABNAME2[10,11]	; kontúr-táblázat 2
DEF REAL ISCOORD[2]	; változó a metszéspont-koordinátákra
DEF BOOL ISPOINT	; változó a metszéspont-állapotra
DEF INT MODE	; változó a feldolgozás módra
…	
MODE=1	; számítás az aktív síktól függetlenül
N10 ISPOINT=INTERSEC(TABNAME1[3],TABNAME2[7], ISCOORD,MODE)	; kontúrelem metszéspontjának felhívása
N20 IF ISPOINT==FALSE GOTOF KEINSCH	; ugrás KEINSCH-hez
…	

19.5 Egy táblázat kontúrelemeinek mondatonkénti megtétele (EXECTAB)

Az EXECTAB-bal egy táblázat kontúrelemeit, amelyek pl. CONTPRON-nal lettek létrehozva, mondatonként lehet megtenni.

Szintaxis

```
EXECTAB(<kontúr-táblázat>[<kontúrelem>])
```

Jelentés

EXECTAB:	Előre definiált eljárás egy kontúrelem megtételére
<kontúr-táblázat>:	kontúr-táblázat neve
<kontúrelem>:	kontúrelem száma

Példa

A KTAB táblázat 0 ... 2 kontúrelemeit mondatonként kell megtenni.

Programkód	Kommentár
N10 EXECTAB(KTAB[0])	; KTAB táblázat 0 elemét megtenni
N20 EXECTAB(KTAB[1])	; KTAB táblázat 1 elemét megtenni
N30 EXECTAB(KTAB[2])	; KTAB táblázat 2 elemét megtenni

19.6 Kör-adatok kiszámítása (CALCDAT)

A `CALCDAT`-tal a kör három vagy négy ismert pontjából ki lehet számítani a sugarat és a körközéppont koordinátákat. A megadott pontok különbözők kell legyenek.

4 pontnál, amelyek nem pontosan a körön fekszenek, a körközéppontra és a sugárra egy középérték lesz képezve.

Megjegyzés

Számítási előírás átlagérték képzéséhez

A körív kiszámítása 4-szer lesz végrehajtva:

1. 1, 2, 3 körpontokkal
2. 1, 2, 4 körpontokkal
3. 1, 3, 4 körpontokkal
4. 2, 3, 4 körpontokkal

A kör-középpont koordináták abszcissza értéke és ordináta értéke úgy lesz kiszámítva, hogy a négy körív számítás abszcissza értékei ill. ordináta értékei összegezve és 4-gyel osztva lesznek.

A sugár úgy lesz kiszámítva, hogy a körív számítások négy sugarának összegéből gyök lesz képezve és az eredmény 0,5-tel lesz megszorozva.

Szintaxis

```
<állapot>=CALCDAT(<kör pontok>[<szám>,<mód>],<szám>,<eredmény>)
```

Jelentés

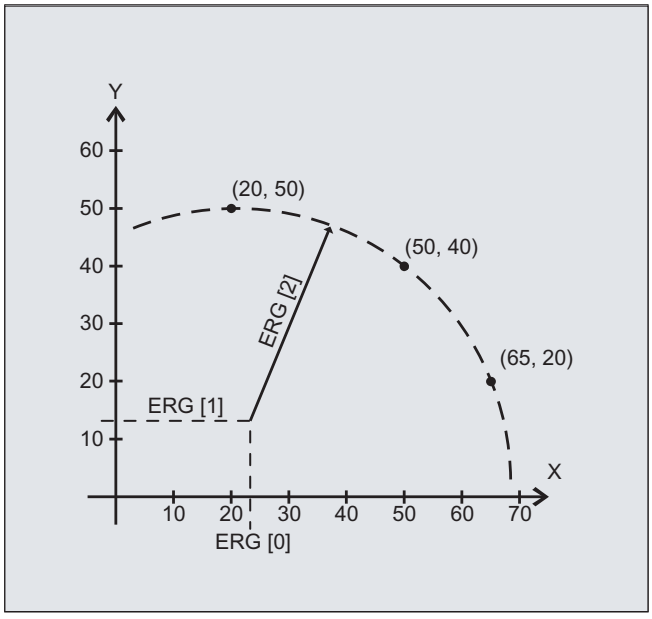
`CALCDAT`:	Előre definiált eljárás egy kör sugarának és középpont koordinátáinak kiszámítására 3 vagy 4 pontból		
`<állapot>`:	változó a kör számítás állapotra		
	típus:	BOOL	
	érték:	TRUE	A megadott pontok egy körön vannak.
		FALSE	A megadott pontok **nincsenek** egy körön.
`<körpontok>[]`:	változó a körpontok megadására paraméterekkel:		
	`<darabszám>`:	Kör pontok darabszáma (3 vagy 4)	
	`<mód>`:	koordináta megadás módja, pl. 2 = 2 pont koordináta megadás	
`<darabszám>`:	paraméter a számításhoz használt pontok számára (3 vagy 4)		
`<eredmény>[3]`:	eredmény változó: körközéppont koordináták és sugár megadása		
	0	körközéppont koordináták: abszcissza érték	
	1	körközéppont koordináták: ordináta érték	
	2	sugár	

Megjegyzés

Vegyük figyelembe, hogy a változókat az alkalmazásuk előtt kell definiálni.

Példa

Három pontról meg kell állapítani, hogy egy körszakaszon fekszenek-e.



Programkód	Kommentár
`N10 DEF REAL PKT[3,2]=(20,50,50,40,65,20)`	`; változó a kör pontok megadására`
`N20 DEF REAL ERG[3]`	`; eredmény változó`
`N30 DEF BOOL STATUS`	`; állapot változó`
`N40 STATUS=CALCDAT(PKT,3,ERG)`	`; kiszámított köradatok felhívása`
`N50 IF STATUS == FALSE GOTOF ERROR`	`; ugrás hibához`

19.7 Kontúr-feldolgozást kikapcsolni (EXECUTE)

Az EXECUTE-tal a kontúr-feldolgozást lekapcsolja és egyidejűleg a normális megmunkálási módot visszakapcsolja.

Szintaxis

```
EXECUTE(<HIBA>)
```

Jelentés

EXECUTE:	Előre definiált eljárás a kontúr-feldolgozás befejezésére	
<hiba>:	változó a hiba visszajelzésre	
	típus:	INT
	A változó értéke mutatja, hogy a kontúrt hibamentesen fel lehetett-e dolgozni:	
	0	hiba
	1	nincs hiba

Példa

Programkód

```
...
N30 CONTPRON(...)
N40 G1 X… Z...
...
N100 EXECUTE(...)
...
```

Ciklusok külső programozása 20

20.1 Technológiai ciklusok

20.1.1 Bevezetés

Tartalom

Ez a fejezet a ciklusok leírását tartalmazza az esztergálás, marás, köszörülés technológiákhoz.

Felépítés

Egy ciklus leírása a következők szerint van felépítve:

- **Szintaxis**
  Ciklus-név és az átadási paraméterek felhívási sorrendje
- **Paraméter**
  Táblázat az egyes paraméterek magyarázatára

Paraméter leírás

A táblázatban egy paraméterhez a következő adatok vannak megadva: Név, leírás, értéktartomány és függőségek más paraméterektől

Az oszlop az utalással a paraméterekre a maszkban a vezérlésben programozott értékek újra megtalálását könnyíti a kívül generált ciklus-hívások visszafordításánál.

Paraméter "csak a felülethez"

A táblázatban paraméterek "csak a felülethez"-zel vannak megjelölve. Ezeknek a ciklus működésében nincs jelentőségük. Ezek csak a ciklus felhívások teljes visszafordításához szükségesek. Ha ezek nincsenek programozva, a ciklus akkor is vissza lehet fordítani, a mezők akkor megfelelően színesen vannak megjelölve és azokat a maszkban kell kitölteni.

Paraméter "foglalt"

A "foglalt"-tal leírt paramétereket 0 értékkel vagy üres hellyel kell programozni, hogy a következő felhívási paraméterek hozzárendelése a ciklus-paraméterekhez illeszkedjen. Kivétel: string paramétereknél "" érték vagy üres vessző

Ciklusok ismétlése pozíció-mintára

Fúró- és maró-ciklusokat lehet ismételni pozíciómintára (modális felhívások). Ekkor a ciklus elé ugyanabban a sorban `MCALL`-t kell írni, pl. `MCALL CYCLE83(...)`.

Megjegyzés

Ha adott paraméterek (pl. `<_VARI>, <_GMODE>, <_DMODE>, <_AMODE>`) közvetetten paraméterként vannak programozva, a beadási maszk a visszafordításnál megnyílik, de nem lehet eltárolni, mert bizonyos kiválasztási mezőkhöz nincs egyértelmű hozzárendelés.

20.1.2 Technológia-specifikus áttekintés

A következő áttekintő táblázatban az összes rendelkezésre álló külső programozható technológiai ciklus fel van tüntetve és a mindenkori technológiához hozzá van rendelve:

Technológia	Technológiai ciklus
Fúrás	• CYCLE81 - Fúrás, központozás (Oldal 735) • CYCLE82 - Fúrás, sík-süllyesztés (Oldal 736) • CYCLE85 - dörzsölés (Oldal 745) • CYCLE86 - Kiesztergálás (Oldal 746) • CYCLE83 - Mély-lyuk fúrás (Oldal 739) • CYCLE830 - Mélylyuk fúrása 2 (Oldal 769) • CYCLE84 - Menetfúrás kiegyenlítő tokmány nélkül (Oldal 742) • CYCLE840 - Menetfúrás kiegyenlítő tokmánnyal (Oldal 779) • CYCLE78 - Furatmenet marás (Oldal 731) • CYCLE802 - Tetszőleges pozíciók (Oldal 767) • Lyuk-sor - HOLES1 (Oldal 698) • CYCLE801 - Rács vagy keret (Oldal 765) • Lyuk-kör - HOLES2 (Oldal 698)
Esztergálás	• CYCLE951 - Leforgácsolás (Oldal 789) • CYCLE930 - Beszúrás (Oldal 784) • CYCLE940 - Szabadra szúrás formák (Oldal 787) • CYCLE99 - Menetesztergálás (Oldal 755) • CYCLE98 - Menetlánc (Oldal 751) • CYCLE92 - Leszúrás (Oldal 747)
Kontúresztergálás	• CYCLE62 - Kontúrhívás (Oldal 717) • CYCLE952 - Kontúr-szúrás (Oldal 792)

Technológia	Technológiai ciklus
Marás	• CYCLE61 - Síkmarás (Oldal 715) • POCKET3 - Négyszögzseb marása (Oldal 700) • POCKET4 - Körzseb marása (Oldal 703) • CYCLE76 - négyszögcsap marás (Oldal 726) • CYCLE77 - Körcsap marása (Oldal 729) • CYCLE79 - sokszög (Oldal 733) • SLOT1 - hosszvájat (Oldal 705) • SLOT2 - körvájat (Oldal 708) • CYCLE899 - Nyitott vájat marása (Oldal 782) • LONGHOLE - Hosszlyuk (Oldal 710) • CYCLE70 - Menetmarás (Oldal 721) • CYCLE60 - Gravírozás ciklus (Oldal 712)
Kontúrmarás	• CYCLE62 - Kontúrhívás (Oldal 717) • CYCLE72 - Pályamarás (Oldal 723) • CYCLE63 - Kontúrzsebet marni (Oldal 718) • CYCLE64 - Kontúrzsebet előfúrni (Oldal 720)
Köszörülés	• CYCLE495 - Profilozás (Oldal 760) • CYCLE435 - Lehúzó koordináták beállítása (Oldal 760) • CYCLE4071 - hossz-köszörülés fogással a fordulóponton (Oldal 798) • CYCLE4072 - Hossz-köszörülés fogással a fordulóponton és megszakítás jel (Oldal 799) • CYCLE4073 - Hossz-köszörülés folyamatos fogásvétellel (Oldal 803) • CYCLE4074 - Hossz-köszörülés folyamatos fogásvétellel és megszakítás jellel (Oldal 804) • CYCLE4075 - Sík-köszörülés fogással a fordulóponton (Oldal 807) • CYCLE4077 - Sík-köszörülés fogással a fordulóponton és megszakítás jel (Oldal 810) • CYCLE4078 - Sík-köszörülés folyamatos fogásvétellel (Oldal 813) • CYCLE4079 - Sík-köszörülés közbenső fogásvétellel (Oldal 815)
Egyebek	• CYCLE800 - Billentés (Oldal 762) • CYCLE832 - High Speed Settings (Oldal 775)
Összes	• GROUP_BEGIN - programblokk kezdete (Oldal 817) • GROUP_END - programblokk vége (Oldal 818) • GROUP_ADDEND - bejáratos kiegészítés vége (Oldal 818)

20.1.3 Lyuk-sor - HOLES1

Szintaxis

```
HOLES1(<SPCA>, <SPCO>, <STA1>, <FDIS>, <DBH>, <NUM>, <_VARI>,
<_UMODE>, <_HIDE>, <_NSP>, <_DMODE>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
1	X0	<SPCA>	REAL	lyuk-sor vonatkoztatási pont az 1. tengelyen (absz)		
2	Y0	<SPCO>	REAL	lyuk-sor vonatkoztatási pont az 2. tengelyen (absz)		
3	α0	<STA1>	REAL	alap forgatási szög (szög az 1. tengelyhez)		
4	L0	<FDIS>	REAL	1. furat távolsága a vonatkoztatási ponttól		
5	L	<DBH>	REAL	két furat távolsága		
6	N	<NUM>	INT	furatok száma		
7		<_VARI>	INT	foglalt		
8		<_UMODE>	INT	foglalt		
9		<_HIDE>	STRING [200]	kihagyott pozíciók • max. 198 karakter • folytatólagos pozíciószámok megadása, pl. "1,3" (1 és 3 pozíciók nem lesznek végrehajtva)		
10		<_NSP>	INT	foglalt		
11		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/18/19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)

20.1.4 Lyuk-kör - HOLES2

Szintaxis

```
HOLES2(<CPA>, <CPO>, <RAD>, <STA1>, <INDA>, <NUM>, <_VARI>,
<_UMODE>, <_HIDE>, <_NSP>, <_DMODE>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belső	Adattípus	Jelentés			
1	X0	<CPA>	REAL	lyuk-kör középpontja az 1. tengelyen (absz) vonatkoztatási az 1. tengelyen			(XY esetén) (XA, YB, ZC esetén)
2	Y0	<CPO>	REAL	lyuk-kör középpontja az 2. tengelyen (absz) vonatkoztatási az 2. tengelyen			(XY esetén) (XA, YB, ZC esetén)
3	R	<RAD>	REAL	lyuk-kör sugara			(XY esetén)
4	α0	<STA1>	REAL	kezdőszög vagy 1. körtengely pozíció			(XY esetén) (XA, YB, ZC esetén)
5	α1	<INDA>	REAL	tovább-kapcsolási szög (csak rész-körnél)			(XY, XA, YB, ZC esetén)
					< 0 =	órajárás irányában	
					> 0 =	órajárás irányával szemben	
6	N	<NUM>	INT	pozíciók száma			
7		<_VARI>	INT	Megmunkálási mód			
				EGYES:	foglalt		
				TÍZES:	pozícionálási mód		
					0 =	pozícióra menet - lineáris	
					1 =	pozícióra menet - körpályán	
				SZÁZAS:	foglalt		
				EZRES:	körminta		
					0 =	kompatibilitás módus, ha INDA = 0 akkor teljes kör, INDA <> 0 akkor rész-kör)	
					1 =	teljes kör	
					2 =	rész-kör	
				TÍZEZRES:	pozíció-minta körtengellyel		
					0 =	XY (körtengely nélkül)	(XY esetén)
					1 =	XA (X tengely és körtengely X körül)	(csak XA esetén)
					2 =	YB (Y tengely és körtengely Y körül)	(csak YB esetén)
					3 =	ZC (Z tengely és körtengely C körül)	(csak ZC esetén)
				EGYMILLIÓS + SZÁZEZRES:	Offset (több körtengely azonos tengely körül esetén; ha az index túl nagy, akkor az 1. tengely)		
					00 =	1. A, B vagy C tengely	
					01 =	2. A, B vagy C tengely	
					...		
					10 =	20 A, B vagy C tengely	
8		<_UMODE>	INT	foglalt			

Nr.	Paraméter maszk	Paraméter belső	Adattípus	Jelentés		
9		<_HIDE>	STRING [200]	foglalt		
10		<_NSP>	INT	foglalt		
11		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/18/19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)

20.1.5 POCKET3 - Négyszögzseb marása

Szintaxis

```
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>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
1	RP	<_RTP>	REAL	visszahúzási sík (absz.)		
2	Z0	<_RFP>	REAL	szerszám-tengely vonatkoztatási pont (absz.)		
3	SC	<_SDIS>	REAL	biztonsági távolság (vonatkoztatási ponthoz adódik, előjel nélkül beadni)		
4	Z1	<_DP>	REAL	zsebmélység (absz/növ), lásd <_AMODE>)		
5	L	<_LENG>	REAL	zsebhossz (növ, előjellel beadni)		
6	W	<_WID>	REAL	zsebszélesség (növ, előjellel beadni)		
7	R	<_CRAD>	REAL	zseb saroksugara		
8	X0	<_PA>	REAL	vonatkoztatási pont 1. tengelyen (absz)		
9	YO	<_PO>	REAL	vonatkoztatási pont 2. tengelyen (absz)		
10	α0	<_STA>	REAL	forgatás-szög, szög a hossztengely (L) és az 1. tengely között		
11	DZ	<_MID>	REAL	maximális fogásmélység		
12	UXY	<_FAL>	REAL	sík simítási ráhagyás		
13	UZ	<_FALD>	REAL	simítási ráhagyás mélység		
14	F	<_FFP1>	REAL	előtolás a síkban		
15	FZ	<_FFD>	REAL	mélységi fogásvétel előtolás		
16		<_CDIR>	INT	marásirány:	0 =	egyirányú
					1 =	ellenirányú

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
17		<_VARI>	INT	megmunkálási mód		
				EGYES:		
					1 =	nagyolás
					2 =	simítás
					4 =	simítás a peremen
					5 =	letörés
				TÍZES:		
					0 =	előfúrva, ráállás G0-lal
					1 =	függőleges, ráállás G1-gyel
					2 =	helikális
					3 =	zseb hossztengelyen ingázva
				SZÁZAS:	foglalt	
18	DXY	<_MIDA>	REAL	maximális sík fogásvétel, egységet lásd <_AMODE>		
19	L1	<_AP1>	REAL	elő-megmunkálás hossza (növ)		
20	W1	<_AP2>	REAL	elő-megmunkálás szélessége (növ)		
21	AZ	<_AD>	REAL	elő-megmunkálás mélysége (növ)		
22	ER	<_RAD1>	REAL	helix-pálya sugara helikális bemerülésnél		
	EW			maximális bemerülési szög ingázásnál		
23	EP	<_DP1>	REAL	helix-emelkedés helikális bemerülésnél		
24		<_UMODE>	INT	foglalt		
25	FS	<_FS>	REAL	letörés szélesség (növ)		
26	ZFS	<_ZFS>	REAL	bemerülési mélység (szerszámcsúcs) letörésnél (absz/növ), lásd <_AMODE>		

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
27		<_GMODE>	INT	geometria módus (programozott geometria-értékek értékelése)		
				EGYES:	foglalt	
				TÍZES:	foglalt	
				SZÁZAS:	megmunkálás kiválasztása / csak kezdőpont kiszámítás	
					0 =	kompatibilitás módus
					1 =	normál megmunkálás
				EZRES:	méretezés középpel/sarokkal	
					0 =	kompatibilitás módus
					1 =	méretezés középpel
					2 =	sarokpont méretezés, zsebhossz +LENG/+WID
					3 =	sarokpont méretezés, zsebhossz +LENG/+WID
					4 =	sarokpont méretezés, zsebhossz +LENG/-WID
					5 =	sarokpont méretezés, zsebhossz -LENG/-WID
				TÍZEZRES:	teljes megmunkálás / utó-megmunkálás	
					0 =	kompatibilitás módus (<_AP1>, <_AP2> és <_AD> mint eddig kezelni)
					1 =	teljes megmunkálás
					2 =	utó-megmunkálás
28		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/G18/G19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
				TÍZES:	előtolás mód: G-csoport (G94/G95) felületi és mélységi előtoláshoz	
					0 =	kompatibilitás módus
					1 =	G utasítás mint a ciklushívás előtt. G94/G95 felületi és mélységi előtoláshoz azonos
				SZÁZAS:	---	foglalt
				EZRES:	---	foglalt
				TÍZEZRES:	Technológia skálázás ciklus maszkokon belül (Oldal 818)	
					0 =	beadás: teljes
					1 =	beadás: egyszerű

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
29		<_AMODE>	INT	alternatív módus		
				EGYES:	zsebmélység (Z1)	
					0 =	abszolút (kompatibilitás módus)
					1 =	növekményes
				TÍZES:	egység sík fogásvételhez (DXY)	
					0 =	mm
					1 =	szerszámátmérő %-a
				SZÁZAS:	bemerülési mélység letörésnél (ZFS)	
					0 =	abszolút
					1 =	növekményes

20.1.6 POCKET4 - Körzseb marása

Szintaxis

```
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>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
1	RP	<_RTP>	REAL	visszahúzási sík (absz.)		
2	Z0	<_RFP>	REAL	szerszám-tengely vonatkoztatási pont (absz.)		
3	SC	<_SDIS>	REAL	biztonsági távolság (vonatkoztatási ponthoz adódik, előjel nélkül beadni)		
4	Z1	<_DP>	REAL	zsebmélység (absz/növ), lásd <_AMODE>		
5	∅	<_CDIAM>	REAL	zsebátmérő vagy zsebsugár, lásd <_DMODE>		
6	X0	<_PA>	REAL	vonatkoztatási pont 1. tengelyen (absz)		
7	Y0	<_PO>	REAL	vonatkoztatási pont 2. tengelyen (absz)		
8	DZ	<_MID>	REAL	maximális mélységi fogásvétel, lásd <_VARI> = síkonként maximális helix emelkedés, lásd <_VARI> = helikális		
9	UXY	<_FAL>	REAL	sík simítási ráhagyás		
10	UZ	<_FALD>	REAL	simítási ráhagyás mélység		
11	F	<_FFP1>	REAL	előtolás felület megmunkáláshoz		
12	FZ	<_FFD>	REAL	mélységi fogásvétel előtolás		
13		<_CDIR>	INT	marásirány	0 =	egyirányú
					1 =	ellenirányú

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
14		<_VARI>	INT	megmunkálási mód		
				EGYES:	megmunkálás	
					1 =	nagyolás
					2 =	simítás
					4 =	simítás a peremen
					5 =	letörés
				TÍZES:	fogásvételi mód (nagyolás és simítás)	
					0 =	előfúrva, ráállás G0-lal (zseb elő van munkálva)
					1 =	függőleges, ráállás G1-gyel
					2 =	helikális
				SZÁZAS:	foglalt	
				EZRES:		
					0 =	síkonként
					1 =	helikális
15	DXY	<_MIDA>	REAL	maximális sík-fogásvétel, lásd <_AMODE>, 0 = 0,8 · szerszámátmérő		
16	∅	<_AP1>	REAL	elő-megmunkálás átmérő/sugár (növ)		
17	AZ	<_AD>	REAL	elő-megmunkálás mélysége (növ)		
18	ER	<_RAD1>	REAL	helix-pálya sugara helikális bemerülésnél		
19	EP	<_DP1>	REAL	helix-emelkedés helix-pályán bemerülésnél		
20		<_UMODE>	INT	foglalt		
21	FS	<_FS>	REAL	letörés szélesség (növ)		
22	ZFS	<_ZFS>	REAL	bemerülési mélység (szerszámcsúcs) letörésnél (absz/növ), lásd <_AMODE>		
23		<_GMODE>	INT	geometria módus (programozott geometria-értékek értékelése)		
				EGYES:	foglalt	
				TÍZES:	foglalt	
				SZÁZAS:	megmunkálás / kezdőpont kiszámítás	
					0 =	kompatibilitás módus
					1 =	normál megmunkálás
				EZRES:	foglalt	
				TÍZEZRES:	teljes megmunkálás / utó-megmunkálás	
					0 =	kompatibilitás módus (<_AP1> és <_AD> mint eddig kezelni)
					1 =	teljes megmunkálás
					2 =	utó-megmunkálás

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
24		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/18/19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
				TÍZES:	előtolás mód: G-csoport (G94/G95) felületi és mélységi előtoláshoz	
					0 =	kompatibilitás módus
					1 =	G utasítás mint a ciklushívás előtt. G94/G95 felületi és mélységi előtoláshoz azonos
				SZÁZAS:		
					0 =	kompatibilitás módus (<_CDIAM>/<_AP1> sugárként beadni)
					1 =	<_CDIAM>/<_AP1> átmérőként beadni
				EZRES:	---	foglalt
				TÍZEZRES:	Technológia skálázás ciklus maszkokon belül (Oldal 818)	
					0 =	beadás: teljes
					1 =	beadás: egyszerű
25		<_AMODE>	INT	alternatív módus		
				EGYES:	zsebmélység (Z1)	
					0 =	abszolút (kompatibilitás módus)
					1 =	növekményes
				TÍZES:	egység fogásvétel szélességhez (DXY)	
					0 =	mm
					1 =	szerszámátmérő %-a
				SZÁZAS:	bemerülési mélység letörésnél (ZFS)	
					0 =	abszolút
					1 =	növekményes

20.1.7 SLOT1 - hosszvájat

Szintaxis

```
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>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
1	RP	<RTP>	REAL	visszahúzási sík (absz.)		
2	Z0	<RFP>	REAL	szerszám-tengely vonatkoztatási pont (absz.)		
3	SC	<SDIS>	REAL	biztonsági távolság (vonatkoztatási ponthoz adódik, előjel nélkül beadni)		
4	Z1	<_DP>	REAL	vájatmélység (absz.)		
5		<_DPR>	REAL	vájatmélység (növ), Z0-ra vonatkoztatva (előjel nélkül beadni)		
6		<NUM>	INT	vájatok száma =1		
7	L	<LENG>	REAL	vájat hossza		
8	W	<WID>	REAL	vájat szélessége		
9	X0	<_CPA>	REAL	vonatkoztatási pont 1. tengelye a síknak		
10	Y0	<_CPO>	REAL	vonatkoztatási pont 2. tengelye a síknak		
11		<RAD>	REAL	foglalt		
12	α	<STA1>	REAL	forgásszög		
13		<INDA>	REAL	foglalt		
14	FZ	<FFD>	REAL	mélységi fogásvétel előtolás		
15	F	<FFP1>	REAL	előtolás		
16	DZ	<_MID>	REAL	maximális fogásmélység		
17		<CDIR>	INT	marásirány	0 =	egyirányú
					1 =	ellenirányú
18	UXY	<_FAL>	REAL	simítási ráhagyás a síkban és a vájat peremén		
19		<VARI>	INT	megmunkálási mód		
				EGYES:		
					0 =	foglalt
					1 =	nagyolás
					2 =	simítás
					4 =	simítás a peremen (csak a peremet megmunkálni)
					5 =	letörés
				TÍZES:	rámenet	
					0 =	előfúrva, ráállás G0-lal (vájat elő van munkálva)
					1 =	függőleges, ráállás G1-gyel
					2 =	helikális
					3 =	ingázva
				SZÁZAS:	foglalt	
20	DZF	<_MIDF>	REAL	foglalt		
21	FF	<FFP2>	REAL	foglalt		
22	SF	<SSF>	REAL	foglalt		
23	UZ	<_FALD>	REAL	simítási ráhagyás mélység		
24	ER	<_STA2>	REAL	helix-pálya sugara helikális bemerülésnél		
	EW			maximális bemerülési szög ingázásnál		

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
25	EP	<_DP1>	REAL	bemerülési mélység fordulatonként helixnek		
26		<_UMODE>	INT	foglalt		
27	FS	<_FS>	REAL	letörés szélesség (növ) letörésnél		
28	ZFS	<_ZFS>	REAL	bemerülési mélység (szerszámcsúcs) letörésnél (absz/növ), lásd <_AMODE>		
29		<_GMODE>	INT	geometria módus (programozott geometria-értékek értékelése)		
				EGYES:	foglalt	
				TÍZES:	foglalt	
				SZÁZAS:	megmunkálás kiválasztása vagy csak kezdőpont kiszámítás	
					1 =	normál megmunkálás
				EZRES:	vonatkoztatási pont beméretezése, vájathossz	
					0 =	közép
					1 =	balra belül +L
					2 =	jobbra belül -L
					3 =	bal perem +L
					4 =	jobb perem -L
30		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/18/19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
				TÍZES:	foglalt	
				SZÁZAS:	foglalt	
				EZRES:	SW verzió jelölés	
					1 =	SLOT1 funkció-bővítés
				TÍZEZRES:	Technológia skálázás ciklus maszkokon belül (Oldal 818)	
					0 =	beadás: teljes
					1 =	beadás: egyszerű
31		<_AMODE>	INT	alternatív módus		
				EGYES:	végső mélység Z1 (absz/növ)	
					0 =	kompatibilitás
					1 =	Z1 (növ)
					2 =	Z1 (absz)
				TÍZES:	foglalt	
				SZÁZAS:	bemerülési mélység letörésnél ZFS	
					0 =	ZFS (absz)
					1 =	ZFS (növ)

Megjegyzés

A ciklus a korábbi SW verziókhoz képest új funkciókkal van ellátva. Ennek az a következménye, hogy bizonyos paraméterek a beadási maszkban már nincsenek kijelezve (`<NUM>`, `<RAD>`, `<INDA>`). Több vájatot egy pozíció-mintában "MCALL"-lal és a kívánt pozíció-minta felhívását pl. HOLES2-vel kell programozni.

20.1.8 SLOT2 - körvájat

Szintaxis

```
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>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
1	RP	`<RTP>`	REAL	visszahúzási sík (absz.)		
2	Z0	`<RFP>`	REAL	szerszám-tengely vonatkoztatási pont (absz.)		
3	SC	`<SDIS>`	REAL	biztonsági távolság (vonatkoztatási ponthoz adódik, előjel nélkül beadni)		
4	Z1	`<_DP>`	REAL	vájatmélység (absz.)		
5		`<_DPR>`	REAL	vájatmélység (növ), Z0-ra vonatkoztatva (előjel nélkül beadni)		
6	N	`<NUM>`	INT	vájatok száma		
7	α1	`<AFSL>`	REAL	vájat nyílásszöge		
8	W	`<WID>`	REAL	vájat szélessége		
9	X0	`<_CPA>`	REAL	vonatkoztatási pont = a kör középpontja, az 1. tengelye a síknak		
10	Y0	`<_CPO>`	REAL	vonatkoztatási pont = a kör középpontja, a 2. tengelye a síknak		
11	R	`<RAD>`	REAL	kör sugara		
12	α0	`<STA1>`	REAL	kezdőszög		
13	α2	`<INDA>`	REAL	továbbkapcsolási szög		
14	FZ	`<FFD>`	REAL	mélységi fogásvétel előtolás		
15	F	`<FFP1>`	REAL	előtolás		
16	DZ	`<_MID>`	REAL	maximális fogásmélység		
17		`<CDIR>`	INT	marásirány	0 =	egyirányú
					1 =	ellenirányú
18	UXY	`<_FAL>`	REAL	simítási ráhagyás a síkban vagy a vájat peremén		

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
19		<VARI>	INT	megmunkálási mód		
				EGYES:		
					0 =	teljes megmunkálás
					1 =	nagyolás
					2 =	simítás
					3 =	simítás a peremen
					5 =	letörés
				TÍZES:		
					0 =	közbenső pozícionálás G0 egyenessel
					1 =	közbenső pozícionálás körpályán
				SZÁZAS:	foglalt	
				EZRES:		
					0 =	kompatibilitás módus, ha <INDA> = 0 akkor teljes kör, <INDA> <> 0 akkor rész-kör)
					1 =	teljes kör
					2 =	rész-kör
20	DZF	<_MIDF>	REAL	foglalt		
21		<FFP2>	REAL	foglalt		
22		<SSF>	REAL	foglalt		
23	FF	<_FFCP>	REAL	foglalt		
24		<_UMODE>	INT	foglalt		
25	FS	<_FS>	REAL	letörés szélesség (növ)		
26	ZFS	<_ZFS>	REAL	bemerülési mélység (szerszámcsúcs) letörésnél (absz/növ), lásd <_AMODE>		
27		<_GMODE>	INT	geometria módus (programozott geometria-értékek értékelése)		
				EGYES:	foglalt	
				TÍZES:	foglalt	
				SZÁZAS:	megmunkálás kiválasztása vagy csak kezdőpont kiszámítás	
					0 =	kompatibilitás módus
					1 =	normál megmunkálás

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
28		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/18/19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
				TÍZES:	foglalt	
				SZÁZAS:	foglalt	
				EZRES:	SW verzió jelölés	
					1 =	SLOT2 funkciók SW 2.5-től
				TÍZEZRES:	Technológia skálázás ciklus maszkokon belül (Oldal 818)	
					0 =	beadás: teljes
					1 =	beadás: egyszerű
29		<_AMODE>	INT	alternatív módus		
				EGYES:	végső mélység Z1 (absz/növ)	
					0 =	kompatibilitás
					1 =	Z1 (növ)
					2 =	Z1 (absz)
				TÍZES:	foglalt	
				SZÁZAS:	bemerülési mélység letörésnél ZFS	
					0 =	ZFS (absz)
					1 =	ZFS (növ)

20.1.9 LONGHOLE - Hosszlyuk

Szintaxis

```
LONGHOLE(<RTP>, <RFP>, <SDIS>, <_DP>, <_DPR>, <NUM>, <LENG>, <_CPA>,
<_CPO>, <RAD>, <STA1>, <INDA>, <FFD>, <FFP1>, <MID>, <_VARI>,
<_UMODE>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés
1	RP	<RTP>	REAL	visszahúzási sík (absz.)
2	Z0	<_RFP>	REAL	szerszám-tengely vonatkoztatási pont (absz.)
3	SC	<SDIS>	REAL	biztonsági távolság (vonatkoztatási ponthoz adódik, előjel nélkül beadni)
4	Z1	<_DP>	REAL	hosszlyuk-mélység (absz)

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
5		<_DPR>	REAL	hosszlyuk-mélység (növ), Z0-ra vonatkoztatva (előjel nélkül beadni)		
6		<NUM>	INT	hosszlyukak száma =1		
7	L	<LENG>	REAL	hosszlyuk hossza		
8	X0	<_CPA>	REAL	vonatkoztatási pont 1. tengelye irányában		
9	Y0	<_CPO>	REAL	vonatkoztatási pont 2. tengelye irányában		
10		<RAD>	REAL	foglalt		
11	α0	<STA1>	REAL	forgásszög		
12		<INDA>	REAL	foglalt		
13	FZ	<FFD>	REAL	mélységi fogásvétel előtolás		
14	F	<FFP1>	REAL	előtolás		
15	DZ	<MID>	REAL	maximális fogásmélység		
16		<_VARI>	INT	megmunkálási mód		
				EGYES:	fogásvétel mód	
					1 =	függőleges G1-gyel
					3 =	ingázva
				SZÁZAS:	foglalt	
17		<_UMODE>	INT	foglalt		
18		<_GMODE>	INT	geometria módus (programozott geometria-értékek értékelése)		
				EGYES:	foglalt	
				TÍZES:	foglalt	
				SZÁZAS:	megmunkálás kiválasztása vagy csak kezdőpont kiszámítás	
					0 =	kompatibilitás módus
					1 =	normál megmunkálás
				EZRES:	vonatkoztatási pont beméretezése, vájathossz	
					0 =	közép
					1 =	balra belül +L
					2 =	jobbra belül -L
					3 =	bal perem +L
					4 =	jobb perem -L

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
19		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/18/19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
				TÍZES:	előtolás mód: G-csoport (G94/G95) felületi és mélységi előtoláshoz	
					0 =	kompatibilitás módus
					1 =	G utasítás mint a ciklushívás előtt. G94/G95 felületi és mélységi előtoláshoz azonos
				SZÁZAS:	foglalt	
				EZRES:	SW verzió jelölés	
					1 =	LONGHOLE funkció-bővítés (vonatkoztatási pont beméretezés)
20		<_AMODE>	INT	alternatív módus		
				EGYES:	végső mélység Z1 (absz/növ)	
					0 =	kompatibilitás
					1 =	Z1 (növ)
					2 =	Z1 (absz)

Megjegyzés

A ciklus a korábbi SW verziókhoz képest új funkciókkal van ellátva. Ennek az a következménye, hogy bizonyos paraméterek a beadási maszkban már nincsenek kijelezve (<NUM>, <RAD>, <INDA>). Több vájatot egy pozíció-mintában "MCALL"-lal és a kívánt pozíció-minta felhívását pl. HOLES2-vel kell programozni.

20.1.10 CYCLE60 - Gravírozás ciklus

Szintaxis

```
CYCLE60(<_TEXT>, <_RTP>, <_RFP>, <_SDIS>, <_DP>, <_DPR>, <_PA>,
<_PO>, <_STA>, <_CP1>, <_CP2>, <_WID>, <_DF>, <_FFD>, <_FFP1>,
<_VARI>, <_CODEP>, <_UMODE>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés
1		<_TEXT>	STRING [200]	gravírozandó szöveg (maximum 100 karakter
2	RP	<_RTP>	REAL	visszahúzási sík (absz.)
3	Z0	<_RFP>	REAL	szerszám-tengely vonatkoztatási pont (absz.)
4	SC	<_SDIS>	REAL	biztonsági távolság (referencia síkhoz adódik, előjel nélkül beadni)
5	Z1	<_DP>	REAL	mélység (absz), lásd <_AMODE>
6	Z1	<_DPR>	REAL	mélység (növ), lásd <_AMODE>
7	X0	<_PA>	REAL	vonatkoztatási pont 1. tengelye a síknak (absz) - derékszögű. lásd <_VARI>
	R			vonatkoztatási pont, hossz (sugár) - polár, lásd <_VARI>
8	Y0	<_PO>	REAL	vonatkoztatási pont 2. tengelye a síknak (absz) - derékszögű. lásd <_VARI>
	α0			vonatkoztatási pont, szög az 1. tengelyre vonatkoztatva - polár, lásd <_VARI>
9	α1	<_STA>	REAL	szöveg iránya, szöveg-vonal szöge az 1. tengelyre vonatkoztatva, lásd <_VARI>
10	XM	<_CP1>	REAL	szöveg-kör középpontja, az 1. tengelye a síknak (absz) - derékszögű. lásd <_VARI>
	LM			szöveg-kör középpontja, hossz (sugár) a WINP-re vonatkoztatva - polár, lásd <_VARI>
11	YM	<_CP2>	REAL	szöveg-kör középpontja, az 2. tengelye a síknak (absz) - derékszögű. lásd <_VARI>
	αM			Szöveg-kör középpontja, szög az 1. tengelyre vonatkoztatva - polár, lásd <_VARI>
12	W	<_WID>	REAL	karakter magasság (előjel nélkül beadni)
13	DX1 DX2	<_DF>	REAL	karakter távolság / össz-szélesség, lásd <_VARI>
	α2			nyílásszög, lásd <_VARI>
14	FZ	<_FFD>	REAL	fogásvétel előtolás mélység, lásd <_DMODE>
15	F	<_FFP1>	REAL	előtolás felület megmunkáláshoz

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
16		<_VARI>	INT	megmunkálás (gravírozás szöveg iránya és vonatkoztatási pontja)		
				EGYES:	vonatkoztatási pont	
					0 =	derékszögű
					1 =	polár
				TÍZES:	szöveg iránya	
					0 =	szöveg egy vonalon
					1 =	szöveg egy köríven felül
					2 =	szöveg egy köríven alul
				SZÁZAS:	foglalt	
				EZRES:	szöveg vonatkoztatási pontja vízszintes	
					0 =	balra
					1 =	központos
					2 =	jobbra
				TÍZEZRES:	szöveg vonatkoztatási pontja függőleges	
					0 =	lent
					1 =	központos
					2 =	fent
				SZÁZEZRES:	szöveghossz	
					0 =	karakter távolság
					1 =	szöveg teljes hossza (csak egyenes szövegnél)
					2 =	nyílásszög (csak szövegnél köríven)
				EGYMILLIÓS:	kör középpont	
					0 =	derékszögű (kartéziánus)
					1 =	polár
				TÍZMILLIÓS:	tükörírás	
					0 =	kompatibilitás
					1 =	tükörírás BE
					2 =	tükörírás KI
17		<_CODEP>	INT	írás codepage száma (jelenleg csak 1252)		
18		<_UMODE>	INT	foglalt		
19		_GMODE>	INT	geometria módus (programozott geometria-értékek értékelése)		
				EGYES:	foglalt	
				TÍZES:	foglalt	
				SZÁZAS:	megmunkálás kiválasztása / csak kezdőpont kiszámítás	
					0 =	kompatibilitás módus
					1 =	normál megmunkálás

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
20		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/18/19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
				TÍZES:	Előtolás mód: G-csoport (G94/G95) felületi és mélységi előtoláshoz	
					0 =	kompatibilitás módus
					1 =	G utasítás mint a ciklushívás előtt. G94/G95 felületi és mélységi előtoláshoz azonos
21		<_AMODE>	INT	alternatív módus		
				EGYES:	végső mélység (<_DP>, <_DPR>)	
					0 =	kompatibilitás
					1 =	növekményes (<_DPR>)
					2 =	abszolút (<_DP>)

20.1.11 CYCLE61 - Síkmarás

Szintaxis

```
CYCLE61(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_PA>, <_PO>, <_LENG>,
<_WID>, <_MID>, <_MIDA>, <_FALD>, <_FFP1>, <_VARI>, <_LIM>,
<_DMODE>, <_AMODE>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés
1	RP	<_RTP>	REAL	visszahúzási sík (absz.)
2	Z0	<_RFP>	REAL	szerszám-tengely vonatkoztatási pont, nyersdarab magasság (absz.)
3	SC	<_SDIS>	REAL	biztonsági távolság (vonatkoztatási ponthoz adódik, előjel nélkül beadni)
4	Z1	<_DP>	REAL	készdarab magasság (absz/növ), lásd <_AMODE>
5	X0	<_PA>	REAL	sarokpont 1 az 1. tengelyen (absz)
6	Y0	<_PO>	REAL	sarokpont 1 az 2. tengelyen (absz)
7	X1	<_LENG>	REAL	sarokpont 2 az 1. tengelyen (absz/növ), lásd <_AMODE>
8	Y1	<_WID>	REAL	sarokpont 2 az 2. tengelyen (absz/növ), lásd <_AMODE>
9	DZ	<_MID>	REAL	maximális fogásmélység
10	DXY	<_MIDA>	REAL	maximális sík fogásvétel (egységet lásd <_AMODE>)

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
11	UZ	<_FALD>	REAL	simítási ráhagyás mélység		
12	F	<_FFP1>	REAL	megmunkálási előtolás		
13		<_VARI>	INT	Megmunkálási mód		
				EGYES:	Megmunkálás	
					1 =	nagyolás
					2 =	simítás
				TÍZES:	megmunkálási irány	
					1 =	párhuzamosan az 1. tengellyel, egy irány
					2 =	párhuzamosan a 2. tengellyel, egy irány
					3 =	párhuzamosan az 1. tengellyel, váltakozó irány
					4 =	párhuzamosan a 2. tengellyel, váltakozó irány
14		<_LIM>	INT	határolások		
				EGYES:	határolás 1. tengely mínusz	
					0 =	nem
					1 =	igen
				TÍZES:	határolás 1. tengely plusz	
					0 =	nem
					1 =	igen
				SZÁZAS:	határolás 2. tengely mínusz	
					0 =	nem
					1 =	igen
				EZRES:	határolás 2. tengely plusz	
					0 =	nem
					1 =	igen
15		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/18/19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
16		<_AMODE>	INT	alternatív módus		
				EGYES:	végső mélység (<_DP>)	
					0 =	abszolút
					1 =	növekményes
				TÍZES:	egység sík fogásvételhez (<_MIDA>)	
					0 =	mm
					1 =	szerszámátmérő %-a
				SZÁZAS:	foglalt	
				EZRES:	felület hossza	
					0 =	növekményes
					1 =	abszolút
				TÍZEZRES:	felület szélessége	
					0 =	növekményes
					1 =	abszolút

20.1.12 CYCLE62 - Kontúrhívás

Szintaxis

```
CYCLE62(<_KNAME>,<_TYPE>, <_LAB1>, <_LAB2>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
1	PRG/CON	<_KNAME>	STRING [140]	kontúrnév vagy alprogramnév, nem kell programozni _TYPE = 2 esetén		
2		<_TYPE>	INT	kontúr-beadás meghatározása		
					0 =	alprogram
					1 =	kontúrnév
					2 =	címkék
					3 =	címkék alprogramban
3	LAB1	<_LAB1>	STRING[32]	címke 1, kontúr kezdet		
4	LAB2	<_LAB2>	STRING[32]	címke 2, kontúr vége		

20.1.13 CYCLE63 - Kontúrzsebet marni

Szintaxis

```
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>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
1	PRG	<_PRG>	STRING [100]	kiforgácsoló program neve		
2		<_VARI>	INT	megmunkálási mód		
				EGYES:	technológiai megmunkálás	
					1 =	nagyolás
					3 =	simítás az alján
					4 =	simítás a peremen
					5 =	letörés
				TÍZES:	fogásvétel mód	
					0 =	bemerülés központosan
					1 =	bemerülés helikálisan
					2 =	bemerülés ingázva
				SZÁZAS:	foglalt	
				EZRES:	leemelés modus	
					0 =	leemelés a visszahúzási síkra
					1 =	leemelés a vonatkoztatási pont + biztonsági távolságra
				TÍZEZRES:	kezdőpont fenék nagyolásnál és simításnál	
					0 =	auto
					1 =	kézi
3	RP	<_RP>	REAL	visszahúzási sík (absz.)		
4	Z0	<_Z0>	REAL	szerszám-tengely vonatkoztatási pont (absz.)		
5	SC	<_SC>	REAL	biztonsági távolság (vonatkoztatási ponthoz adódik, előjel nélkül beadni)		
6	Z1	<_Z1>	REAL	végső mélység, (lásd <_AMODE> EGYES)		
7	F	<_F>	REAL	nagyolás/simítás előtolás a síkban		
8	FZ	<_FZ>	REAL	mélységi fogásvétel előtolás		
9	DXY	<_DXY>	REAL	sík fogásvétel - egység, (lásd <_AMODE> TIZES)		
10	DZ	<_DZ>	REAL	mélységi fogásvétel		
11	UXY	<_UXY>	REAL	sík simítási ráhagyás		
12	UZ	<_UZ>	REAL	simítási ráhagyás mélység		

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
13		<_CDIR>	INT	Marásirány	0 =	egyirányú
					1 =	ellenirányú
14	XS	<_XS>	REAL	kezdőpont X, abszolút		
15	YS	<_YS>	REAL	kezdőpont Y, abszolút		
16	ER	<_ER>	REAL	bemerülés helikálisan: sugár		
17	EP	<_EP>	REAL	bemerülés helikálisan: emelkedés		
18	EW	<_EW>	REAL	bemerülés ingázva: maximális bemerülési szög		
19	FS	<_FS>	REAL	letörés szélesség (növ) letörésnél		
20	ZFS	<_ZFS>	REAL	szerszámcsúcs bemerülés mélysége letörésnél (lásd <_AMODE> SZÁZAS)		
21	TR	<_TR>	STRING[32]	referencia szerszám neve maradékanyag megmunkálásnál		
22	DR	<_DR>	INT	referencia szerszám D-száma maradékanyag megmunkálásnál		
23		<_UMODE>	INT	foglalt		
24		<_GMODE>	INT	geometria módus (programozott geometria-értékek értékelése)		
				EGYES:	foglalt	
				TÍZES:	foglalt	
				SZÁZAS:	megmunkálás kiválasztása / csak kezdőpont kiszámítás	
					0 =	normál megmunkálás (kompatibilitás módus nem szükséges)
					1 =	normál megmunkálás
					2 =	foglalt
25		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/G18/G19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
				TÍZES:	foglalt	
				SZÁZAS:	technológia módus	
					1 =	zseb
					2 =	csap
				EZRES:	maradékanyag megmunkálás	
					0 =	nem
					1 =	igen
				TÍZEZRES:	Technológia skálázás ciklus maszkokon belül (Oldal 818)	
					0 =	beadás: teljes
					1 =	beadás: egyszerű

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
26		<_AMODE>	INT	alternatív módus		
				EGYES:	végső mélység (Z1)	
					0 =	abszolút (kompatibilitás módus)
					1 =	növekményes
				TÍZES:	egység sík fogásvételhez (DXY)	
					0 =	mm
					1 =	szerszámátmérő %-a
				SZÁZAS:	bemerülési mélység letörésnél (ZFS)	
					0 =	abszolút
					1 =	növekményes
				EZRES:	---	foglalt

20.1.14 CYCLE64 - Kontúrzsebet előfúrni

Szintaxis

```
CYCLE64(<_PRG>, <_VARI>, <_RP>, <_Z0>, <_SC>, <_Z1>, <_F>, <_DXY>,
<_UXY>, <_UZ>, <_CDIR>, <_TR>, <_DR>, <_UMODE>, <_GMODE>, <_DMODE>,
<_AMODE>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
1	PRG	<_PRG>	STRING [100]	fúró/központozó program neve		
2		<_VARI>	INT	Megmunkálási mód		
				EGYES:	foglalt	
				TÍZES:	foglalt	
				SZÁZAS:	foglalt	
				EZRES:	Leemelés modus	
					0 =	leemelés a visszahúzási síkra
					1 =	leemelés a vonatkoztatási pont + biztonsági távolságra
3	RP	<_RP>	REAL	visszahúzási sík (absz.)		
4	Z0	<_Z0>	REAL	vonatkoztatási pont (absz.)		
5	SC	<_SC>	REAL	biztonsági távolság (vonatkoztatási ponthoz adódik, előjel nélkül beadni)		
6	Z1	<_Z1>	REAL	fúrás/központozás mélység, (lásd <_AMODE> EGYES)		
7	F	<_F>	REAL	fúrás/központozás előtolás		
8	DXY	<_DXY>	REAL	sík fogásvétel - egység, (lásd <_AMODE> TIZES)		
9	UXY	<_UXY>	REAL	sík simítási ráhagyás		

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
10	UZ	<_UZ>	REAL	simítási ráhagyás mélység		
11		<_CDIR>	INT	marásirány	0 =	egyirányú
					1 =	ellenirányú
12	TR	<_TR>	STRING[20]	referencia szerszám		
13	DR	<_DR>	INT	referencia szerszám D-szám		
14		<_UMODE>	INT	foglalt		
15		<_GMODE>	INT	geometria módus (programozott geometria-értékek értékelése)		
				EGYES:	foglalt	
				TÍZES:	foglalt	
				SZÁZAS:	megmunkálás kiválasztása / csak kezdőpont kiszámítás	
					0 =	normál megmunkálás (kompatibilitás módus nem szükséges)
					1 =	normál megmunkálás
					2 =	foglalt
25		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/G18/G19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
				TÍZES:	technológia módus)	
					1 =	előfúrás
					2 =	központozás
26		<_AMODE>	INT	alternatív módus		
				EGYES:	fúrás/központozás mélység Z1	
					0 =	abszolút (kompatibilitás módus)
					1 =	növekményes
				TÍZES:	egység sík fogásvételhez (DXY)	
					0 =	mm
					1 =	szerszámátmérő %-a

20.1.15 CYCLE70 - Menetmarás

Szintaxis

```
CYCLE70(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_DIATH>, <_H1>, <_FAL>,
<_PIT>, <_NT>, <_MID>, <_FFR>, <_TYPTH>, <_PA>, <_PO>, <_NSP>,
<_VARI>, <_PITA>, <_PITM>, <_PTAB>, <_PTABA>, <_GMODE>, <_DMODE>,
<_AMODE>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
1	RP	<_RTP>	REAL	visszahúzási sík (absz.)		
2	Z0	<_RFP>	REAL	szerszám-tengely vonatkoztatási pont (absz.)		
3	SC	<_SDIS>	REAL	biztonsági távolság (vonatkoztatási ponthoz adódik, előjel nélkül beadni)		
4	Z1	<_DP>	REAL	menethossz (absz, növ), lásd <_AMODE> kifutást a furatfenéken figyelembe venni (minimum fél emelkedés)		
5	∅	<_DIATH>	REAL	menet névleges átmérő:		
6	H1	<_H1>	REAL	menet-mélység		
7	U	<_FAL>	REAL	simítási ráhagyás		
8	P	<_PIT>	REAL	menetemelkedés (kiválasztás <_PITA>: mm, hüvelyk, MODUL, menet/hüvelyk)		
9	NT	<_NT>	INT	fogak száma a vágólapkán Szerszámhossz mindig az alsó fogra vonatkozik!		
10	DXY	<_MID>	REAL	maximális fogás vágásonként <_MID> > <_H1>: mindent egy vágással		
11	F	<_FFR>	REAL	marás előtolás		
12		<_TYPTH>	INT	menettípus	0 =	belső menet
					1 =	külső menet
13	X0	<_PA>	REAL	kör középpont, 1. tengelyen (absz)		
14	Y0	<_PO>	REAL	kör középpont, 2. tengelyen (absz)		
15	αS	<_NSP>	REAL	kezdőszög (több bekezdésű menet)		
16		<_VARI>	INT	Megmunkálási mód		
				EGYES:		
					1 =	nagyolás
					2 =	simítás
				TÍZES:		
					1 =	fentről lefelé
					2 =	lentről felfelé
				SZÁZAS:		
					0 =	jobbmenet
					1 =	balmenet
17		<_PITA>	INT	menetemelkedés értékelése		
					0 =	kompatibilitás módus
					1 =	emelkedés mm-ben
					2 =	emelkedés járat/hüvelykben (TPI)
					3 =	emelkedés hüvelykben
					4 =	emelkedés MODUL-ként
18		<_PITM>	STRING[15]	string menetemelkedési beadás jelölőjeként (csak a felülethez)		
19		<_PTAB>	STRING[20]	string menettáblázathoz ("", "ISO", "BSW", "BSP", "UNC") (csak a felülethez)		

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
20		<_PTABA>	STRING[20]	string kiválasztáshoz a menettáblázatban (pl. "M 10", "M 12", ...) (csak a felülethez)		
21		<_GMODE>	INT	geometria módus (programozott geometria-értékek értékelése)		
				EGYES:	foglalt	
				TÍZES:	foglalt	
				SZÁZAS:	megmunkálás / kezdőpont kiszámítás	
					0 =	kompatibilitás módus
					1 =	normál megmunkálás
22		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/G18/G19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
23		<_AMODE>	INT	alternatív módus		
				EGYES:	menethossz (<_DP>)	
					0 =	abszolút
					1 =	növekményes

20.1.16 CYCLE72 - Pályamarás

Szintaxis

```
CYCLE72(<_KNAME>, <_RTP>, <_RFP>, <_SDIS>, <_DP>, <_MID>, <_FAL>,
<_FALD>, <_FFP1>, <_FFD>, <_VARI>, <_RL>, <_AS1>, <_LP1>, <_FF3>,
<_AS2>, <_LP2>,<_UMODE>, <_FS>, <_ZFS>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés
1		<_KNAME>	STRING [141]	kontúr alprogram neve
2	RP	<_RTP>	REAL	visszahúzási sík (absz.)
3	Z0	<_RFP>	REAL	szerszám-tengely vonatkoztatási pont (absz.)
4	SC	<_SDIS>	REAL	biztonsági távolság (vonatkoztatási ponthoz adódik, előjel nélkül beadni)
5	Z1	<_DP>	REAL	végpont, végső mélység (absz/növ), lásd <_AMODE>
6	DZ	<_MID>	REAL	maximális mélységi fogásvétel (növ, előjel nélkül beadni)
7	UXY	<_FAL>	REAL	sík simítási ráhagyás (növ), a perem-kontúron

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
8	UZ	<_FALD>	REAL	simítási ráhagyás mélység (növ), ráhagyást a fenéken előjel nélkül beadni)		
9	FX	<_FFP1>	REAL	előtolás a kontúron		
10	FZ	<_FFD>	REAL	előtolás mélység fogásvételhez (vagy térbeli fogásvételhez)		
11		<_VARI>	INT	megmunkálási mód		
				EGYES:	megmunkálás	
					1 =	nagyolás
					2 =	simítás
					5 =	letörés
				TÍZES:		
					0 =	közbenső utak G0-val
					1 =	közbenső utak G1-gyel
				SZÁZAS:	visszahúzás a kontúr végén	
					0 =	visszahúzás a kontúr végén vonatkoztatási pontra
					1 =	visszahúzás a kontúr végén vonatkoztatási pontra + <_SDIS>
					2 =	visszahúzás a kontúr végén <_SDIS>-val
					3 =	nincs visszahúzás a kontúr végén, következő kezdőpont felvétele kontúr-előtolással
				EZRES:	foglalt	
				TÍZEZRES:	kontúrt megmunkálni	
					0 =	kontúrt előrefelé megmunkálni
					1 =	kontúrt hátrafelé megmunkálni korlátozások visszafelé esetén: • max. 170 kontúrelem (beleértve a letöréseket és lekerekítéseket) • csak a síkbeli értékek (X/Y) és F lesz kiértékelve
12		<_RL>	INT	megmunkálási irány		
					40 =	kontúr középen (G40, rá- és lemenet: egyenes vagy merőleges)
					41 =	kontúrtól balra (G41, rá- és lemenet: egyenes vagy kör)
					42 =	kontúrtól jobbra (G42, rá- és lemenet: egyenes vagy kör)

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
13		<_AS1>	INT	kontúr rámenet mozgás		
				EGYES:		
					1 =	egyenes
					2 =	negyedkör
					3 =	félkör
					4 =	merőleges rá- és lemenet
				TÍZES:		
					0 =	utolsó mozgás, síkban
					1 =	utolsó mozgás, térben
14	L1	<_LP1>	REAL	rámenet út vagy rámenet sugár (növ, előjel nélkül beadni)		
15	FZ	<_FF3>	REAL	előtolás közbenső utakra (G94/G95 mint a kontúron)		
16		<_AS2>	INT	kontúr lemenet mozgás (nem merőleges rá-/lemenet)		
				EGYES:		
					1 =	egyenes
					2 =	negyedkör
					3 =	félkör
				TÍZES:		
					0 =	utolsó mozgás, síkban
					1 =	utolsó mozgás, térben
17	L2	<_LP2>	REAL	lemenet út vagy lemenet sugár (növ, előjel nélkül beadni)		
18		<_UMODE>	INT	foglalt		
19	FS	<_FS>	REAL	letörés szélesség (növ)		
20	ZFS	<_ZFS>	REAL	bemerülési mélység (szerszámcsúcs) letörésnél (absz/növ), lásd <_AMODE>		
21		<_GMODE>	INT	geometria módus (programozott geometria-értékek értékelése)		
				EGYES:	foglalt	
				TÍZES:	foglalt	
				SZÁZAS:	megmunkálás kiválasztása / csak kezdőpont kiszámítás	
					0 =	kompatibilitás módus
					1 =	normál megmunkálás

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
22		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/18/19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
				TÍZES:	Előtolás mód: G-csoport (G94/G95) felületi és mélységi előtoláshoz	
					0 =	kompatibilitás módus
					1 =	G utasítás mint a ciklushívás előtt. G94/G95 felületi és mélységi előtoláshoz azonos
				EZRES:		
					0 =	kompatibilitás módus: kontúrnév <_KNAME>ben
					1 =	kontúrnév a CYCLE62-ben lesz programozva és a _SC_CONT_NAME-ben átadva
23		<_AMODE>	INT	alternatív módus		
				EGYES:	végpont Z1 (<_DP>)	
					0 =	abszolút (kompatibilitás módus)
					1 =	növekményes
				TÍZES:	egység sík fogásvételhez	
					0 =	mm, hüv
					1 =	foglalt
				SZÁZAS:	bemerülési mélység letörésnél (<_ZFS>)	
					0 =	abszolút
					1 =	növekményes

Megjegyzés

Ha a következő átadási paraméterek közvetetten (paraméterként) vannak programozva, a beadási maszk nem lesz visszafordítva:

```
<_VARI>, <_RL>, <_AS1>, <_AS2>, <_UMODE>, <_GMODE>, <_DMODE>,
<_AMODE>
```

20.1.17 CYCLE76 - négyszögcsap marás

Szintaxis

```
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>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
1	RP	<_RTP>	REAL	visszahúzási sík (absz.)		
2	Z0	<_RFP>	REAL	szerszám-tengely vonatkoztatási pont (absz.)		
3	SC	<_SDIS>	REAL	biztonsági távolság (vonatkoztatási ponthoz adódik, előjel nélkül beadni)		
4	Z1	<_DP>	REAL	csapmélység (absz)		
5		<_DPR>	REAL	csapmélység (növ) Z0-ra vonatkoztatva (előjel nélkül beadni)		
6	L	<_LENG>	REAL	csaphossz, lásd <_GMODE> (előjel nélkül beadni)		
7	W	<_WID>	REAL	csapszélesség, lásd <_GMODE> (előjel nélkül beadni)		
8	R	<_CRAD>	REAL	csap saroksugara (előjel nélkül beadni)		
9	X0	<_PA>	REAL	csap vonatkoztatási pont, az 1. tengelye a síknak (absz)		
10	Y0	<_PO>	REAL	csap vonatkoztatási pont, a 2. tengelye a síknak (absz)		
11	α0	<_STA>	REAL	forgatás-szög, szög a hossztengely (L) és az 1. tengelye a síknak		
12	DZ	<_MID>	REAL	maximális mélységi fogásvétel (növ, előjel nélkül beadni)		
13	UXY	<_FAL>	REAL	sík simítási ráhagyás (növ), a perem-kontúron		
14	UZ	<_FALD>	REAL	mélységi simítási ráhagyás (növ) a fenéken (előjel nélkül beadni)		
15	FX	<_FFP1>	REAL	előtolás a kontúron		
16	FZ	<_FFD>	REAL	mélységi fogásvétel előtolás		
17		<_CDIR>	INT	marásirány (előjel nélkül beadni)		
				EGYES:		
					0 =	egyirányú
					1 =	ellenirányú
18		<_VARI>	INT	Megmunkálás		
				EGYES:		
					1 =	nagyolás
					2 =	simítás
					5 =	letörés
19	L1	<_AP1>	REAL	nyersdarab csap hossza		
20	W1	<_AP2>	REAL	nyersdarab csap szélessége		
21	FS	<_FS>	REAL	letörés szélesség (növ)		
22	ZFS	<_ZFS>	REAL	bemerülési mélység (szerszámcsúcs) letörésnél (absz, növ), lásd <_AMODE>		

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
23		<_GMODE>	INT	geometria módus (programozott geometria-értékek értékelése)		
				EGYES:	foglalt	
				TÍZES:	foglalt	
				SZÁZAS:	megmunkálás kiválasztása vagy csak kezdőpont kiszámítás	
					0 =	kompatibilitás módus
					1 =	normál megmunkálás
				EZRES:	csap beméretezése középpel vagy sarokkal	
					0 =	kompatibilitás módus
					1 =	méretezés középpel
					2 =	sarokpont méretezés, csap +L +W
					3 =	sarokpont méretezés, csap -L +W
					4 =	sarokpont méretezés, csap +L -W
					5 =	sarokpont méretezés, csap -L -W
				TÍZEZRES:	teljes megmunkálás vagy utó-megmunkálás	
					0 =	kompatibilitás módus
					1 =	teljes megmunkálás
					2 =	utó-megmunkálás
24		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/18/19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
				TÍZES:	---	foglalt
				SZÁZAS:	---	foglalt
				EZRES:	---	foglalt
				TÍZEZRES:	Technológia skálázás ciklus maszkokon belül (Oldal 818)	
					0 =	beadás: teljes
					1 =	beadás: egyszerű

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
25		<_AMODE>	INT	alternatív módus		
				EGYES:	végső mélység Z1 (DP)	
					0 =	kompatibilitás
					1 =	növekményes
					2 =	abszolút
				TÍZES:	foglalt	
				SZÁZAS:	bemerülési mélység letörésnél (ZFS)	
					0 =	abszolút
					1 =	növekményes

20.1.18 CYCLE77 - Körcsap marása

Szintaxis

```
CYCLE77(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_DPR>, <_CDIAM>, <_PA>,
<_PO>, <_MID>, <_FAL>, <_FALD>, <_FFP1>, <_FFD>, <_CDIR>, <_VARI>,
<_AP1>, <_FS>, <_ZFS>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
1	RP	<_RTP>	REAL	visszahúzási sík (absz.)		
2	Z0	<_RFP>	REAL	szerszám-tengely vonatkoztatási pont (absz.)		
3	SC	<_SDIS>	REAL	biztonsági távolság (vonatkoztatási ponthoz adódik, előjel nélkül beadni)		
4	Z1	<_DP>	REAL	csapmélység (absz)		
5		<_DPR>	REAL	csapmélység (növ) Z0-ra vonatkoztatva (előjel nélkül beadni)		
6	∅	<_CDIAM>	REAL	csap átmérője (előjel nélkül beadni)		
7	X0	<_PA>	REAL	csap vonatkoztatási pont, az 1. tengelye a síknak (absz)		
8	Y0	<_PO>	REAL	csap vonatkoztatási pont, a 2. tengelye a síknak (absz)		
9	DZ	<_MID>	REAL	maximális mélységi fogásvétel (növ, előjel nélkül beadni)		
10	UXY	<_FAL>	REAL	sík simítási ráhagyás (növ), a perem-kontúron		
11	UZ	<_FALD>	REAL	mélységi simítási ráhagyás (növ) a fenéken (előjel nélkül beadni)		
12	FX	<_FFP1>	REAL	előtolás a kontúron		
13	FZ	<_FFD>	REAL	mélységi fogásvétel előtolás		
14		<_CDIR>	INT	marásirány (előjel nélkül beadni)		
				EGYES:		
					0 =	egyirányú
					1 =	ellenirányú

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
15		<_VARI>	INT	megmunkálási mód		
				EGYES:	Megmunkálás	
					1 =	nagyolás a simítási ráhagyásig
					2 =	simítás (ráhagyás X/Y/Z = 0)
					5 =	letörés
16	∅1	<_AP1>	REAL	nyersdarab csap átmérője		
17	FS	<_FS>	REAL	letörés szélesség (növ)		
18	ZFS	<_ZFS>	REAL	bemerülési mélység (szerszámcsúcs) letörésnél (absz/növ), lásd <_AMODE>		
19		<_GMODE>	INT	geometria módus (programozott geometria-értékek értékelése)		
				EGYES:	foglalt	
				TÍZES:	foglalt	
				SZÁZAS:	megmunkálás kiválasztása / csak kezdő-pont kiszámítás	
					0 =	kompatibilitás módus
					1 =	normál megmunkálás
				EZRES:	foglalt	
				TÍZEZRES:	teljes megmunkálás / utó-megmunkálás	
					0 =	kompatibilitás módus (<_AP1> mint eddig kezelni)
					1 =	teljes megmunkálás
					2 =	utó-megmunkálás
20		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/18/19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
				TÍZES:	---	foglalt
				SZÁZAS:	---	foglalt
				EZRES:	---	foglalt
				TÍZEZRES:	Technológia skálázás ciklus maszkokon belül (Oldal 818)	
					0 =	beadás: teljes
					1 =	beadás: egyszerű

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
21		<_AMODE>	INT	alternatív módus		
				EGYES:	végső mélység Z1 (DP)	
					0 =	abszolút (kompatibilitás módus)
					1 =	növekményes
					2 =	abszolút
				TÍZES:	foglalt	
				SZÁZAS:	bemerülési mélység letörésnél (ZFS)	
					0 =	abszolút
					1 =	növekményes

20.1.19 CYCLE78 - Furatmenet marás

Szintaxis

```
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>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
1	RP	<_RTP>	REAL	visszahúzási sík (absz.)		
2	Z0	<_RFP>	REAL	szerszám-tengely vonatkoztatási pont (absz.)		
3	SC	<_SDIS>	REAL	biztonsági távolság (vonatkoztatási ponthoz adódik, előjel nélkül beadni)		
4	Z1	<_DP>	REAL	végső furatmélység (absz/növ), lásd <_AMODE>		
5		<_ADPR>	REAL	felfúrás mélység csökkentett pályaelőtolással (növ.), <_VARI> TÍZEZRES-sel hatásos		
6	D	<_FDPR>	REAL	maximális fogásmélység (növ) D ≥ Z1 ⇒ egy fogás végső furatmélységre D < Z1 ⇒ mélyfúró ciklus több fogással és forgácstalanítással		
7	ZR	<_LDPR>	REAL	maradék furatmélység átfúrásnál (növ), FR előtolással		
8	∅	<_DIAM>	REAL	menet névleges átmérő:		
9	P	<_PIT>	REAL	menetemelkedés számértékként		
10		<_PITA>	INT	P menetemelkedés értékelése		
					1 =	emelkedés mm/ford-ban
					2 =	emelkedés járat/hüvelykben
					3 =	emelkedés hüv/ford-ban
					4 =	emelkedés MODUL-ként
11	DF	<_DAM>	REAL	érték/százalék minden további fogáshoz (degresszió), lásd <_AMODE>		

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
12	V1	<_MDEP>	REAL	minimális fogás (növ), csak degressziónál hatásos		
13		<_VARI>	INT	Megmunkálási mód		
				EGYES:	foglalt	
				TÍZES:	forgácstalanítás menetmarás előtt	
					0 =	nincs forgácstalanítás menetmarás előtt (csak a végső furatmélységre hat)
					1 =	nincs forgácstalanítás menetmarás előtt (csak a végső furatmélységre hat)
				SZÁZAS:	jobb-/balmenet	
					0 =	jobbmenet
					1 =	balmenet
				EZRES:	maradék furatmélység fúró-előtolással	
					0 =	nincs maradék furatmélység FR előtolással
					1 =	maradék furatmélység FR előtolással
				TÍZEZRES:	felfúrni csökkentett előtolással	
					0 =	nincs felfúrás csökkentett előtolással
					1 =	felfúrás csökkentett előtolással felfúrás előtolás = 0,3 F1, Ha F1 < 0,15 mm/ford felfúrás előtolás = 0,1 mm/ford, Ha F1 ≥0,15 mm/ford
14		<_CDIR>	INT	Marásirány	0 =	egyirányú
					1 =	ellenirányú
					4 =	ellenirányú + egyenirányú (nagyolás + simítás kombináció)
15	Z2	<_GE>	REAL	visszahúzás érték menetmarás előtt (növ)		
16	F1	<_FFD>	REAL	pályaelőtolás (mm/perc ill. hüvelyk/perc vagy mm/ford)		
17	FR	<_FRDP>	REAL	pályaelőtolás maradék furatmélységre (mm/perc vagy mm/ford)		
18	F2	<_FFR>	REAL	előtolás menetmarásra (mm/perc vagy mm/fog)		
19	FS	<_FFP2>	REAL	simítás előtolás <_CDIR> =4-re (mm/perc vagy mm/fog)		
20		<_FFA>	INT	előtolások értékelése		
				EGYES:	pályaelőtolás F1	
				TÍZES:	pályaelőtolás maradék-furatmélységre FR	
				SZÁZAS:	előtolás menetmaráshoz F2	
				EZRES:	simítás-előtolás FS	
21		<_PITM>	STRING[15]	string menetemelkedési beadás jelölőjeként (csak a felülethez)[1)]		
22		<_PTAB>	STRING[20]	string menettáblázathoz ("", "ISO", "BSW", "BSP", "UNC") (csak a felülethez)[1)]		

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
23		<_PTABA>	STRING[20]	string kiválasztáshoz a menettáblázatban (pl. "M 10", "M 12", ...) (csak a felülethez)[1]		
24		<_GMODE>	INT	geometria módus (programozott geometria-értékek értékelése), foglalat		
25		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/18/19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
26		<_AMODE>	INT	alternatív módus		
				EGYES:	furatmélység = végső furatmélység Z1 absz/növ	
					0 =	abszolút
					1 =	növekményes
				TÍZES:	érték/százalék DF minden további fogáshoz (degresszió)	
					0 =	érték
					1 =	százalék (0.001 ... 100 %)

Megjegyzés

[1] A 21, 22, és 23 paraméterek csak a beadási maszk menettáblázatában a menet kiválasztásnál lesznek használva. A hozzáférés a menettáblázatokhoz ciklus-definícióval a ciklus futásakor nem lehetséges.

20.1.20 CYCLE79 - sokszög

Szintaxis

```
CYCLE79(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_NUM>, <_SWL>, <_PA>,
<_PO>, <_STA>, <_RC>, <_AP1>, <_MIDA>, <_MID>, <_FAL>, <_FALD>,
<_FFP1>, <_CDIR>, <_VARI>, <_FS>, <_ZFS>, <_GMODE>, <_DMODE>,
<_AMODE>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés
1	RP	<_RTP>	REAL	visszahúzási sík (absz.)
2	Z0	<_RFP>	REAL	szerszám-tengely vonatkoztatási pont (absz.)
3	SC	<_SDIS>	REAL	biztonsági távolság (vonatkoztatási ponthoz adódik, előjel nélkül beadni)

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
4	Z1	<_DP>	REAL	sokszög-mélység (absz/növ), lásd <_AMODE>		
5	N	<_NUM>	INT	élek száma (N = 1....n)		
6	SW/L	<_SWL>	REAL	vállszélesség vagy élhossz (<_VARI> szerint) ("SW" vállszélességnél, "L" élhossznál) vállszélesség csak páros számú élnél és egy élnél		
7	X0	<_PA>	REAL	csap vonatkoztatási pont 1. tengelyen (absz)		
8	Y0	<_PO>	REAL	csap vonatkoztatási pont 2. tengelyen (absz)		
9	α0	<_STA>	REAL	él-közép forgatásszöge az 1. tengelyhez (X tengely)		
10	R1/FS1	<_RC>	REAL	sarok lekerekítés <_NUM> > 2-nél (sugár/letörés, lásd <_AMODE>) (növ, előjel nélkül beadni) ("R1" sugárnál, "FS1" letörésnél)		
11	∅	<_AP1>	REAL	csap nyers-átmérője		
12	DXY	<_MIDA>	REAL	maximális fogás-szélesség (egység, lásd <_AMODE>)		
13	DZ	<_MID>	REAL	maximális fogásmélység		
14	UXY	<_FAL>	REAL	sík simítási ráhagyás		
15	UZ	<_FALD>	REAL	simítási ráhagyás mélység		
16	F	<_FFP1>	REAL	megmunkálási előtolás		
17		<_CDIR>	INT	Marásirány	0 =	egyirányú
					1 =	ellenirányú
18		<_VARI>	INT	Megmunkálási mód		
				EGYES:	Megmunkálás	
					1 =	nagyolás
					2 =	simítás
					3 =	simítás a peremen
					5 =	letörés
				TÍZES:	vállszélesség vagy élhossz	
					0 =	vállszélesség
					1 =	élhossz
19	FS	<_FS>	REAL	letörés szélesség (növ)		
20	ZFS	<_ZFS>	REAL	bemerülési mélység (szerszámcsúcs) letörésnél (absz/növ), lásd <_AMODE>		
21		<_GMODE>	INT	geometria módus (programozott geometria-értékek értékelése)		
				EGYES:	foglalt	
				TÍZES:	foglalt	
				SZÁZAS:	megmunkálás kiválasztása vagy csak kezdőpont kiszámítás	
					1 =	normál megmunkálás

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
22		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/18/19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
				TÍZES:	---	foglalt
				SZÁZAS:	---	foglalt
				EZRES:	---	foglalt
				TÍZEZRES:	Technológia skálázás ciklus maszkokon belül (Oldal 818)	
					0 =	beadás: teljes
					1 =	beadás: egyszerű
23		<_AMODE>	INT	alternatív módus		
				EGYES:	végső mélység (<_DP>)	
					0 =	abszolút
					1 =	növekményes
				TÍZES:	egység sík fogásvételhez (<_MIDA>)	
					0 =	mm
					1 =	szerszámátmérő %-a
				SZÁZAS:	bemerülési mélység letörésnél (<_ZFS>)	
					0 =	abszolút
					1 =	növekményes
				EZRES:	sarok lekerekítés (<_RC>)	
					0 =	sugár
					1 =	letörés

20.1.21 CYCLE81 - Fúrás, központozás

Szintaxis

```
CYCLE81(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <DTB>, <_GMODE>,
<_DMODE>, <_AMODE>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés
1	RP	<RTP>	REAL	visszahúzási sík (absz.)
2	Z0	<RFP>	REAL	vonatkoztatási pont (absz.)

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
3	SC	<SDIS>	REAL	biztonsági távolság (vonatkoztatási ponthoz adódik, előjel nélkül beadni)		
4	Z1/∅	<DP>	REAL	furatmélység (absz) / központozás átmérő (absz), lásd <_GMODE>		
5	Z1	<DPR>	REAL	furatmélység (növ.)		
6	DT	<DTB>	REAL	várakozási idő végső furatmélységen, lásd <_AMODE>		
7		<_GMODE>	INT	geometria módus (programozott geometria-értékek értékelése)		
				EGYES:	foglalt	
				TÍZES:	központozás a mélységre / átmérőre vonatkoztatva	
					0 =	kompatibilitás, mélység
					1 =	átmérő
8		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/G18/G19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
9		<_AMODE>	INT	alternatív módus		
				EGYES:	furatmélység Z1 (absz/növ)	
					0 =	kompatibilitás a DP/DPR programozásból
					1 =	növekményes
					2 =	abszolút
				TÍZES:	várakozási idő a DT végső furatmélységen másodpercekben/fordulatokban	
					0 =	kompatibilitás a DTB előjeléből (> 0 másodpercek vagy < 0 fordulatok)
					1 =	másodpercekben
					2 =	fordulatokban

20.1.22 CYCLE82 - Fúrás, sík-süllyesztés

Szintaxis

```
CYCLE82(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <DTB>, <_GMODE>,
<_DMODE>, <_AMODE>, <_VARI>, <S_ZA>, <S_FA>, <S_ZD>, <S_FD>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
1	RP	<RTP>	REAL	visszahúzási sík (absz.)		
2	Z0	<RFP>	REAL	vonatkoztatási pont (absz.)		
3	SC	<SDIS>	REAL	biztonsági távolság (vonatkoztatási ponthoz adódik, előjel nélkül beadni)		
4	Z1	<DP>	REAL	furatmélység (absz), lásd <_AMODE>		
5	Z1	<DPR>	REAL	furatmélység (növ), lásd <_AMODE>		
6	DT	<DTB>	REAL	várakozási idő végső furatmélységen, lásd <_AMODE>		
7		<_GMODE>	INT	geometria módus (programozott geometria-értékek értékelése)		
				EGYES:	foglalt	
				TÍZES:	furatmélység csúcsra/szárra vonatkoztatva	
					0 =	kompatibilitás, csúcs
					1 =	szár
8		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/G18/G19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
				TÍZES:	foglalt	
				SZÁZAS:	foglalt	
				EZRES:	foglalt	
				TÍZEZRES:	Technológia skálázás ciklus maszkokon belül (Oldal 818)	
					0 =	beadás: teljes
					1 =	beadás: egyszerű

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
9		<_AMODE>	INT	alternatív módus		
				EGYES:	furatmélység Z1 (absz/növ)	
					0 =	kompatibilitás a DP/DPR programozásból
					1 =	növekményes
					2 =	abszolút
				TÍZES:	várakozási idő DT a végső furatmélységen másodpercekben/fordulatokban	
					0 =	kompatibilitás, a DT előjeléből (> 0 másodpercek / < 0 fordulatok)
					1 =	másodpercekben
					2 =	fordulatokban
				SZÁZAS:	felfúrás mélység ZA absz/növ	
					0 =	növekményes
					1 =	abszolút
				EZRES:	felfúrás előtolás értéke	
					0 =	pályaelőtolás %-ában
					1 =	F/prec
					2 =	F/ford
				TÍZEZRES:	maradék furatmélység ZD absz/növ	
					0 =	növekményes
					1 =	abszolút
				SZÁZEZRES:	maradék furat előtolás értéke	
					0 =	pályaelőtolás %-ában
					1 =	F/prec
					2 =	F/ford
10		<_VARI>	INT	felfúrás/átfúrás megmunkálási mód		
				EGYES:	foglalt	
				TÍZES:	foglalt	
				SZÁZAS:	foglalt	
				EZRES:	átfúrás	
					0 =	átfúrás "nem"
					1 =	átfúrás "igen"
				TÍZEZRES:	felfúrni	
					0 =	felfúrás "nem"
					1 =	felfúrás "igen"
11	ZA	<S_ZA>	REAL	felfúrás mélység növekményesen vonatkoztatva a vonatkoztatási pontra vagy abszolút (lásd <_AMODE> SZÁZAS)		
12	FA	<S_FA>	REAL	felfúrás előtolás értékként vagy %-ban (összeköttetésben a <_AMODE>-vel EZRES)		

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés
13	ZD	<S_ZD>	REAL	maradék furat mélység növekményesen vonatkoztatva a vonatkoztatási pontra vagy abszolút (lásd <_AMODE> TIZEZRES)
14	FD	<S_FD>	REAL	maradék furat előtolás értékként vagy %-ban (összeköttetésben a <_AMODE>-vel SZÁZEZRES)

20.1.23 CYCLE83 - Mély-lyuk fúrás

Szintaxis

```
CYCLE83(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <FDEP>, <FDPR>, <_DAM>,
<DTB>, <DTS>, <FRF>, <VARI>, <_AXN>, <_MDEP>, <_VRT>, <_DTD>,
<_DIS1>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
1	RP	<RTP>	REAL	visszahúzási sík (absz.)		
2	Z0	<RFP>	REAL	vonatkoztatási pont (absz.)		
3	SC	<SDIS>	REAL	biztonsági távolság (vonatkoztatási ponthoz adódik, előjel nélkül beadni)		
4	Z1	<DP>	REAL	végső furatmélység (növ), lásd <_AMODE>		
5	Z1	<DPR>	REAL	végső furatmélység (növ), lásd <_AMODE>		
6	D	<FDEP>	REAL	1. furatmélység (absz), lásd <_AMODE>		
7	D	<FDPR>	REAL	1. furatmélység (növ), lásd <_AMODE>		
8	DF	<_DAM>	REAL	csökkenés érték/százalék minden további fogáshoz, lásd <_AMODE>		
9	DTB	<DTB>	REAL	várakozási idő furatmélységen, lásd <_AMODE>		
10	DTS	<DTS>	REAL	várakozási érték kezdőpontra (csak leforgácsolásnál), lásd <_AMODE>		
11	FD1	<FRF>	REAL	előtolás százalék az első fogásnál, lásd <_AMODE>		
12		<VARI>	INT	Megmunkálási mód		
				EGYES:	forgácstörés / forgácstalanítás	
					0 =	forgácstörés
					1 =	forgácstalanítás
13		<_AXN>	INT	szerszám-tengely		
					0 =	3. geometriai tengely
					1 =	1. geometriai tengely
					2 =	2. geometriai tengely
					> 2	3. geometriai tengely
14	V1	<_MDEP>	REAL	minimális fogás (csak degressziónál százalékban)		
15	V2	<_VRT>	REAL	visszahúzási érték minden megmunkálás után (csak forgácstörésnél)		
					> 0	változó visszahúzási érték
					0 =	Alapérték 1 mm

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
16	DT	<_DTD>	REAL	várakozási idő végső furatmélységen, lásd <_AMODE>		
17	V3	<_DIS1>	REAL	elő-leállás távolság (csak forgácstalanításnál), lásd <_AMODE>		
18		<_GMODE>	INT	geometria módus (programozott geometria-értékek értékelése)		
				EGYES:	foglalt	
				TÍZES:	furatmélység csúcsra/szárra vonatkoztatva	
					0 =	csúcs
					1 =	szár
19		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/G18/G19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
				TÍZES:	---	foglalt
				SZÁZAS:	---	foglalt
				EZRES:	---	foglalt
				TÍZEZRES:	Technológia skálázás ciklus maszkokon belül (Oldal 818)	
					0 =	beadás: teljes
					1 =	beadás: egyszerű

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
20		<_AMODE>	INT	alternatív módus		
				EGYES:	furatmélység = végső furatmélység Z1 (absz/növ)	
					0 =	kompatibilitás a <DP>/<DPR> programozásból
					1 =	növekményes
					2 =	abszolút
				TÍZES:	várakozási idő a DTB végső furatmélységen másodpercekben/fordulatokban	
					0 =	kompatibilitás a DTB előjeléből (> 0 másodpercek vagy < 0 fordulatok)
					1 =	másodpercekben
					2 =	fordulatokban
				SZÁZAS:	várakozási idő DTS kezdőponton másodpercekben/fordulatokban	
					0 =	kompatibilitás a DTS előjeléből (> 0 másodpercek vagy < 0 fordulatok)
					1 =	másodpercekben
					2 =	fordulatokban
				EZRES:	várakozási idő a DT végső furatmélységen másodpercekben/fordulatokban	
					0 =	kompatibilitás a DTS előjeléből (> 0 másodpercek vagy < 0 fordulatok)
					1 =	másodpercekben
					2 =	fordulatokban
				TÍZEZRES:	1. furatmélység D (absz/növ)	
					0 =	kompatibilitás a <FDEPF>/<DPR> programozásból
					1 =	növekményes
					2 =	abszolút
				SZÁZEZRES:	csökkenés érték/százalék <_DAM> minden további fogáshoz	
					0 =	kompatibilitás, a <_DAM> előjeléből (> 0 csökkenés érték vagy < 0 0,001 ... 1,0 tényező)
					1 =	csökkenés érték
					2 =	százalék (0,001 ... 100 %)
				EGYMILLIÓS:	elő-leállítás távolság V3 automatikus/kézi	
					0 =	kompatibilitás a <_DIS1> előjeléből (> 0 automatikus vagy < 0 kézi)

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
					1 =	automatikus (a ciklusban lesz kiszámítva)
					2 =	kézi (programozott érték)
				TÍZMILLIÓS:	első fogásvétel előtolási tényező <FRF> tényező/százalék	
					0 =	kompatibilitás, tényezőként (0,001 ... 1,0, FRF = 0 jelentése 100%)
					1 =	százalék (0,001 ... 999,999 %)

20.1.24 CYCLE84 - Menetfúrás kiegyenlítő tokmány nélkül

Szintaxis

```
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>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
1	RP	<RTP>	REAL	visszahúzási sík (absz.)		
2	Z0	<RFP>	REAL	vonatkoztatási pont (absz.)		
3	SC	<SDIS>	REAL	biztonsági távolság (vonatkoztatási ponthoz adódik, előjel nélkül beadni)		
4	Z1	<DP>	REAL	furatmélység = végső furatmélység (absz), lásd <_AMODE>		
5	Z1	<DPR>	REAL	furatmélység = végső furatmélység (növ), lásd <_AMODE>		
6	DT	<DTB>	REAL	várakozási idő a furatmélységen másodpercben		
7	SDE	<SDAC>	INT	forgásirány a ciklus vége után		
8		<MPIT>	REAL	menetméret csak "ISO metrikus" számára (emelkedés futás közben belül lesz kiszámítva)		
9	P	<PIT>	REAL	menetemelkedés értékként, mértékegység lásd <_PITA>		
10	αS[1)]	<POSS>	REAL	orsópozíció tájolt orsó állj-hoz		
11	S	<SST>	REAL	orsó-fordulatszám menetfúráshoz		
12	SR	<SST1>	REAL	orsó-fordulatszám visszahúzáshoz		
13		<_AXN>	INT	fúró-tengely	0 =	3. geometria-tengely
					1 =	1. geometria-tengely
					2 =	2. geometria-tengely
					≥ 3 =	3. geometria-tengely

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés			
14		<_PITA>	INT	menetemelkedés mértékegysége (kiértékelés <PIT> és <MPIT>)			
					0 =	emelkedés mm-ben	- kiértékelés <MPIT>/<PIT>
					1 =	emelkedés mm-ben	- kiértékelés <PIT>
					2 =	emelkedés TPI-ben	- kiértékelés <PIT> (menetszám hüvelyenként)
					3 =	emelkedés hüvelykben	- kiértékelés <PIT>
					4 =	MODUL	- kiértékelés <PIT>
15		<_TECHNO>	INT	technológia[1)]			
				EGYES:	pontos-állj viselkedés		
					0 =	pontos állj viselkedés mint a ciklus felhívása előtt aktív	
					1 =	pontos állj G601	
					2 =	pontos állj G602	
					3 =	pontos állj G603	
				TÍZES:	elővezérlés		
					0 =	elővezérléssel/anélkül mint a ciklus felhívása előtt aktív	
					1 =	elővezérléssel FFWON	
					2 =	elővezérlés nélkül FFWOF	
				SZÁZAS:	gyorsulás		
					0 =	SOFT/BRISK/DRIVE mint a ciklus felhívása előtt aktív	
					1 =	rándulás-határolással SOFT	
					2 =	rándulás-határolás nélkül BRISK	
					3 =	csökkentett gyorsulás DRIVE	
				EZRES:	MCALL orsóüzem		
					0 =	MCALL-nál orsóüzemet ismét aktiválni	
					1 =	MCALL-nál helyzetszabályozásban maradni	
16		<_VARI>	INT	megmunkálási mód			
				EGYES:	0 =	1 vágás	
					1 =	forgácstörés (mély-lyuk menetfúrás)	
					2 =	forgácstalanítás (mély-lyuk menetfúrás)	
				EZRES:	ISO/SIEMENS módus a beadási maszkra nem releváns		
					0 =	felhívás ISO kompatibilitásból	
					1 =	felhívás SIEMENS kontextusból	
17	D	<_DAM>	REAL	maximális fogásmélység (csak forgácstalanításnál/forgácstörésnél)			
18	V2	<_VRT>	REAL	visszahúzási érték minden megmunkálás után (csak forgácstörésnél), lásd <_AMODE>			

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
19		<_PITM>	STRING[15]	string menetemelkedési beadás jelölőjeként[2)]		
20		<_PTAB>	STRING[5]	string menettáblázathoz ("", "ISO", "BSW", "BSP", "UNC")[2)]		
21		<_PTABA>	STRING[20]	string kiválasztáshoz a menettáblázatban (pl. "M 10", "M 12", ...)[2)]		
22		<_GMODE>	INT	geometria módus (programozott geometria-értékek értékelése)		
				EGYES:	foglalt	
				TÍZES:	foglalt	
23		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/G18/G19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
				TÍZES:	foglalt	
				SZÁZAS:	foglalt	
				EZRES:	kompatibilitás módus (csak a beadási maszknak a visszafordításhoz), ha MD 52216 Bit0 = 1[1)]	
					0 =	technológia paraméterek ki lesznek jelezve (kompatibilitás): TECHNO-paraméterek hatnak
					1 =	technológia paraméterek nem lesznek jelezve: technológia "mint a ciklusfelhívás előtt" hat
				TÍZEZRES:	Technológia skálázás ciklus maszkokon belül (Oldal 818)	
					0 =	beadás: teljes
					1 =	beadás: egyszerű

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
24		<_AMODE>	INT	alternatív módus		
				EGYES:	furatmélység = végső furatmélység Z1 (absz/ növ)	
					0 =	kompatibilitás a <DP>/<DPR> programozásból
					1 =	növekményes
					2 =	abszolút
				TÍZES:	foglalt	
				SZÁZAS:	foglalt	
				EZRES:	menet forgásirány jobb/bal	
					0 =	kompatibilitás, a PIT/MPIT előjelből
					1 =	jobb
					2 =	bal
				TÍZEZRES:	foglalt	
				SZÁZEZRES:	foglalt	
				EGYMILLIÓS:	visszahúzási érték minden megmunkálás után V2 kézi/automatikus	
					0 =	kompatibilitás, <_VRT> programozásából (> 0 változó érték vagy ≤ 0 1 mm/ 0.0394 hüvelyk alapérték)
					1 =	automatikus (alapérték 1 mm/ 0,0394 hüv)
					2 =	kézi (mint V2-nél programozva)

[1] Technológia mezők lehetnek kihagyva az SD52216 $MCS_FUNCTION_MASK_DRILL beállítási adattól függően

[2] A 19, 20, és 21 paraméterek csak a beadási maszk menettáblázatában a menet kiválasztásnál lesznek használva. A hozzáférés a menettáblázatokhoz ciklus-definícióval a ciklus futásakor nem lehetséges.

20.1.25 CYCLE85 - dörzsölés

Szintaxis

```
CYCLE85(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <DTB>, <FFR>, <RFF>,
<_GMODE>, <_DMODE>, <_AMODE>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés
1	RP	<RTP>	REAL	visszahúzási sík (absz.)
2	Z0	<RFP>	REAL	vonatkoztatási pont (absz.)
3	SC	<SDIS>	REAL	biztonsági távolság (vonatkoztatási ponthoz adódik, előjel nélkül beadni)
4	Z1	<DP>	REAL	furatmélység (absz), lásd <_AMODE>

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
5	Z1	<DPR>	REAL	furatmélység (növ), lásd <_AMODE>		
6	DT	<DTB>	REAL	várakozási idő végső furatmélységen, lásd <_AMODE>		
7	F	<FFR>	REAL	Előtolás		
8	FR	<RFF>	REAL	előtolás visszahúzásnál		
9		<_GMODE>	INT	foglalt		
10		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/G18/G19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
11		<_AMODE>	INT	alternatív módus (fúrás)		
				EGYES:	furatmélység Z1 (absz/növ)	
					0 =	kompatibilitás a DP/DPR programozásból
					1 =	növekményes
					2 =	abszolút
				TÍZES:	várakozási idő DT a végső furatmélységen másodpercekben/fordulatokban	
					0 =	kompatibilitás, a DT előjeléből (> 0 másodpercek vagy < 0 fordulatok)
					1 =	másodpercekben
					2 =	fordulatokban

20.1.26 CYCLE86 - Kiesztergálás

Szintaxis

```
CYCLE86(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <DTB>, <SDIR>, <RPA>,
<RPO>, <RPAP>, <POSS>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés
1	RP	<RTP>	REAL	visszahúzási sík (absz.)
2	Z0	<RFP>	REAL	vonatkoztatási pont (absz.)
3	SC	<SDIS>	REAL	biztonsági távolság (vonatkoztatási ponthoz adódik, előjel nélkül beadni)
4	Z1	<DP>	REAL	furatmélység (absz), lásd <_AMODE>

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
5	Z1	<DPR>	REAL	furatmélység (növ), lásd <_AMODE>		
6	DT	<DTB>	REAL	várakozási idő végső furatmélységen, lásd <_AMODE>		
7	DIR	<SDIR>	INT	orsó-forgásirány	3 =	M3
					4 =	M4
8	DX	<RPA>	REAL	leemelés érték X irányban		
9	DY	<RPO>	REAL	leemelés érték Y irányban		
10	DZ	<RPAP>	REAL	leemelés érték Z irányban		
11	SPOS	<POSS>	REAL	orsópozíció leemeléshez (tájolható orsó állj-hoz, fokban)		
12		<_GMODE>	INT	geometria módus (programozott geometria-értékek értékelése)		
				EGYES:	Leemelés modus	
					0 =	leemelés, kompatibilitás
					1 =	nem leemelni
13		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/G18/G19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
14		<_AMODE>	INT	alternatív módus		
				EGYES:	furatmélység Z1 (absz/növ)	
					0 =	kompatibilitás a <DP>/<DPR> programozásból
					1 =	növekményes
					2 =	abszolút
				TÍZES:	várakozási idő a DT végső furatmélységen másodpercekben/fordulatokban	
					0 =	kompatibilitás, a DT előjeléből (> 0 másodpercek vagy < 0 fordulatok)
					1 =	másodpercekben
					2 =	fordulatokban

20.1.27 CYCLE92 - Leszúrás

Szintaxis

```
CYCLE92(<_SPD>, <_SPL>, <_DIAG1>, <_DIAG2>, <_RC>, <_SDIS>, <_SV1>,
<_SV2>, <_SDAC>, <_FF1>, <_FF2>, <_SS2>, <_DIAGM>, <_VARI>, <_DN>,
<_DMODE>, <_AMODE>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
1	X0	<_SPD>	REAL	vonatkoztatási pont (absz. mindig átmérő)		
2	Y0	<_SPL>	REAL	vonatkoztatási pont (absz.)		
3	X1	<_DIAG1>	REAL	fordulatszám-csökkentés mélysége, lásd <_AMODE> (EGYES)		
4	X2	<_DIAG2>	REAL	végső mélység, lásd <_AMODE> (TÍZES)		
5	R/FS	<_RC>	REAL	lekerekítés-sugár vagy letörés-szélesség, lásd <_AMODE> (EZRES)		
6	SC	<_SDIS>	REAL	biztonsági távolság (vonatkoztatási ponthoz adódik, előjel nélkül beadni)		
7	S	<_SV1>	REAL	állandó orsó-fordulatszám, lásd <_AMODE> (TÍZEZRES)		
	V			állandó vágósebesség		
8	SV	<_SV2>	REAL	maximális fordulatszám állandó vágósebességnél		
9	DIR	<_SDAC>	INT	orsó-forgásirány	3 =	M3-hoz
					4 =	M4-hez
10	F	<_FF1>	REAL	előtolás a mélységig a fordulatszám-csökkentéshez		
11	FR	<_FF2>	REAL	csökkentett előtolás a végső mélységig		
12	SR	<_SS2>	REAL	csökkentett fordulatszám a végső mélységig		
13	XM	<_DIAGM>	REAL	darab-megfogó mélységét megtenni (absz. mindig átmérő)		
14		<_VARI>	INT	Megmunkálási mód		
				EGYES:	Visszahúzás	
					0 =	visszahúzás <_SPD> + <_SDIS>-ra
					1 =	nincs visszahúzás a végén
				TÍZES:	darab-megfogó	
					0 =	nem, M utasítást nem végrehajtani
					1 =	igen, CUST_TECHCYC(101) felhívása - fiókot kinyitni, CUST_TECHCYC(102) - fiókot becsukni
15		<_DN>	INT	D-szám a 2. vágóélhez, ha nincs programozva ⇒ D+1		
20		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/G18/G19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
21		<_AMODE>	INT	alternatív módus		
				EGYES:	mélység a fordulatszám-csökkentéshez (<_DIAG1>)	
					0 =	abszolút, sík-tengely értéke átmérőben
					1 =	növekményes, sík-tengely értéke sugárban
				TÍZES:	végső mélység (<_DIAG2>)	
					0 =	abszolút, sík-tengely értéke átmérőben
					1 =	növekményes, sík-tengely értéke sugárban
				SZÁZAS:	foglalt	
				EZRES:	sugár/letörés (<_RC>)	
					0 =	sugár
					1 =	letörés
				TÍZEZRES:	orsó-fordulatszám/vágósebesség (<_SV1>)	
					0 =	állandó orsó-fordulatszám
					1 =	állandó vágósebesség

20.1.28 CYCLE95 - Kontúr leforgácsolás

Szintaxis

```
CYCLE95(<NPP>, <MID>, <FALZ>, <FALX>, <FAL>, <FF1>, <FF2>, <FF3>,
<_VARI>, <DT>, <DAM>, <_VRT>, <_GMODE>, <_DMODE>)
```

Paraméterek

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés
1	CON	<NPP>	STRING [140]	kontúrnév
2	D	<MID>	REAL	maximális mélységi fogás nagyolásnál, lásd <_GMODE>
3	UZ	<FALZ>	REAL	simítási ráhagyás Z-ben
4	UX	<FALX>	REAL	simítási ráhagyás X-ben
5	U	<FAL>	REAL	simítási ráhagyáskontúr-párhuzamosan (mindkét tengelyen hat)
6	F	<FF1>	REAL	előtolás nagyoláshoz
7	FY	<FF2>	REAL	hátravágások bemerülési előtolás
8	FS	<FF3>	REAL	simító előtolás

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
9		<_VARI>	INT	megmunkálási mód		
				EGYES vagy TIZES:		
					1 =	nagyolás, hosszban, kívül
					2 =	nagyolás, síkban, kívül
					3 =	nagyolás, hosszban, belül
					4 =	nagyolás, síkban, belül
					5 =	simítás, hosszban, kívül
					6 =	simítás, síkban, kívül
					7 =	simítás, hosszban, belül
					8 =	simítás, síkban, belül
					9 =	teljes megmunkálás, hosszban, kívül
					10 =	teljes megmunkálás, síkban, kívül
					11 =	teljes megmunkálás, hosszban, belül
					12 =	teljes megmunkálás, síkban, belül
				SZÁZAS:		
					0 =	utánhúzással a kontúron, maradéksarkok nélkül
					1 =	utánhúzás nélkül a kontúron
					2 =	utánhúzás csak az előző metszéspontig, keletkezhetnek maradéksarkok
10	DT	<DT>	REAL	várakozási idő előtolás megszakításnál		
11	DI	<DAM>	REAL	előtolás megszakítások távolsága		
12	VRT	<_VRT>	REAL	leemelési út a kontúrról		
					0 =	belül egy 1 mm-es leemelési út lesz használva a hüvelyk/metrikus méretrendszertől függetlenül
					> 0 =	leemelési út
13		<_GMODE>	INT	geometria módus (programozott geometria-értékek értékelése)		
				EGYES:	fogásmélység értékelése	
					0 =	fogásmélység a DIAMON/DIAMOF D-csoportnak megfelelően beszámítva
					1 =	fogásmélység sugárértékként hat (DIAMON/DIAMOF-tól függetlenül)

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
14		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/G18/G19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
				EZRES:		
					0 =	kompatibilitás módus: kontúrnév az NPP-ben
					1 =	kontúrnév a CYCLE62-ben lesz programozva és a _SC_CONT_NAME-ben átadva

20.1.29 CYCLE98 - Menetlánc

Szintaxis

```
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>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés
1	Z0	<_PO1>	REAL	vonatkoztatási pont Z-ben (absz)
2	X0	<_DM1>	REAL	vonatkoztatási pont X-ben (absz), átmérőben
3	Z1	<_PO2>	REAL	közbenső pont 1 Z-ben (absz/növ), lásd <_AMODE> (EGYES)
4	X1	<_DM2>	REAL	közbenső pont 1 X-ben (absz/növ), lásd <_AMODE> (TIZES) vagy
	X1α			menet-ferdeség 1 (-90° ... 90°) absz mindig átmérő, növ mindig sugár
5	Z2	<_PO3>	REAL	közbenső pont 2 Z-ben (absz/növ), lásd <_AMODE> (SZÁZAS)
6	X2	<_DM3>	REAL	közbenső pont 2 X-ben (absz/növ), lásd <_AMODE> (EZRES) vagy
	X2α			menet-ferdeség 2 (-90° ... 90°) absz mindig átmérő, növ mindig sugár
7	Z3	<_PO4>	REAL	végpont Z-ben (absz/növ), lásd <_AMODE> (TÍZEZRES)
8	X3	<_DM4>	REAL	végpont X-ben (absz/növ), lásd <_AMODE> (SZÁZEZRES) vagy
	X3α			menet-ferdeség 3 (-90° ... 90°) absz mindig átmérő, növ mindig sugár
9	LW	<APP>	REAL	menet előfutás (növ, előjel nélkül beadni)

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
10	LR	<ROP>	REAL	menet kifutás (növ, előjel nélkül beadni)		
11	H1	<TDEP>	REAL	menet mélység (növ, előjel nélkül beadni)		
12	U	<FAL>	REAL	simítási ráhagyás X-ben és Z-ben		
13	DP	<_IANG>	REAL	fogás-ferdeség távolságként vagy szögként, lásd <_AMODE> (MILLIÓS)		
	αP			A fogás-ferdeség a <_VARI> (SZÁZEZRES) paraméter beállításának megfelelően hat.		
					Definíció a <_VARI_SZÁZAS = 0 kompatibilitás módra:	
					> 0 =	él fogásvétel egy élen
					0 =	fogás merőlegesen a menetben
					< 0 =	él fogásvétel váltakozó éleken
					definíció _VARI_SZÁZAS<>0-ra:	
					> 0 =	fogás a pozitív élen
					0 =	fogás középen
					< 0 =	fogás a negatív élen
14	α0	<NSP>	REAL	kezdőpont eltolás az 1. menetjáratra		
15		<NRC>	INT	nagyoló fogások száma, lásd <_VARI> (TÍZEZRES)		
16	NN	<NID>	INT	üres vágások száma		
17	P0	<_PP1>	REAL	menetemelkedés 1. menetszakasz, lásd "<_PITA>"		
18	P1	<_PP2>	REAL	menetemelkedés 2. menetszakasz, lásd "<_PITA>"		
19	P2	<_PP3>	REAL	menetemelkedés 3. menetszakasz, lásd "<_PITA>"		

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
20		<_VARI>	INT	Megmunkálás		
				EGYES:	technológia	
					1 =	külső menet lineáris fogással
					2 =	belső menet lineáris fogással
					3 =	külső menet degresszív fogás-sal, forgács-keresztmetszet ál-landó marad
					4 =	belső menet degresszív fogás-sal, forgács-keresztmetszet ál-landó marad
				TÍZES:	foglalt	
				SZÁZAS:	fogásvétel mód	
					0 =	kompatibilitás módus <_IANG>-hez
					1 =	egyoldalas fogásvétel
					2 =	váltakozó fogásvétel
				EZRES:	foglalt	
				TÍZEZRES:	alternatív mélységi fogásvétel	
					0 =	kompatibilitás, nagyoló fogások számának elő-megadása (<_NRC>)
					1 =	érték megadása 1. fogásra (<_MID>)
				SZÁZEZRES:	Megmunkálási mód	
					0 =	Kompatibilitás (nagyolás és simí-tás)
					1 =	nagyolás
					2 =	simítás
					3 =	nagyolás és simítás
				EGYMILLIÓS:	megmunkálási sorrend több-bekezdésű menetnél	
					0 =	bekezdések sorrendje növekvő
					1 =	bekezdések sorrendje szemben
21	N	<_NUMTH>	INT	menet-bekezdések száma		
22		<_VRT>	REAL	visszafutási távolság (növ.)		
					0 =	belül egy 1 mm-es leemelési út lesz használva a hüvelyk/metri-kus méretrendszertől függetlenül
					> 0 =	leemelési út
23	D1	<_MID>	REAL	első fogás, lásd <_VARI> (TÍZEZRES)		
24	DA	<_GDEP>	REAL	járatváltás mélység (csak "több-bekezdésű"-nél hat)		
					0 =	járatváltás mélységet nem figye-lembe venni
					> 0 =	járatváltás mélységet figyelembe venni

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
25		<_IFLANK>	REAL	fogás-ferdeség szélességként (csak a felülethez)		
26		<_PITA>	INT	menetemelkedés értékelése		
					0 =	kompatibilitás módus menetemelkedés, <_PP1> ... <_PP3> kiértékelés mint eddig, az aktív metrikus/hüvelyk rendszernek megfelelően
					1 =	emelkedés mm-ben
					2 =	emelkedés TPI-ben (menetjáratok hüvelykenként)
					3 =	emelkedés hüvelykben
					4 =	MODUL
27		<_PITM1>	STRING[15]	string menetemelkedési beadás jelölőjeként (csak a felülethez)		
28		<_PITM2>	STRING[15]	string menetemelkedési beadás jelölőjeként (csak a felülethez)		
29		<_PITM3>	STRING[15]	string menetemelkedési beadás jelölőjeként (csak a felülethez)		
30		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/G18/G19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
				TÍZES:	---	foglalt
				SZÁZAS:	---	foglalt
				EZRES:	---	foglalt
				TÍZEZRES:	Technológia skálázás ciklus maszkokon belül (Oldal 818)	
					0 =	beadás: teljes
					1 =	beadás: egyszerű

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
31		<_AMODE>	INT	alternatív módus		
				EGYES:	1. közbenső pont Z-ben (Z1)	
					0 =	abszolút
					1 =	növekményes
				TÍZES:	1. közbenső pont X-ben (X1)	
					0 =	abszolút
					1 =	növekményes
					2 =	α
				SZÁZAS:	2. közbenső pont Z-ben (Z2)	
					0 =	abszolút
					1 =	növekményes
				EZRES:	2. közbenső pont X-ben (X2)	
					0 =	abszolút
					1 =	növekményes
					2 =	α
				TÍZEZRES:	végpont Z-ben (Z3)	
					0 =	abszolút
					1 =	növekményes
				SZÁZEZRES:	végpont X-ben (X3)	
					0 =	abszolút
					1 =	növekményes
					2 =	α
				EGYMILLIÓS:	fogás ferdeség választása szögként vagy szélességként	
					0 =	fogásszög <_IANG>
					1 =	fogás ferdeség<_IFLANK>
				TÍZMILLIÓS:	egy-bekezdésű/több-bekezdésű	
					0 =	kompatibilitás mód (<_NSP> kezdőszög ki lesz értékelve)
					1 =	egy-bekezdésű (kezdőszög eltolással <_NSP>)
					2 =	több-bekezdésű

20.1.30 CYCLE99 - Menetesztergálás

Szintaxis

```
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>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
1	Z0	<_SPL>	REAL	vonatkoztatási pont (absz.)		
2	X0	<_SPD>	REAL	vonatkoztatási pont (absz. mindig átmérő)		
3	Z1	<_FPL>	REAL	végpont, <_AMODE> (EGYES)-sel kapcsolatban		
4	X1	<_FPD>	REAL	végpont, <_AMODE> (TÍZES)-sel kapcsolatban		
5	LW/LW2	<_APP>	REAL	menet-előfutás <_AMODE>-val (SZÁZAS) kapcsolatban vagy menet befutás = menet kifutás <_AMODE> (SZÁZAS)-sal kapcsolatban		
6	LR	<_ROP>	REAL	menet-kifutás		
7	H1	<_TDEP>	REAL	menet-mélység		
8	U	<_FAL>	REAL	simítási ráhagyás X-ben és Z-ben		
9	DP	<_IANG>	REAL	fogás-ferdeség távolságként vagy szögként, <_AMODE> (EZRES)-sel kapcsolatban		
	αP				> 0 =	fogás a pozitív élen
					< 0 =	fogás a negatív élen
					0 =	fogás középen
10	α0	<_NSP>	REAL	kezdőszög eltolás (csak "egy-menetes"-nél hat)		
11	ND	<_NRC>	INT	nagyolási fogások száma, <_VARI> (TÍZEZRES)-sel kapcsolatban		
12	NN	<_NID>	INT	üres vágások száma		
13	P	<_PIT>	REAL	menetemelkedés értékként, <_PITA>-val kapcsolatban		

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
14		<_VARI>	INT	Megmunkálási mód		
				EGYES:	technológia	
					1 =	külső menet lineáris fogással
					2 =	belső menet lineáris fogással
					3 =	külső menet degresszív fogás-sal, forgács-keresztmetszet ál-landó marad
					4 =	belső menet degresszív fogás-sal, forgács-keresztmetszet ál-landó marad
				TÍZES:	foglalt	
				SZÁZAS:	fogásvétel mód	
					1 =	egyoldalas fogásvétel
					2 =	váltakozó fogásvétel
				EZRES:	foglalt	
				TÍZEZRES:	alternatív mélységi fogásvétel	
					0 =	nagyoló fogások számának elő-megadása (<_NRC>)
					1 =	érték megadása 1. fogásra (<_MID>)
				SZÁZEZRES:	Megmunkálási mód	
					1 =	nagyolás
					2 =	simítás
					3 =	nagyolás és simítás
				EGYMILLIÓS:	megmunkálási sorrend több-bekezdésű menetnél	
					0 =	bekezdések sorrendje növekvő
					1 =	bekezdések sorrendje szemben
15	N	<_NUMTH>	INT	menet-bekezdések száma		
16	VR	<_SDIS>	REAL	visszafutási távolság, növ		
17	D1	<_MID>	REAL	első fogás-mélység, <_VARI> (TÍZEZRES)-sel kapcsolatban		
18	DA	<_GDEP>	REAL	járatváltás mélység (csak "több-bekezdésű"-nél hat)		
					0 =	járatváltás mélységet nem figye-lembe venni
					> 0 =	járatváltás mélységet figyelembe venni
19	G	<_PIT1>	REAL	emelkedés változása fordulatonként		
					0 =	menetemelkedés állandó (G33)
					> 0 =	menetemelkedés nagyobb lesz (G34)
					< 0 =	menetemelkedés kisebb lesz (G35)
20		<_FDEP>	REAL	bemerülési mélység (előjel nélkül beadni)		
21	N1	<_GST>	INT	kezdő menet N1 = 1...N, <_AMODE> (SZÁZEZRES)-sel kapcsolatban		

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
22		<_GUD>	INT	foglalt		
23		<_IFLANK>	REAL	fogás-ferdeség szélességként (csak a felülethez)		
24		<_PITA>	INT	menetemelkedés mértékegysége (PIT és/vagy MPIT kiértékelés)		
					0 =	emelkedés mm-ben - MPIT/PIT kiértékelés
					1 =	emelkedés mm-ben - PIT kiértékelés
					2 =	emelkedés TPI-ben - PIT kiértékelés (menetszám hüvelykenként)
					3 =	emelkedés hüvelykben - PIT kiértékelés
					4 =	MODUL - PIT kiértékelés
25		<_PITM>	STRING[15]	string menetemelkedési beadás jelölőjeként (csak a felülethez)[1)]		
26		<_PTAB>	STRING[20]	string menet-táblázathoz(csak a felülethez)[1)]		
27		<_PTABA>	STRING[20]	string kiválasztáshoz menet-táblázaban (csak a felülethez)[1)]		
28		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/G18/G19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
				TÍZES:	menet fajta	
					0 =	hossz-menet
					1 =	síkmenet
					2 =	kúpmenet
				SZÁZAS:	---	foglalt
				EZRES:	---	foglalt
				TÍZEZRES:	Technológia skálázás ciklus maszkokon belül (Oldal 818)	
					0 =	beadás: teljes
					1 =	beadás: egyszerű

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
29		<_AMODE>	INT	alternatív módus		
				EGYES:	menethossz Z-ben	
					0 =	abszolút
					1 =	növekményes
				TÍZES:	menethossz X-ben	
					0 =	abszolút, sík-tengely értéke átmérőben
					1 =	növekményes, sík-tengely értéke sugárban
					2 =	α
				SZÁZAS:	előfutás/befutás-út kiértékelés <_APP>	
					0 =	menet-előfutás <_APP>
					1 =	menet-befutás = menet-kifutás <_APP> = -<_ROP>
					2 =	menet bekezdési utat megadni <_APP> = -<_APP>
				EZRES:	fogás ferdeség választása szögként vagy szélességként	
					0 =	fogásszög <_IANG>
					1 =	fogás ferdeség<_IFLANK>
				TÍZEZRES:	egy-bekezdésű/több-bekezdésű	
					0 =	egy-bekezdésű (kezdőszög eltolással <_NSP>)
					1 =	több-bekezdésű
				SZÁZEZRES:	kezdőmenet <_GST>	
					0 =	teljes megmunkálás
					1 =	ettől a járattól indítani a megmunkálást
					2 =	csak ezt a járatot megmunkálni
				EGYMILLIÓS:	belógás-kompenzáció hossz-menetek-nél	
					0 =	íves menet szegmens magasság XS
					1 =	íves menet sugár RS
30	XS / RS	<_S_XRS>	REAL	belógás-kompenzáció hossz-meneteknél <_AMODE>-val kapcsolatban EGYMILLIÓS		

Megjegyzés

[1)] A <_PITM>, <_PTAB> és <_PTABA> paraméterek csak a beadási maszk menettáblázatában a menet kiválasztásnál lesznek használva.
A hozzáférés a menettáblázatokhoz ciklus-definícióval a ciklus futásakor nem lehetséges.

20.1.31 CYCLE435 - Lehúzó koordináták beállítása

Szintaxis

```
CYCLE435(<_T>, <_DD>, <S_TA>, <S_DA>, <S_AD>, <S_AL>, <S_PVD>,
<S_PVL>, <S_PD>, <S_PL>, <_AMODE>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
1		<_T>	STRING[32]	köszörűtárcsa szerszámnév		
2		<_DD>	INT	köszörűtárcsa vágóél-szám		
3		<S_TA>	STRING[32]	lehúzó vonatkoztatási pont - lehúzó neve		
4		<S_DA>	INT	lehúzó vágóél-száma		
5		<S_AD>	REAL	lehúzó érték átmérő		
6		<S_AL>	REAL	lehúzó érték sík		
7		<S_PVD>	REAL	profilirozó átmérő		
8		<S_PVL>	REAL	profilirozó eltolás sík		
9		<S_PD>	REAL	profilozás ráhagyás átmérő		
10		<S_PL>	REAL	profilozás ráhagyás sík		
11		<_AMODE>	INT	alternatív módus		
				EGYES:	aktív szerszám ciklus végén	
					0 =	lehúzó aktív
					1 =	tárcsa aktív

20.1.32 CYCLE495 - Profilozás

Szintaxis

```
CYCLE495(<_T>, <_DD>, <_SC>, <_F>, <_VARI>, <_D>, <_DX>, <_DZ>,
<S_PA>, <S_N>, <_DMODE>, <_AMODE>, <S_FW>, <S_HW>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés
1		<_T>	STRING[20]	köszörűtárcsa szerszámnév
2		<_DD>	INT	köszörűtárcsa vágóél-szám
3		<_SC>	REAL	biztonsági távolság az akadályok megkerüléséhez, növekményes
4		<_F>	REAL	profilozás előtolás

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
5		<_VARI>	INT	megmunkálási mód		
				EGYES:	profilozás mód	
					1 =	tengely-párhuzamosan
					2 =	kontúr-párhuzamosan
				TÍZES:	megmunkálási irány	
					0 =	húzva 1...4 vágóélhelyzettel lehetséges
					1 =	lökve 1...4 vágóélhelyzettel lehetséges
					2 =	váltakozva 1...8 vágóélhelyzettel lehetséges
					3 =	kezdet → vég 1...8 vágóélhelyzettel lehetséges
					4 =	vég → kezdet 1...8 vágóélhelyzettel lehetséges
				SZÁZAS:	fogásirány	
					1 =	fogásvétel X- G18-nál ill. Y- G19-nél
					2 =	fogásvétel X+ G18-nál ill. Y+ G19-nél
					3 =	fogásvétel Z- G18-nál és G19-nél
					4 =	fogásvétel Z+ G18-nál és G19-nél
6		<_D>	REAL	lehúzási érték tengely-párhuzamos profilírozás módnál		
7		<_DX>	REAL	lehúzási érték X G18-nál ill. Y G19-nél tengely-párhuzamos profilírozás módnál		
8		<_DZ>	REAL	lehúzási érték Z G18-nál és G19-nél tengely-párhuzamos profilírozás módnál		
9		<S_PA>	REAL	profilozás ráhagyás		
10		<S_N>	INT	löketek száma a profilozás programban		
11		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/G18/G19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
12		<_AMODE>	INT	alternatív módus		
				EGYES:	profilozás választás új/folytatás	
					1 =	új
					2 =	folytatás
				TÍZES:	profilozás ráhagyás kiválasztása	
					0 =	a nyers kontúrtól a kontúr legmélyebb pontjáig
					1 =	a nyers kontúrtól a kontúr legmagasabb pontjáig
13		<S_FW>	REAL	lehúzó szabadszöge		
14		<S_HW>	REAL	lehúzó tartószöge		

20.1.33 CYCLE800 - Billentés

Szintaxis

```
CYCLE800(<_FR>, <_TC>, <_ST>, <_MODE>, <_X0>, <_Y0>, <_Z0>, <_A>,
<_B>, <_C>, <_X1>, <_Y1>, <_Z1>, <_DIR>, <_FR_I>, <_DMODE>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
1		<_FR>	INT	szabadra-menet módus:	0 =	nincs szabadra menet
					1 =	szabadra menet Z géptengely
					2 =	szabadra-menet géptengely Z és utána XY
					3 =	foglalt
					4 =	szabadra menet szerszámirányban maximális
					5 =	szabadra menet szerszámirányban növekményes
2		<_TC>	STRING[32]	billentési adatkészlet neve:	""	"" (nincs név) ha csak 1 billentési adatkészlet van
					"0"	billentési adatkészlet kikapcsolása (billentési frame törlése)

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
3		<_ST>	INT	transzformációk állapota		
				EGYES:		
					0 =	új, billentési sík törölve és az aktuális paraméterekkel újra számítva lesz
					1 =	additív, billentési sík hozzáadódik az aktív billentési síkhoz
				TÍZES:	szerszámcsúcs utánvezetés igen=nem)csak akkor aktív, ha BILLENTÉS ÜH-ben be lett állítva)	
					0 =	szerszámcsúcs utánvezetés
					1 =	szerszámcsúcs utánvezetése (TRAORI)
				SZÁZAS:	szerszámot ráállítani = beállítani (funkció a szerszám BILLENTÉS beadási maszkban lesz kijelezve)	
					0 =	szerszámot nem ráállítani
					1 =	szerszámot (főleg sugármarót) ráállítani
					2 =	**eszterga**szerszámot beállítani (ha B tengely-kinematika az esztergálás-technológiához be lett állítva a billentés ÜH-nél)
					3 =	**maró**szerszámot beállítani (ha B tengely-kinematika az esztergálás-technológiához be lett állítva a billentés ÜH-nél)
				EZRES:	belső paraméter billentés JOG-ban	
				TÍZEZRES:	lásd paraméter irány <_DIR>	
					0 =	billentés "igen"
					1 =	billentés "nem" irány "mínusz" [3]
					2 =	billentés "nem" irány "plusz" [3]
				SZÁZEZRES:	lásd paraméter irány <_DIR>	
					0 =	kompatibilitás
					1 =	irány-választás "mínusz" optimalizálva (csak a kezelőfelületre) [4]
					2 =	irány-választás "plusz" optimalizálva (csak a kezelőfelületre) [4]

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
4		<_MODE> [5]	INT	billentés modus: billentési szög és billentési sorrend kiértékelése (bit-kódolt!)		
				bit: 7 6	0 0:	billentési szög tengelyenként -> lásd paraméter <_A>, <_B>, <_C>
					0 1:	térszög -> lásd paraméter <_A>, <_B> [1]
					1 0:	vetítési szög -> lásd paraméter <_A>, <_B>, <_C> [1]
					1 1:	körtengelyek billentési mód közvetlen -> lásd paraméter <_A>, <_B>[1]
				bit: 5 4 3 2 1 0 (térszögnél nincs jelentősége!)	x x x x 0 1	1. forgatás _A X körül
					x x x x 1 0	1. forgatás _A Y körül
					x x x x 1 1	1. forgatás _A Z körül
					x x 0 1 x x	2. forgatás _B X körül
					x x 1 0 x x	2. forgatás _B Y körül
					x x 1 1 x x	2. forgatás _B Z körül
					0 1 x x x x	3. forgatás _C X körül
					1 0 x x x x	3. forgatás _C Y körül
					1 1 x x x x	3. forgatás _C Z körül
5	X0	<_X0>	REAL	vonatkoztatási pont X forgatás előtt		
6	Y0	<_Y0>	REAL	vonatkoztatási pont Y forgatás előtt		
7	Z0	<_Z0>	REAL	vonatkoztatási pont Z forgatás előtt		
8	X(A)	<_A>	REAL	1. forgatás a <_MODE> paraméterben beállított szerint		
9	Y(B)	<_B>	REAL	2. forgatás a <_MODE> paraméterben beállított szerint		
10	Z(C)	<_C>	REAL	3. forgatás a <_MODE> paraméterben beállított szerint		
11	X1	<_X1>	REAL	vonatkoztatási pont X forgatás után		
12	Y1	<_Y1>	REAL	vonatkoztatási pont Y forgatás után		
13	Z1	<_Z1>	REAL	vonatkoztatási pont Z forgatás után		
14	- vagy -	<_DIR>	INT	körtengelyek elmozgatását kiváltani (alapérték = -1!)		
					-1 =	körtengely 1 vagy 2 kisebb értékére pozicionálni[2]
					+1 =	körtengely 1 vagy 2 nagyobb értékére pozicionálni[2]
					0 =	billentés nem (csak billentés-frame-t kiszámítani) [1] [3]
15	FR	<_FR_I>	REAL	érték (növ) szabadra menet szerszámirányban növekményes		

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
16		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/G18/G19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
				TÍZES:	béta érték ábrázolása szerszám beállításnál	
					0 =	érték
					1 =	nyíl

Megjegyzés

Ha a következő átadási paraméterek közvetetten (paraméterként) vannak programozva, a beadási maszk nem lesz visszafordítva: <_FR>, <_ST>, <_TC>, <_MODE>, <_DIR>

[1] Kiválasztás lehetséges, ha BILLENTÉS ÜH-ben be lett állítva.

[2] Kiválasztás lehetséges, ha BILLENTÉS ÜH-ben az irány-vonatkoztatás a körtengely 1 vagy 2-re be lett állítva.

nincs kiválasztási mező irány-vonatkoztatás nem esetén

[3] Billentés "nem" kiválasztása eltüntethető SD 55221 bit 0

Billentés "nem" irány "mínusz" megfelel <_DIR> = 0 és _ST TÍZEZRES = 1-nek

Billentés "nem" irány "plusz" megfelel <_DIR> = 0 és _ST TÍZEZRES = 2-nek

[4] A körtengely 1 vagy 2 irány kiválasztás megtörténik akkor is, ha a körtengely az irány-vonatkoztatással a pólushelyen)pozícióérték nulla) van.

[5] Kódolás példa: forgatás tengelyenként, forgatási sorrend ZYX

bináris: 00011011; decimális: 27

Az XYZ tengelyjelölők megfelelnek az NC csatorna geometria-tengelyeinek. Az XYZ tengelyek körüli forgatásokat egyenként végre lehet hajtani. Pl. ZXZ forgatási sorrend a CYCLE800 felhívásában nem megengedett

20.1.34 CYCLE801 - Rács vagy keret

Szintaxis

```
CYCLE801(<_SPCA>, <_SPCO>, <_STA>, <_DIS1>, <_DIS2>, <_NUM1>,
<_NUM2>, <_VARI>, <_UMODE>, <_ANG1>, <_ANG2>, <_HIDE>, <_NSP>,
<_DMODE>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belső	Adattípus	Jelentés		
1	X0	<_SPCA>	REAL	pozícióminta (rács/keret) vonatkoztatási pont az 1. tengelyen (absz)		
2	Y0	<_SPCO>	REAL	pozícióminta (rács/keret) vonatkoztatási pont az 2. tengelyen (absz)		
3	α0	<_STA>	REAL	alap forgatási szög (szög az 1. tengelyhez)	< 0 =	forgatás órajárás irányában
					> 0 =	forgatás órajárással szemben
4	L1	<_DIS1>	REAL	oszlopok távolsága (1. tengely pozíció-távolság, előjel nélkül beadni)		
5	L2	<_DIS2>	REAL	sorok távolsága (2. tengely pozíció-távolság, előjel nélkül beadni)		
6	N1	<_NUM1>	INT	oszlopok száma		
7	N2	<_NUM2>	INT	sorok száma		
8		<_VARI>	INT	megmunkálási mód		
				EGYES:	pozícióminta	
					0 =	rács
					1 =	keretek
				TÍZES:	foglalt	
				SZÁZAS:	foglalt	
9		<_UMODE>	INT	foglalt		
10	αX	<_ANG1>	REAL	ollószög az 1. tengelyhez (sorok ferdesége az 1. tengelyre vonatkoztatva)		
					< 0 =	mérés órajárás irányában (0 ... -90 fok)
					> 0 =	mérés órajárással szemben (0 ... 90 fok)
11	αY	<_ANG2>	REAL	ollószög az 2. tengelyhez (sorok ferdesége az 2. tengelyre vonatkoztatva)		
					< 0 =	mérés órajárás irányában (0 ... -90 fok)
					> 0 =	mérés órajárással szemben (0 ... 90 fok)
12		<_HIDE>	STRING [200]	kihagyott pozíciók • max. 198 karakter • folytatólagos pozíciószámok megadása, pl. "1,3" (1 és 3 pozíciók nem lesznek végrehajtva)		
13		<_NSP>	INT	foglalt		
14		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/G18/G19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)

20.1.35 CYCLE802 - Tetszőleges pozíciók

Szintaxis

```
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>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belső	Adattípus	Jelentés		
1		<_XA>	INT	alternatíva minden X pozícióra (9 jegyű decimális érték) helyszám: 876543210 (a hely megfelel az Xn furatpozíciónak)		
				helyiérték	1 =	abszolút (1. programozott pozíció mindig abszolút)
					2 =	növekményes
2		<_YA>	INT	alternatíva minden Y pozícióra (9 jegyű decimális érték) helyszám: 876543210 (a hely megfelel az Yn furatpozíciónak)		
				helyiérték	1 =	abszolút (1. programozott pozíció mindig abszolút)
					2 =	növekményes
3	X0	<_X0>	REAL	1. X pozíció		
4	Y0	<_Y0>	REAL	1. Y pozíció		
5	X1	<_X1>	REAL	2. X pozíció		
6	Y1	<_Y1>	REAL	2. Y pozíció		
7	X2	<_X2>	REAL	3. X pozíció		
8	Y2	<_Y2>	REAL	3. Y pozíció		
9	X3	<_X3>	REAL	4. X pozíció		
10	Y3	<_Y3>	REAL	4. Y pozíció		
11	X4	<_X4>	REAL	5. X pozíció		
12	Y4	<_Y4>	REAL	5. Y pozíció		
13	X5	<_X5>	REAL	6. X pozíció		
14	Y5	<_Y5>	REAL	6. Y pozíció		
15	X6	<_X6>	REAL	7. X pozíció		
16	Y6	<_Y6>	REAL	7. Y pozíció		
17	X7	<_X7>	REAL	8. X pozíció		
18	Y7	<_Y7>	REAL	8. Y pozíció		
19	X8	<_X8>	REAL	9. X pozíció		
20	Y8	<_Y8>	REAL	9. Y pozíció		

Nr.	Paraméter maszk	Paraméter belső	Adattípus	Jelentés		
21		<_VARI>	INT	megmunkálás		
				SZÁZAS:	(csak Jobshop felhívásra) (egyenlőre csak 0 és 2 van kiértékelve)	
					0 =	nincs orsó rögzítés
					1 =	orsó rögzítés csak merőleges bemerülésnél G00-val vagy G01-gyel
					2 =	orsó rögzítés a teljes megmunkálás alatt
				EZRES:	foglalt	
				TÍZEZRES:	pozícióminta körtengellyel/anélkül – tengely kombináció (<_VARI> SZÁZEZRES-sel)	
					0 =	XY (csak XY körtengely nélkül, kompatibilitás)
					1 =	X,Y vagy Z és körtengely: XA, YB, ZC (1 körtengely geometria-tengellyel, amelyik körül a körtengely forog)
					2 =	XY és körtengely: XYA, XYB, XYC (1 körtegely az 1. és 2. geometria tengellyel, TRACYL nélkül)
				SZÁZEZRES:	körtengely	
					0 =	körtengely nélkül (csak XY kompatibilitás)
					1 =	A tengely (körtengely X körül)
					2 =	B tengely (körtengely Y körül)
					3 =	C tengely (körtengely Z körül)
				TÍZMILLIÓ + EGYMILLIÓ:	pozíció minta körtengellyel - offset (több körtengely azonos tengely körül esetén; ha az index túl nagy, akkor az 1. tengely)	
					00 =	1. A, B vagy C tengely vagy kompatibilitás
					01 =	2. A, B vagy C tengely
					...	
					19 =	20 A, B vagy C tengely
22		<_UMODE>	INT	rögzítendő orsó kiválasztása: (csak Jobshop híváshoz) (CUST_TECHCYC felhasználói ciklus felhívása)		
					3 =	főorsó rögzítés/oldás
					23 =	ellenorsó rögzítés/oldás

Nr.	Paraméter maszk	Paraméter belső	Adattípus	Jelentés		
23		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/G18/G19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
24		<S_ABA>	INT	alternatíva minden AB pozícióra (9 jegyű decimális érték) helyszám: 876543210 (a hely megfelel az ABn pozíciónak)		
				helyiérték:	1 =	abszolút (1. programozott pozíció mindig abszolút)
					2 =	növekményes
25	A0	<S_AB0>	REAL	1. körtengely pozíciók pozíció-minta körtengellyel esetén (<_VARI>-vel kapcsolatban)		
26	A1	<S_AB1>	REAL	2. körtengely pozíciók pozíció-minta körtengellyel esetén		
27	A2	<S_AB2>	REAL	3. körtengely pozíciók pozíció-minta körtengellyel esetén		
28	A3	<S_AB3>	REAL	4. körtengely pozíciók pozíció-minta körtengellyel esetén		
29	A4	<S_AB4>	REAL	5. körtengely pozíciók pozíció-minta körtengellyel esetén		
30	A5	<S_AB5>	REAL	6. körtengely pozíciók pozíció-minta körtengellyel esetén		
31	A6	<S_AB6>	REAL	7. körtengely pozíciók pozíció-minta körtengellyel esetén		
32	A7	<S_AB7>	REAL	8. körtengely pozíciók pozíció-minta körtengellyel esetén		
33	A8	<S_AB8>	REAL	9. körtengely pozíciók pozíció-minta körtengellyel esetén		

Megjegyzés

Az X1/Y1/A1 ... X8/Y8/A8 paraméterek nem szükséges pozícióit el lehet hagyni. Az <_XA>, <_YA> és <S_ABA> alternatív értékeket azonban teljesen meg kell adni mind a 9 pozícióra.

Az XA, YB vagy ZC pozíciómintánál (egy geometria tengely és körtengely) a megmunkálási sík tengelye, a pozíciómintán túl nem mozog (G17 és XA esetén az Y) a ciklus hívás előtt pozicionálni.

20.1.36 CYCLE830 - Mélylyuk fúrása 2

Szintaxis

```
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>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
1	RP	<RTP>	REAL	visszahúzási sík (absz.)		
2	Z0	<RFP>	REAL	vonatkoztatási pont (absz.)		
3	SC	<SDIS>	REAL	biztonsági távolság (vonatkoztatási ponthoz adódik, előjel nélkül beadni)		
4	Z1	<_DP>	REAL	végső furatmélység absz/növ, (lásd <_AMODE> EGYES)		
5	D	<FDEP>	REAL	1. furatmélység forgácstöréshez/forgácstalanításhoz abszolút vagy növekményes a vonatkoztatási pontra vonakoztatva felfúrásnál/felfúrás nélkül (Lásd <_AMODE> TÍZEZRES)		
6	DF	<_DAM>	REAL	érték/százalék minden további fogás csökkenési értékhez vagy százalékhoz (lásd <_AMODE> SZÁZEZRES)		
7	DTB	<DTB>	REAL	várakozási idő minden furatmélységen (lásd <_AMODE> TIZES)		
8	DTS	<DTS>	REAL	várakozási idő forgácstalanítás a kezdőponton (lásd <_AMODE> SZÁZAS)		
9	FD1	<FRF>	REAL	előtolás százalék az első fogásnál (lásd <_AMODE> TÍZMILLIÓS)		
10		<VARI>	INT	megmunkálás		
				EGYES:	forgácstörés/forgácstalanítás	
					0 =	egy vágással
					1 =	forgácstörés
					2 =	forgácstalanítás
					3 =	forgácstörés és forgácstalanítás
				TÍZES:	visszahúzás forgácstalanításnál	
					0 =	próba-fúrás mélységre
					1 =	biztonsági távolságra
				SZÁZAS:	puha bevágás	
					0 =	nem
					1 =	igen
				EZRES:	átfúrás	
					0 =	nem
					1 =	igen
				TÍZEZRES:	felfúrás/próba-fúrás	
					0 =	felfúrás nélkül
					1 =	felfúrással
					2 =	próba-fúrással
				SZÁZEZRES:	visszahúzás	
					0 =	próba-fúrás mélységre
					1 =	visszahúzási síkra
11	V1	<_MDEP>	REAL	minimális fogás növekményes (csak degresszió százaléknál)		
12	V2	<_VRT>	REAL	visszahúzási érték minden megmunkálás után (csak forgácstörésnél)		
					0 =	alapérték 1 mm
					> 0 =	változó visszahúzási érték
13	DT	<_DTD>	REAL	várakozási idő végső furatmélységen (lásd <_AMODE> EZRES)		

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
14	V3	<_DIS1>	REAL	elő-leállás távolság csak forgácstalanításnál (lásd <_AMODE> EGY-MILLIÓS)		
15	FP	<S_FP>	REAL	maradék furat előtolás értékként vagy %-ban (összeköttetésben a <S_AMODE2> -vel SZÁZEZRES)		
16		<S_SDAC2>	INT	orsó-forgásirány rámenetnél		
					3 =	M3
					4 =	M4
					5 =	M5 (alap)
17	SP	<S_SV2>	REAL	rámenet	állandó orsó-fordulatszámmal (lásd <S_AMODE2> TÍZMILLIÓS)	
	V4				állandó vágósebesség	
				orsó-fordulatszám a fúrás fordulatszám %-ában		
18	F	<S_FB>	REAL	fúrás előtolás (lásd <S_AMODE2> EGYES)		
19		<_SDAC>	REAL	orsó-forgásirány fúrásnál		
					3 =	M3
					4 =	M4
20	S	<_SV1>	REAL	fúrás	állandó orsó-fordulatszámmal (lásd <S_AMODE2> EGYMILLIÓS)	
	V5				állandó vágósebesség	
21	SPOS	<S_SPOS>	REAL	orsópozíció, csak ha rámenet M5-tel		
22	ZA	<S_ZA>	REAL	felfúrás mélység növekményesen vonatkoztatva a vonatkoztatási pontra vagy abszolút (lásd <S_AMODE3> EGYES)		
23	FA	<S_FA>	REAL	felfúrás előtolás értékként vagy %-ban (összeköttetésben a <S_AMODE2> -vel TÍZES)		
24	ZP	<S_ZP>	REAL	próba-fúrás növekményesen vonatkoztatási pontra vonatkoztatva, abszolút vagy furat-átmérő tényező (lásd <S_AMODE3> TÍZES)		
25	FS	<S_FS>	REAL	előtolás vágás értékként vagy %-ban (összeköttetésben a <S_AMODE2> -vel EZRES)		
26	ZS1	<S_ZS1>	REAL	minden bevágás mélysége állandó előtolásnál (növ)		
27	ZS2	<S_ZS2>	REAL	minden bevágás mélysége az előtolás emeléshez (növ)		
28	N	<S_N>	INT	forgácstörő löketek száma minden forgácstalanítás előtt		
29	ZD	<S_ZD>	REAL	maradék furat mélység növekményesen vonatkoztatva a végső furat mélységre vagy abszolút (lásd <S_AMODE3> SZÁZAS)		
30	FD	<S_FD>	REAL	maradék furat előtolás értékként vagy %-ban (összeköttetésben a <S_AMODE2> -vel TÍZEZRES)		
31	FR	<S_FR>	REAL	visszahúzás előtolás (összeköttetésben <S_AMODE2> -vel SZÁZEZ-RES)		
32		<S_SDAC3>	INT	orsó-forgásirány visszahúzásnál		
					3 =	M3
					4 =	M4
					5 =	M5
33	SR	<S_SV3>	REAL	visszahúzás	állandó orsó-fordulatszámmal (lásd <S_AMODE2> SZÁZMILLIÓS)	
	V6				állandó vágósebesség	
				orsó-fordulatszám a fúrás fordulatszám %-ában		

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
34	hűtővíz be	<S_CON>	STRING[10]	hűtővíz be, M utasítás vagy alprogramhívás		
35	hűtővíz ki	<S_COFF>	STRING[10]	hűtővíz ki, M utasítás vagy alprogramhívás		
36		<_GMODE>	INT	geometria módus (programozott geometria-értékek értékelése)		
				EGYES:	foglalt	
				TÍZES:	furatmélység csúcsra/szárra vonatkoztatva	
					0 =	csúcs
					1 =	szár
37		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/G18/G19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
				TÍZES:	foglalt	
				SZÁZAS:	foglalt	
				EZRES:	foglalt	
				TÍZEZRES:	Technológia skálázás ciklus maszkokon belül (Oldal 818)	
					0 =	beadás: teljes
					1 =	beadás: egyszerű

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
38		<_AMODE>	INT	alternatív módus 1		
				EGYES:	furatmélység = végső furatmélység Z1 absz/növ	
					0 =	növekményes
					1 =	abszolút
				TÍZES:	várakozási idő minden furatmélységen DTB másodpercekben/fordulatokban	
					0 =	másodpercekben
					1 =	fordulatokban
				SZÁZAS:	várakozási lazításhoz DTS másodpercekben/fordulatokban	
					0 =	másodpercekben
					1 =	fordulatokban
				EZRES:	várakozási idő végső furatmélységen DT másodpercekben/fordulatokban	
					0 =	másodpercekben
					1 =	fordulatokban
				TÍZEZRES:	1. furatmélység D absz/növ	
					0 =	növekményes
					1 =	abszolút
				SZÁZEZRES:	érték/százalék DF minden további fogáshoz (degresszió)	
					0 =	érték
					1 =	százalék (0,001 ... 100 %)
				EGYMILLIÓS:	elő-leállítás távolság V3 automatikus/kézi	
					0 =	automatikus (a ciklusban lesz kiszámítva)
					1 =	kézi (programozott érték)

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
39		<S_AMODE2>	INT	alternatív módus 2		
				EGYES:	EGYES: pályaelőtolás F	
					0 =	F/prec
					1 =	F/ford
				TÍZES:	felfúrás előtolás értéke FA	
					0 =	pályaelőtolás %-ában
					1 =	F/prec
					2 =	F/ford
				SZÁZAS:	előtolás érték bejáratáshoz próba-fúrás-ban FP	
					0 =	pályaelőtolás %-ában
					1 =	F/prec
					2 =	F/ford
				EZRES:	vágás előtolás érték FS	
					0 =	pályaelőtolás %-ában
					1 =	F/prec
					2 =	F/ford
				TÍZEZRES:	átfúrás előtolás érték FD	
					0 =	pályaelőtolás %-ában
					1 =	F/prec
					2 =	F/ford
				SZÁZEZRES:	előtolás visszahúzásnál FR	
					0 =	F/prec
					1 =	gyorsmenet
				EGYMILLIÓS:	fúrás - orsó-fordulatszám/vágósebesség (S/V5)	
					0 =	állandó orsó-fordulatszám
					1 =	állandó vágósebesség
				TÍZMILLIÓS:	rámenet orsó-fordulatszámmal/vágóse-bességgel (SP/V4)	
					0 =	állandó orsó-fordulatszám
					1 =	állandó vágósebesség
					2 =	orsó-fordulatszám a fúrás fordu-latszám %-ában
				SZÁZMILLIÓS:	visszahúzás - orsó-fordulatszám/vágó-sebesség (SR/V6)	
					0 =	állandó orsó-fordulatszám
					1 =	állandó vágósebesség
					2 =	orsó-fordulatszám a fúrás fordu-latszám %-ában

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
40		<S_AMODE3>	INT	alternatív módus 3		
				EGYES:	felfúrás mélység ZA absz/növ	
					0 =	növekményes
					1 =	abszolút
				TÍZES:	próba-fúrás mélysége ZP	
					0 =	növekményes
					1 =	abszolút
					2 =	furat átmérő tényező
				SZÁZAS:	maradék furatmélység ZD absz/növ	
					0 =	növekményes
					1 =	abszolút
41	ZPV	<S_ZPV>	REAL	elő-leállás távolság növekményes próbafúrás-mélységre		

20.1.37 CYCLE832 - High Speed Settings

Szintaxis

```
CYCLE832(<S_TOL>, <S_TOLM>, <S_OTOL>)
```

Megjegyzés

A CYCLE832 nem mentesíti a gépgyártót a szükséges optimalizálási feladatoktól a gép üzembehelyezésénél. Ez érinti a megmunkálásban résztvevő tengelyek optimalizálását és az NCU beállításait (elővezérlés, rántás-határolás, stb.).

Paraméter

Nr.	Paramé-ter maszk	Paraméter belül	Adattípus	Jelentés		
1	Tűrés	<S_TOL>	REAL	kontúr tűrés A kontúr tűrés megfelel a geometria-tengelyek tengelytűrésének.		
2		<S_TOLM>	INT	Megmunkálási mód (technológia)		
				EGYES:		
					0 =	kikapcsolás
					1 =	simítás (Finish)
					2 =	elő-simítás (Semifinish)
					3 =	nagyolás (Rough)
				TÍZES:		
					0 =	kompatibilitás[1] ill. nincs tájolási tűrés
					1 =	tájolási tűrés a <S_OTOL> paraméterben
				SZÁZAS ... SZÁZEZRES	foglalt miatt kompatibilitás	
				EGYMILLIÓS:		
					0 =	Kompatibilitás. Automatikusan az utolsó rendelkezésre álló formakészítés funkció lesz használva: • Top Surface opció nem aktív: ⇒ Advanced Surface • Top Surface opció aktív: ⇒ Top Surface simítással
					1 =	Top Surface simítás nélkül
					2 =	Top Surface simítással
3	ORI tűrés	<S_OTOL>	REAL	tájolás tűrés ill. CYCLE832 verzió jelölő tűrési paraméter a szerszám tájolásához Szükséges egy nagy sebességű megmunkálási program feldolgozásához gépeken dinamikus tájolási transzformációval (pl. 5 tengelyes megmunkálás). A <S_OTOL> paraméter programozva **kell** legyen. Ez érvényes a 3 tengelyes gépeken való alkalmazásokra is a szerszám tájolása nélküli programozásnál (<S_OTOL> = 1).		

[1] tájolási tűrés levezetve a kontúr tűrésből szorozva a tényezővel az SD55441 ... SD55443 ciklus beállítási adatokból.
Irodalom:
Alap szoftver és kezelő szoftver üzembehelyezési kézikönyv; SINUMERIK Operate (IM9), "High Speed Setting (CYCLE832) funkció konfigurálása" fejezet

Szöveg beadása

A ciklus hívásának olvashatósága javítására a <S_TOLM> paramétert (Megmunkálási mód) szövegesen is meg lehet adni. A szövegek nyelv-függetlenek. A következő beadások megengedettek:

_OFF	következőkhöz:	0	kikapcsolás
_FINISH	következőkhöz:	1	simítás
_SEMIFIN	következőkhöz:	2	elősimítás
_ROUGH	következőkhöz:	3	nagyolás
_ORI_FINISH	következőkhöz:	11	simítás egy tájolási tűrés megadásával
_ORI_SEMIFIN	következőkhöz:	12	elősimítás egy tájolási tűrés megadásával
_ORI_ROUGH	következőkhöz:	13	nagyolás egy tájolási tűrés megadásával
_TOP_SURFACE_SMOOTH_OFF	következőkhöz:	1000000	Top Surface simítás nélkül

_TOP_SURFACE_SMOOTH_ON	kö-vet-ke-zők-höz:	2000000	Top Surface simítással

A szöveg beadására Top Surface-nél a szövegek a következő példa szerint lesznek kombinálva:

```
CYCLE832(0.1, _TOP_SURFACE_SMOOTH_OFF+_ORI_FINISH, 1)
```

Megjegyzés

A szövegek hasonlítanak a 59-es G-csoport (dinamika módus pálya-interpolációhoz) funkció neveire. Ezzel a szöveggel a 3 tengelyes gépek a gépek több-tengelyes tájolási transzformációval (TRAORI) az alkalmazásban tisztán el lesznek választva.

CYCLE832 kikapcsolása

A CYCLE832 kikapcsolásánál az `<S_TOL>` paramétert nullával kell átadni.

Példa: `CYCLE832(0,0,1)`

A `CYCLE832()` szintakszis szintén megengedett a CYCLE832 kikapcsolásához.

Példák

Példa 1: CYCLE832 3 tengelyes gépen tájolási transzformáció nélkül

a) ciklus felhívás szöveg beadással

Programkód	Kommentár
G710	; mértékrendszer metrikus
CYCLE832(0.004,_FINISH,1)	; CYCLE832 felhívása: kontúr tűrés = 0,004 mm, megmunkálási mód: simítás
...	; egy nagy sebességű megmunkáló program feldolgozása

b) ciklus felhívás szöveg beadás nélkül

Programkód	Kommentár
G710	; lásd fent
CYCLE832(0.004,1,1)	; lásd fent
...	; lásd fent

Példa 2: CYCLE832 5 tengelyes gépen tájolási transzformációval

a) ciklus felhívás és kikapcsolás szöveg beadással

Programkód	Kommentár
G710	; mértékrendszer metrikus
TRAORI	; tájolás-transzformációt aktiválni
CYCLE832(0.3,_ORI_ROUGH,0.8)	; CYCLE832 felhívása: kontúr tűrés = 0,3 mm, megmunkálási mód: nagyolás egy tájolási tűrés megadásával, tájolási tűrés = 0,8 fok
...	; egy nagy sebességű megmunkáló program feldolgozása
CYCLE832(0,_OFF,1)	; kontúr tűrés = 0, megmunkálási mód: CYCLE832 kikapcsolása, tájolási tűrés = 0 fok

b) ciklus felhívás és kikapcsolás szöveg beadás nélkül

Programkód	Kommentár
G710	; lásd fent
TRAORI	; lásd fent
CYCLE832(0.3,13,0.8)	; lásd fent
...	; lásd fent
CYCLE832(0,0,1)	; lásd fent

20.1.38 CYCLE840 - Menetfúrás kiegyenlítő tokmánnyal

Szintaxis

```
CYCLE840(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <DTB>, <SDR>, <SDAC>,
<ENC>, <MPIT>, <PIT>, <_AXN>, <_PITA>, <_TECHNO>, <_PITM>, <_PTAB>,
<_PTABA>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés
1	RP	<RTP>	REAL	visszahúzási sík (absz.)
2	Z0	<RFP>	REAL	vonatkoztatási pont (absz.)
3	SC	<SDIS>	REAL	biztonsági távolság (vonatkoztatási ponthoz adódik, előjel nélkül beadni)
4	Z1	<DP>	REAL	furatmélység (absz), lásd <_AMODE>
5	Z1	<DPR>	REAL	furatmélység (növ), lásd <_AMODE>
6	DT	<DTB>	REAL	várakozási idő furatmélységen / biztonsági távolságon visszahúzás után másodpercben, lásd <ENC>
7		<SDR>	INT	forgásirány visszahúzáshoz

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés			
8	SDE	<SDAC>	INT	forgásirány a ciklus vége után			
9		<ENC>	INT	menetfúrás orsó-jeladóval (G33) / menetfúrás orsó-jeladó nélkül (G63)			
					0 =	orsó jeladóval	- emelkedés <MPIT>/<PIT> - DT nélkül
					20 =	orsó jeladóval	- emelkedés <MPIT>/<PIT> - ből - DT-vel biztonsági távolságra visszahúzás után
					11 =	orsó jeladó nélkül	- emelkedés <MPIT>/<PIT> - ből - DT-vel furatmélységre
					1 =	orsó jeladó nélkül	- emelkedés programozott előtolásból - DT-vel furatmélységre (előtolás = fordulatszám · emelkedés)
10		<MPIT>	REAL	menetméret csak "ISO metrikus" számára (emelkedés futás közben belül lesz kiszámítva) értéktartomány: 3 ... 48 (M3 ... M48-ra), <PIT> alternatíva			
11		<PIT>	REAL	menetemelkedés értékként, mértékegység lásd <_PITA> értéktartomány: > 0, MPIT alternatíva			
12		<_AXN>	INT	fúró-tengely	0 =	3. geometria-tengely	
					1 =	1. geometria-tengely	
					2 =	2. geometria-tengely	
					≥ 3 =	3. geometria-tengely	
13		<_PITA>	INT	menetemelkedés mértékegysége (kiértékelés <PIT> és <MPIT>)			
					0 =	emelkedés mm-ben	- kiértékelés <MPIT>/<PIT>
					1 =	emelkedés mm-ben	- kiértékelés <PIT>
					2 =	emelkedés TPI-ben	- kiértékelés <PIT> (menetszám hüvelyenként)
					3 =	emelkedés hüvelykben	- kiértékelés <PIT>
					4 =	MODUL	- kiértékelés <PIT>

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
14		<_TECHNO>	INT	technológia[1]		
				EGYES:	pontos-állj viselkedés	
					0 =	pontos állj viselkedés mint a ciklus felhívása előtt aktív
					1 =	pontos állj G601
					2 =	pontos állj G602
					3 =	pontos állj G603
				TÍZES:	elővezérlés	
					0 =	elővezérléssel/anélkül mint a ciklus felhívása előtt aktív
					1 =	elővezérléssel FFWON
					2 =	elővezérlés nélkül FFWOF
15		<_PITM>	STRING[15]	string menetemelkedési beadás jelölőjeként[2]		
16		<_PTAB>	STRING[5]	string menettáblázathoz ("", "ISO", "BSW", "BSP", "UNC")[2]		
17		<_PTABA>	STRING[20]	string kiválasztáshoz a menettáblázatban (pl. "M 10", "M 12", ...)[2]		
18		<_GMODE>	INT	foglalt		
19		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/G18/G19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
				TÍZES:	foglalt	
				SZÁZAS:	foglalt	
				EZRES:	kompatibilitás módus (csak a beadási maszknak a visszafordításhoz), ha MD 52216 Bit0 = 1[1]	
					0 =	technológia paraméterek ki lesznek jelezve (kompatibilitás): TECHNO-paraméterek hatnak
					1 =	technológia paraméterek nem lesznek jelezve: technológia "mint a ciklusfelhívás előtt" hat
				TÍZEZRES:	Technológia skálázás ciklus maszkokon belül (Oldal 818)	
					0 =	beadás: teljes
					1 =	beadás: egyszerű

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
20		<_AMODE>	INT	alternatív módus		
				EGYES:	furatmélység Z1 (absz/növ)	
					0 =	kompatibilitás a <DP>/<DPR> programozásból
					1 =	növekményes
					2 =	abszolút

[1] Technológia mezők lehetnek kihagyva az SD52216 MCS_FUNCTION_MASK_DRILL beállítási adattól függően

[2] A 15, 16, és 17 paraméterek csak a beadási maszk menettáblázatában a menet kiválasztásnál lesznek használva. A hozzáférés a menettáblázatokhoz ciklus-definícióval a ciklus futásakor nem lehetséges!

20.1.39 CYCLE899 - Nyitott vájat marása

Szintaxis

```
CYCLE899(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_LENG>, <_WID>, <_PA>,
<_PO>, <_STA>, <_MID>, <_MIDA>, <_FAL>, <_FALD>, <_FFP1>, <_CDIR>,
<_VARI>, <_GMODE>, <_DMODE>, <_AMODE>, <_UMODE>, <_FS>, <_ZFS>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
1	RP	<_RTP>	REAL	visszahúzási sík (absz.)		
2	Z0	<_RFP>	REAL	szerszám-tengely vonatkoztatási pont (absz.)		
3	SC	<_SDIS>	REAL	biztonsági távolság (vonatkoztatási ponthoz adódik, előjel nélkül beadni)		
4	Z1	<_DP>	REAL	vájatmélység (absz/növ), lásd <_AMODE>		
5	L	<_LENG>	REAL	vájat hossza (növ)		
6	W	<_WID>	REAL	vájat szélessége (növ)		
7	X0	<_PA>	REAL	vonatkoztatási pont / kezdőpont az 1. tengelyen (absz)		
8	Y0	<_PO>	REAL	vonatkoztatási pont / kezdőpont az 2. tengelyen (absz)		
9	α0	<_STA>	REAL	forgatási szög az 1. tengelyhez		
10	DZ	<_MID>	REAL	maximális fogásmélység (növ) csak örvénylő maráshoz		
11	DXY	<_MIDA>	REAL	maximális sík fogásvétel, lásd <_AMODE>		
12	UXY	<_FAL>	REAL	sík simítási ráhagyás		
13	UZ	<_FALD>	REAL	simítási ráhagyás mélység		
14	F	<_FFP1>	REAL	előtolás		
15		<_CDIR>	INT	Marásirány		
				EGYES:		
					0 =	egyirányú
					1 =	ellenirányú
					4 =	váltakozva

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
16		<_VARI>	INT	Megmunkálás		
				EGYES:		
					1 =	nagyolás
					2 =	simítás
					3 =	simítás az alján
					4 =	simítás a peremen
					5 =	elősimítás
					6 =	letörés
				TÍZES:	foglalt	
				SZÁZAS:	foglalt	
				EZRES:		
					1 =	örvénylő marás
					2 =	bemerülő marás
17		<_GMODE>	INT	geometria módus (programozott geometria-értékek értékelése)		
				EGYES:	foglalt	
				TÍZES:	foglalt	
				SZÁZAS:	megmunkálás kiválasztása / csak kezdőpont kiszámítás	
					1 =	normál megmunkálás
				EZRES:	beméretezés középpel/éllel	
					0 =	méretezés középpel
					1 =	beméretezés "bal" éllel (1. tengely "-" iránya)
					2 =	beméretezés "jobb" éllel (1. tengely "+" iránya)
18		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/G18/G19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
				TÍZES:	---	foglalt
				SZÁZAS:	---	foglalt
				EZRES:	---	foglalt
				TÍZEZRES:	Technológia skálázás ciklus maszkokon belül (Oldal 818)	
					0 =	beadás: teljes
					1 =	beadás: egyszerű

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
19		<_AMODE>	INT	alternatív módus		
				EGYES:	Z1 vájatmélység	
					0 =	abszolút
					1 =	növekményes
				TÍZES:	egység sík fogásvételhez (<_MIDA>) DXY	
					0 =	mm
					1 =	szerszámátmérő %-a
				SZÁZAS:	bemerülési mélység letörésnél ZFS	
					0 =	abszolút
					1 =	növekményes
20		<_UMODE>	INT	foglalt		
21	FS	<_FS>	REAL	letörés szélesség (növ)		
22	ZFS	<_ZFS>	REAL	bemerülési mélység (szerszámcsúcs) letörésnél (absz/növ), lásd <_AMODE>		

20.1.40 CYCLE930 - Beszúrás

Szintaxis

```
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>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés
1	X0	<_SPD>	REAL	vonatkoztatási pont sík-tengelyben (mindig átmérő)
2	Z0	<_SPL>	REAL	vonatkoztatási pont hossz-tengelyben
3	B1	<_WIDG>	REAL	beszúrás szélesség lent
4	B2	<_WIDG2>	REAL	beszúrás szélesség fent (csak a felülethez)
5	T1	<_DIAG>	REAL	beszúrás szélesség vonatkoztatási ponton absz. és hossz-megmunkálás esetén = átmérő, egyébként növ.
6	T2	<_DIAG2>	REAL	beszúrás szélesség a vonatkoztatási ponttal szemben (csak a felülethez) absz. és hossz-megmunkálás esetén = átmérő, egyébként növ.
7	α0	<_STA>	REAL	ferdeség szöge (-180 ≤ <_STA> ≤ 180)
8	α1	<_ANG1>	REAL	élszög 1 (0 ≤ <_ANG1> < 90) a beszúrásnak a vonatkoztatási pont által meghatározott oldalán
9	α2	<_ANG2>	REAL	élszög 2 (0 ≤ <_ANG2> < 90) szemben a vonatkoztatási ponttal

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
10	R1/FS1	<_RCO1>	REAL	lekerekítési sugár vagy letörés szélesség 1, kívül a vonatkoztatási ponton		
11	R2/FS2	<_RCI1>	REAL	lekerekítési sugár vagy letörés szélesség 2, belül a vonatkoztatási ponton		
12	R3/FS3	<_RCI2>	REAL	lekerekítési sugár vagy letörés szélesség 3, belül szemben a vonatkoztatási ponttal		
13	R4/FS4	<_RCO2>	REAL	lekerekítési sugár vagy letörés szélesség 4, kívül szemben a vonatkoztatási ponttal		
14	U	<_FAL>	REAL	simítási ráhagyás X és Z, lásd <_VARI> (TÍZEZRES) (előjel nélkül beadni)		
15	D	<_IDEP1>	REAL	maximális mélységi fogásvétel bememrülésnél (előjel nélkül beadni)		
					0 =	1. vágás közvetlenül teljes mélységben
					> 0 =	1. vágás <_IDEP1>, 2. vágás 2 · <_IDEP1>, stb.
16	SC	<_SDIS>	REAL	biztonsági távolság (előjel nélkül beadni)		
17		<_VARI>	INT	Megmunkálási mód		
				EGYES:	foglalt	
				TÍZES:	technológiai megmunkálás	
					1 =	nagyolás
					2 =	simítás
					3 =	nagyolás és simítás
				SZÁZAS:	helyzet hossz/sík kívül/belül +Z/+Z ill. +X/-X	
					1 =	hossz/kívül +Z
					2 =	sík/belül -X
					3 =	hossz/beül +Z
					4 =	sík/belül +X
					5 =	hossz/kívül -Z
					6 =	sík/kívül -X
					7 =	hossz/belül -Z
					8 =	sík/kívül +X
				EZRES:	Vonatkoztatási pont helyzete	
					0 =	vonatkoztatási pont fent
					1 =	vonatkoztatási pont lent
				TÍZEZRES:	definíció simítási ráhagyásként hat	
					0 =	simítási ráhagyás U kontúr-párhuzamos
					1 =	simítási ráhagyás UX és UZ külön
18		<_DN>	INT	szerszám 2. vágóél D-száma		
					> 0 =	D-szám a beszúrókés 2. vágóél szerszámkorrekcióhoz
					0 =	nincs 2. vágóél programozva
19	N	<_NUM>	INT	beszúrások száma (0 = 1 beszúrás)		

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
20	DP	<_DBH>	REAL	beszúrások száma (csak akkor szükséges, ha <_NUM>> 1)		
21	F	<_FF1>	REAL	Előtolás		
22		<_NR>	INT	beszúrás forma jelölés megfelel a forma kiválasztás függőleges softkey-nek		
					0 =	90°-os él letörések/lekerekítések nélkül
					1 =	ferde él letörésekkel/lekerekítésekkel (α0 nélkül)
					2 =	mint 1, de kúpon (α0-val)
23	UX	<_FALX>	REAL	simítási ráhagyás X tengelyen, lásd <_VARI> (TÍZEZRES) (előjel nélkül beadni)		
24	UZ	<_FALZ>	REAL	simítási ráhagyás Z tengelyen, lásd <_VARI> (TÍZEZRES) (előjel nélkül beadni)		
25		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/G18/G19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
26		<_AMODE>	INT	alternatív módus		
				EGYES:	mélység méretezés (csak a felülethez)	
					0 =	vonatkoztatási ponton
					1 =	vonatkoztatási ponttal szemben
				TÍZES:	mélység	
					0 =	abszolút
					1 =	növekményes
				SZÁZAS:	szélesség méretezés (csak a felülethez)	
					0 =	külső átmérőn (fent)
					1 =	belső átmérőn (lent)
				EZRES:	sugár/letörés 1 (<_RCO1>)	
					0 =	sugár
					1 =	letörés
				TÍZEZRES:	sugár/letörés 2 (<_RCI1>)	
					0 =	sugár
					1 =	letörés
				SZÁZEZRES:	sugár/letörés 3 (<_RCI2>)	
					0 =	sugár
					1 =	letörés
				EGYMILLIÓS:	sugár/letörés 4 (<_RCO2>)	
					0 =	sugár
					1 =	letörés

20.1.41 CYCLE940 - Szabadra szúrás formák

A CYCLE940-nel különböző szabadra szúrásokat lehet programozni. Ezek a paraméterezésben részben jelentősen eltérnek.

A kiegészítő oszlopok a táblázatban mutatják, melyik paraméter melyik szabadra-szúrás formánál szükséges. Ezek megfelelnek a ciklus maszkok függőleges kiválasztó softkey-knek.

- E: szabadra-szúrás E forma
- F: szabadra-szúrás F forma
- A-D: szabadra-szúrás DIN menet (A-D forma)
- T: szabadra-szúrás menet (forma szabad definíciója)

Szintaxis

```
CYCLE940(<_SPD>, <_SPL>, <_FORM>, <_LAGE>, <_SDIS>, <_FFP>, <_VARI>,
<_EPD>, <_EPL>, <_R1>, <_R2>, <_STA>, <_VRT>, <_MID>, <_FAL>,
<_FALX>, <_FALZ>, <_PITI>, <_PTAB>, <_PTABA>, <_DMODE>, <_AMODE>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Progr. formánál				Jelentés		
				E	F	A-D	T			
1	X0	<_SPD>	REAL	x	x	x	x	vonatkoztatási pont sík-tengelyben (mindig átmérő)		
2	Z0	<_SPL>	REAL	x	x	x	x	hossz-tengely vonatkoztatási pont (absz)		
3	FORM	<_FORM>	CHAR	x	x	x	x	szabadra-szúrás formája (nagybetű, pl. "T") választás, hogy a szabadra-szúrás értékek melyik táblázatból legyenek véve		
									A =	kívül, vonatkoztatás DIN76, A = normál
									B =	kívül, vonatkoztatás DIN76, B = rövid
									C =	belül, vonatkoztatás DIN76, C = normál
									D =	belül, vonatkoztatás DIN76, D = rövid
									E =	vonatkozás DIN509
									F =	vonatkozás DIN509
									T =	szabad forma
4	HELY-ZET	<_LAGE>	INT	x	x	x	x	szabadra-szúrás helyzete (Z párhu-zamos)	0 =	kívül +Z: ____|
									1 =	kívül -Z: |____/
									2 =	belül +Z: /-----|
									3 =	belül -Z: |-----\
5	SC	<_SDIS>		x	x	x	x	biztonsági távolság (növ.)		
6	F	<_FFP>		x	x	x	x	megmunkálási előtolás (mm/ford)		

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Progr. formánál				Jelentés			
7		<_VARI>	INT	-	-	x	x	Megmunkálási mód			
								EGYES:	megmunkálás		
									1 =	nagyolás	
									2 =	simítás	
									3 =	nagyolás + simítás	
								TÍZES:	megmunkálási stratégia		
									0 =	kontúr-párhuzamos	
									1 =	hossz	
				Az E és F forma szabadra-szúrások mindig egy lépésben lesznek megmunkálva, mint a simításnál.							
8	X1	<_EPD>		x	x	-	-	ráhagyás X (absz/növ), lásd <_AMODE>			
				-	-	-	x	szabadra-szúrás mélység (absz/növ), lásd<_AMODE>			
9	Z1	<_EPL>		-	x	-	-	ráhagyás Z			
				-	-	-	x	szabadra-szúrás szélesség (absz/növ), lásd <_AMODE>			
10	R1	<_R1>		-	-	-	x	lekerekítési sugár ferdeségeken			
11	R2	<_R2>		-	-	-	x	lekerekítési sugár sarokban			
12	α	<_STA>		-	-	x	x	bemerülési szög			
13	VX	<_VRT>		x	x	-	-	sík-húzás X (absz/növ), <_AMODE>			
				-	-	x	x	sík-húzás X simításnál (absz/növ), lásd <_AMODE>			
14	D	<_MID>		-	-	x	x	mélységi fogásvétel			
15	U	<_FAL>		-	-	x	x	simítási ráhagyás kontúr-párhuzamos, lásd <_AMODE>			
16	UX	<_FALX>		-	-	x	x	simítási ráhagyás X			
17	UZ	<_FALZ>		-	-	x	x	simítási ráhagyás Z			
18	P	<_PITI>	INT	-	-	x	-	emelkedés kiválasztása, A-D forma, megfelel M1 ... M68-nak			
								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	-	-	sugár/mélység kiválasztása, E, F forma			
								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 menettáblázathoz ("", "ISO", "BSW", "BSP", "UNC") (csak a felülethez)			
20		<_PTABA>	STRING [20]					string kiválasztáshoz a menettáblázatban (pl "M 10", "M 12", ...) (csak a felülethez)			

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Progr. formánál				Jelentés		
21		<_DMODE>	INT					kijelzés módus		
				x	x	x	x	EGYES:	G17/G18/G19 megmunkálási sík	
									0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
									1 =	G17 (csak a ciklusban aktív)
									2 =	G18 (csak a ciklusban aktív)
									3 =	G19 (csak a ciklusban aktív)
22		<_AMODE>	INT					alternatív módus		
				x	x	-	x	EGYES:	paraméter <_EPD> ráhagyás X vagy szabadra szúrás mélység	
									0 =	abszolút (mindig átmérő)
									1 =	növekményes
				x	x	-	x	TÍZES:	paraméter <_EPL> ráhagyás Z vagy szabadra szúrás szélesség	
									0 =	abszolút
									1 =	növekményes
				x	x	x	x	SZÁZAS:	paraméter <_VRT> sík-húzás X	
									0 =	abszolút (mindig átmérő)
									1 =	növekményes
				-	-	x	x	EZRES:	simítási ráhagyás	
									0 =	simítási ráhagyás kontúr-párhuzamos (<_FAL>)
									1 =	simítási ráhagyás külön (<_FALX>/<_FALZ>)

20.1.42 CYCLE951 - Leforgácsolás

Szintaxis

```
CYCLE951(<_SPD>, <_SPL>, <_EPD>, <_EPL>, <_ZPD>, <_ZPL>, <_LAGE>,
<_MID>, <_FALX>, <_FALZ>, <_VARI>, <_RF1>, <_RF2>, <_RF3>, <_SDIS>,
<_FF1>, <_NR>, <_DMODE>, <_AMODE>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés
1	X0	<_SPD>	REAL	vonatkoztatási pont (absz. mindig átmérő)
2	Z0	<_SPL>	REAL	vonatkoztatási pont (absz.)
3	X1	<_EPD>	REAL	végpont
4	Z1	<_EPL>	REAL	végpont

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
5	XM α1 α2	<_ZPD>	REAL	közbenső pont, lásd <_DMODE> (TÍZES)		
6	ZM α1 α2	<_ZPL>	REAL	közbenső pont, lásd <_DMODE> (TÍZES)		
7	helyzet	<_LAGE>	INT	leforgácsolás sarok helyzete	0 =	kívül/hátul
					1 =	kívül/elől
					2 =	belül/hátul
					3 =	belül/elől
8	D	<_MID>	REAL	maximális fogásmélység bemerülésnél		
9	UX	<_FALX>	REAL	simítási ráhagyás X-ben		
10	UZ	<_FALZ>	REAL	simítási ráhagyás Z-ben		
11		<_VARI>	INT	Megmunkálási mód		
				EGYES:	leforgácsolási irány (hossz vagy sík) a koordináta-rendszerben	
					1 =	hossz
					2 =	sík
				TÍZES:		
					1 =	nagyolás a simítási ráhagyásig
					2 =	simítás
				SZÁZAS:	foglalt	
				EZRES:	foglalt	
				TÍZEZRES:	foglalt	
12	R1/FS1	<_RF1>	REAL	lekerekítési sugár vagy letörés szélesség 1, lásd <_AMODE> (TIZEZRES)		
13	R2/FS2	<_RF2>	REAL	lekerekítési sugár vagy letörés szélesség 2, lásd <_AMODE> (SZÁZEZRES)		
14	R3/FS3	<_RF3>	REAL	lekerekítési sugár vagy letörés szélesség 3, lásd <_AMODE> (EGYMILLIÓS)		
15	SC	<_SDIS>	REAL	biztonsági távolság		
16	F	<_FF1>	REAL	előtolás nagyoláshoz/simításhoz		
17		<_NR>	INT	leforgácsolási mód jelölés (megfelel a forma kiválasztás függőleges softkey-nek)		
					0 =	leforgácsolás 1, 90 fokos sarok letörés/lekerekítés nélkül
					1 =	leforgácsolás 2, 90 fokos sarok letöréssel/lekerekítéssel
					2 =	leforgácsolás 3, tetszőleges sarok letöréssel/lekerekítéssel

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
18		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/G18/G19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
				TÍZES:	beadási forma<_ZPD>/<_ZPL>	
					0 =	Xm/Zm
					1 =	Xm/α1
					2 =	Xm/α2
					3 =	α1/Zm
					4 =	α2/Zm
					5 =	α1/α2
21		<_AMODE>	INT	alternatív módus		
				EGYES:	közbenső pont X-ben	
					0 =	abszolút, sík-tengely értéke átmérőben
					1 =	növekményes, sík-tengely értéke sugárban
				TÍZES:	közbenső pont Z-ben	
					0 =	abszolút
					1 =	növekményes
				SZÁZAS:	végpont X-ben	
					0 =	abszolút, sík-tengely értéke átmérőben
					1 =	növekményes, sík-tengely értéke sugárban
				EZRES:	végpont Z-ben	
					0 =	abszolút
					1 =	növekményes
				TÍZEZRES:	sugár/letörés 1	
					0 =	sugár
					1 =	letörés
				SZÁZEZRES:	sugár/letörés 2	
					0 =	sugár
					1 =	letörés
				EGYMILLIÓS:	sugár/letörés 3	
					0 =	sugár
					1 =	letörés

20.1.43 CYCLE952 - Kontúr-szúrás

Szintaxis

```
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>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés
1	PRG	<_PRG>	STRING[100]	leforgácsoló program neve
2	CON	<_CON>	STRING[100]	program neve, amiből az aktualizált nyersdarab-kontúr olvasva lesz (maradék megmunkálásnál)
3	CONR	<_CONR>	STRING[100]	program neve, amibe az aktualizált nyersdarab-kontúr (lásd <_AMODE> TÍZEZRES) írva lesz

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
4		<_VARI>	INT	megmunkálási mód		
				EGYES:	leforgácsolási mód	
					1 =	hossz
					2 =	sík
					3 =	kontúr-párhuzamos
				TÍZES:	technológiai megmunkálás (lásd <_GMODE> SZÁZAS)	
					1 =	nagyolás
					2 =	simítás
					3 =	foglalt
					4 =	nagyolás kétcsatornás
					5 =	simítás kétcsatornás
				SZÁZAS:	megmunkálási irány	
					1 =	megmunkálási irány X -
					2 =	megmunkálási irány X +
					3 =	megmunkálási irány Z -
					4 =	megmunkálási irány Z +
				EZRES:	fogásirány	
					1 =	kívül X -
					2 =	belül X +
					3 =	homlokoldal Z -
					4 =	hátsó-oldal Z +
				TÍZEZRES:	definíció simítási ráhagyásként hat	
					0 =	simítási ráhagyás UX és UZ külön
					1 =	simítási ráhagyás U kontúr-párhuzamos
				SZÁZEZRES:	utánhúzni	
					0 =	kompatibilitás, automatikus utánhúzás
					1 =	utánhúzással a kontúron
					2 =	utánhúzás nélkül
					3 =	automatikus utánhúzás
				EGYMILLIÓS:	hátravágások	
					0 =	hely nem lesz kiértékelve szúrásnál, - maradék és szúró-esztergálás, - maradék
					1 =	hátravágásokat megmunkálni
					2 =	hátravágásokat nem megmunkálni
				TÍZMILLIÓS:	hátul/forgáspont előtt	
					0 =	megmunkálás a forgásközép előtt
					1 =	foglalt

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
5	F	<_F>	REAL	előtolás nagyoláshoz/simításhoz		
	FZ			előtolás abszcissza szúró-esztergálás		
6	FR	<_FR>	REAL	előtolás hátravágásban bemerüléshez nagyolás		
	FX			előtolás ordináta szúró-esztergálás		
7	RP	<_RP>	REAL	visszahúzási sík belső megmunkálásnál (absz. mindig átmérő)		
8	D	<_D>	REAL	nagyolás fogás, (lásd <_AMODE> EGYES)		
9	DX	<_DX>	REAL	X fogás, (lásd <_AMODE> EGYES)		
10	DZ	<_DZ>	REAL	Z fogás, (lásd <_AMODE> EGYES)		
11	UX	<_UX>	REAL	X simítási ráhagyás (lásd <_VARI> TÍZEZRES)		
12	UZ	<_UZ>	REAL	Z simítási ráhagyás (lásd <_VARI> TÍZEZRES)		
13	U	<_U>	REAL	simítási ráhagyás kontúr-párhuzamos (lásd <_VARI>TÍZEZRES)		
14	U1	<_U1>	REAL	kiegészítő simítási ráhagyás simításnál (lásd <_AMODE> EZRES)		
15	BL	<_BL>	INT	nyersdarab definíció	1 =	henger ráhagyással
					2 =	ráhagyás a készdarab-kontúron
					3 =	nyersdarab-kontúr megadva
16	XD	<_XD>	REAL	nyersdarab definíció X, (lásd <_AMODE> SZÁZEZRES)		
17	ZD	<_ZD>	REAL	nyersdarab definíció Z (lásd <_AMODE> EGYMILLIÓS)		
18	XA	<_XA>	REAL	határ 1 X (absz, mindig átmérő)		
19	ZA	<_ZA>	REAL	határ 1 Z (absz)		
20	XB	<_XB>	REAL	határ 2 X, (lásd <_AMODE> TÍZMILLIÓS)		
21	ZB	<_ZB>	REAL	határ 2 Z (lásd <_AMODE> SZÁZMILLIÓS)		
22	XDA	<_XDA>	REAL	beszúrási határ 1 az 1. beszúrási pozícióhoz a homlokoldalon (absz, mindig átmérő)		
23	XDB	<_XDB>	REAL	beszúrási határ 2 az 1. beszúrási pozícióhoz a homlokoldalon (absz, mindig átmérő)		
24	N	<_N>	INT	beszúrások száma		
25	DP	<_DP>	REAL	beszúrások távolsága	hossz-beszúrás: párhuzamosan a Z tengelyhez	
					sík-beszúrás: párhuzamosan az X tengelyhez	
26	DI	<_DI>	REAL	távolság előtolás-megszakításra	0 =	nincs megszakítás
					> 0 =	megszakítással
27	SC	<_SC>	REAL	biztonsági távolság az akadályok megkerüléséhez, növekményes		
28	D2	<_DN>	INT	szerszám 2. vágóélhez, ha nincs programozva ⇒ D+1		

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
29		<_GMODE>	INT	geometria módus (programozott geometria-értékek értékelése)		
				EGYES:	fog-lalt	
				TÍZES:	fog-lalt	
				SZÁZAS:	megmunkálás kiválasztása / csak kezdő-pont kiszámítás	
					0 =	normál megmunkálás (kompatibi-litás módus nem szükséges)
					1 =	normál megmunkálás
					2 =	kezdő-pozíciót kiszámítani - nincs megmunkálás (csak Shop-Mill/ShopTurn-ből felhíváshoz)
				EZRES:	határolás	
					0 =	nem
					1 =	igen
				TÍZEZRES:	határ 1 X-et beadni	
					0 =	nem
					1 =	igen
				SZÁZEZRES:	határ 2 X-et beadni	
					0 =	nem
					1 =	igen
				EGYMILLIÓS:	határ 1 Z-t beadni	
					0 =	nem
					1 =	igen
				TÍZMILLIÓS:	határ 2 Z-t beadni	
					0 =	nem
					1 =	igen

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
30		<_DMODE>	INT	kijelzés módus		
				EGYES:	G17/G18/G19 megmunkálási sík	
					0 =	kompatibilitás, a ciklushívás előtt hatásos sík marad aktív
					1 =	G17 (csak a ciklusban aktív)
					2 =	G18 (csak a ciklusban aktív)
					3 =	G19 (csak a ciklusban aktív)
				TÍZES:	technológia módus	
					1 =	kontúr leforgácsolás
					2 =	kontúr-szúrás
					3 =	leszúró esztergálás
				SZÁZAS:	maradékanyag megmunkálás	
					0 =	nem
					1 =	igen
				EZRES:	---	foglalt
				TÍZEZRES:	Technológia skálázás ciklus maszkokon belül (Oldal 818)	
					0 =	beadás: teljes
					1 =	beadás: egyszerű

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
31		<_AMODE>	INT	alternatív módus		
				EGYES:	fogásvétel választás	
					0 =	fogásvétel DX és DZ kontúr-párhuzamos leforgácsolási módnál
					1 =	fogás D
				TÍZES:	fogásvétel stratégia	
					0 =	változó vágásmélység (90 ... 100 %)
					1 =	állandó vágásmélység
				SZÁZAS:	vágás felosztás	
					0 =	egyenletes
					1 =	élekre beállítani
				EZRES:	U1 kontúr-ráhagyás kiválasztása, dupla simítás	
					0 =	nem
					1 =	igen
				TÍZEZRES:	nyersdarab kiválasztást aktualizálni	
					0 =	nem
					1 =	igen
				SZÁZEZRES:	ráhagyás kiválasztás nyersdarabon XD	
					0 =	abszolút, sík-tengely értéke átmérőben
					1 =	növekményes, sík-tengely értéke sugárban
				EGYMILLIÓS:	ráhagyás kiválasztás nyersdarabon ZD	
					0 =	abszolút
					1 =	növekményes
				TÍZMILLIÓS:	kiválasztás határ 2 XB	
					0 =	abszolút, sík-tengely értéke átmérőben
					1 =	növekményes, sík-tengely értéke sugárban
				SZÁZMILLIÓS:	kiválasztás határ 2 ZB	
					0 =	abszolút
					1 =	növekményes
				EGYMILLIÁRD:		
					0 =	vezető-csatorna
					1 =	követő-csatorna
32		<_PK>	INT	partner csatorna száma, ha a gépen 2-nél több csatorna van		
33	DCH	<_DCH>	REAL	csatorna eltolás		
34	FS	<_FS>	REAL	simítás előtolás teljes megmunkálásnál		

20.1.44 CYCLE4071 - hossz-köszörülés fogással a fordulóponton

Szintaxis

```
CYCLE4071(<S_A>, <S_B>, <S_W>, <S_U>, <S_I>, <S_K>, <S_H>, <S_A1>,
<S_A2>)
```

Paraméter

Nr.	Paraméter	Adattípus	Jelentés
1	<S_A>	REAL	fogás-mélység az elején
2	<S_B>	REAL	fogás-mélység a végén
3	<S_W>	REAL	köszörű szélesség
4	<S_U>	REAL	kiszikrázási idő
5	<S_I>	REAL	fogásvétel előtolás
6	<S_K>	REAL	kereszt-fogásvétel előtolás
7	<S_H>	INT	ismétlések száma
8	<S_A1>	AXIS	fogásvétel-tengely (opcionális) ill. 1. geometria-tengely
9	<S_A2>	AXIS	ingázás-tengely (opcionális) ill. 2. geometria-tengely

Funkció

A ciklus <S_H> fogás megmunkálást szolgálja. Ennek során a fogás-mélység az elején és a végén különböző lehet. A fogások között egy érintőleges mozgás történik.

Lefutás

1. Ciklus Start az inga-tengely aktuális pozíciójában.
2. Fogásvétel tengely mozgatása fogásmélységre az elején P1 <S_A> a P5 <S_I> előtolással a fogásvételhez.
3. Kiszikráztatás P4 <S_U> kiszikráztatás idővel.
4. Ingat-tengely mozgatása P3 <S_W> köszörű szélességgel elmozdulási útként és P6 <S_K> kereszt-fogásvétel előtolással.
5. Fogásvétel tengely mozgatása fogásmélységre az elején P2 <S_B> a P5 <S_I> előtolással a fogásvételhez.
6. Kiszikráztatás P4 <S_U> kiszikráztatás idővel.
7. Ingat-tengely mozgatása P4 <S_A> köszörű szélességgel elmozdulási útként a kezdőpontra és P6 <S_K> kereszt-fogásvétel előtolással.

A lefutás egyes-mondattal nem szakítható meg.

A lefutás a P7 (<S_H>) programozott ismétlés számnak megfelelően lesz ismételve.

Példa

Két ingamozgást végezni a következő ciklus paraméterekkel:

- fogás-mélység az elején: 0,02 mm
- fogás-mélység a végén: 0,01 mm
- löket: 100 mm
- kiszikrázási idő: 1 s
- fogásvétel előtolás: 1 mm/min
- kereszt előtolás: 1000 mm/perc
- ismétlések: 2
- inga- és fogásvétel-tengely: szabványos geometria-tengelyek

Programkód

```
N10 T1 D1
N20 CYCLE4071(0.02,0.01,100,1,1,1000,2)
N30 M30
```

20.1.45 CYCLE4072 - Hossz-köszörülés fogással a fordulóponton és megszakítás jel

Szintaxis

```
CYCLE4072(<S_GAUGE>, <S_A>, <S_B>, <S_W>, <S_U>, <S_I>, <S_K>,
<S_H>, <S_A1>, <S_A2>)
```

Paraméter

Nr.	Paraméter	Adattípus	Jelentés
1	<S_GAUGE>	STRING	fogásvétel megszakítás feltételek: 1. egy gyors bemenet száma 2. logikai kifejezés
2	<S_A>	REAL	fogás-mélység az elején
3	<S_B>	REAL	fogás-mélység a végén
4	<S_W>	REAL	köszörű szélesség
5	<S_U>	REAL	kiszikrázási idő
6	<S_I>	REAL	fogásvétel előtolás
7	<S_K>	REAL	kereszt-fogásvétel előtolás
8	<S_H>	INT	ismétlések száma
9	<S_A1>	AXIS	fogásvétel-tengely (opcionális) ill. 1. geometria-tengely
10	<S_A2>	AXIS	ingázás-tengely (opcionális) ill. 2. geometria-tengely

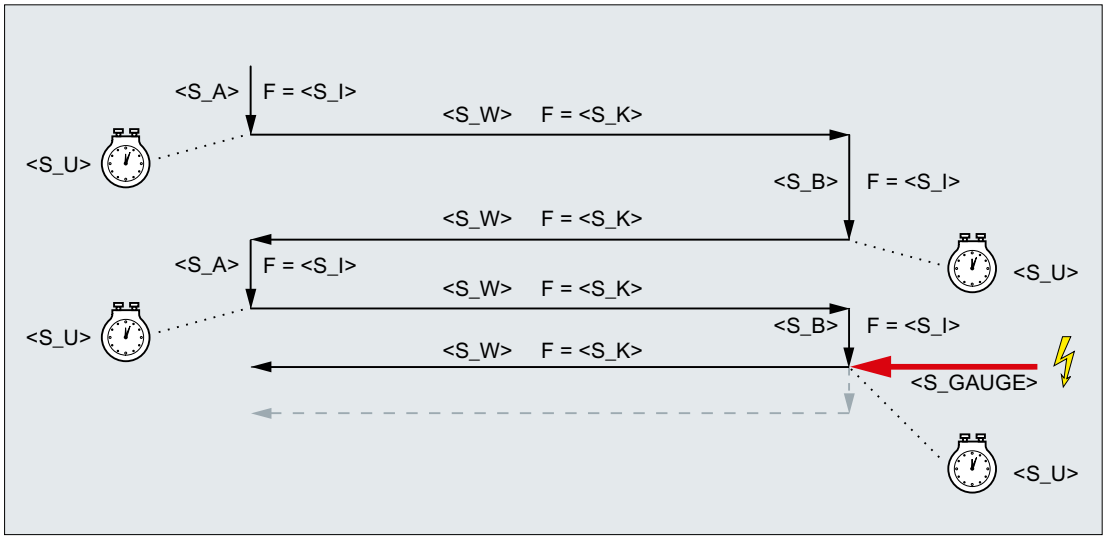
Funkció

A ciklus <S_H> fogás megmunkálást szolgálja a külső megszakítás jel figyelembe vételével. A fogás-mélység az elején és a végén különböző lehet. A fogások között egy érintőleges mozgás történik. A mélységi fogásvétel meg lesz szakítva, ha a megszakítás feltétel teljesül. A mélységi fogásvétel megszakítása után mindig egy teljes löket lesz végrehajtva.

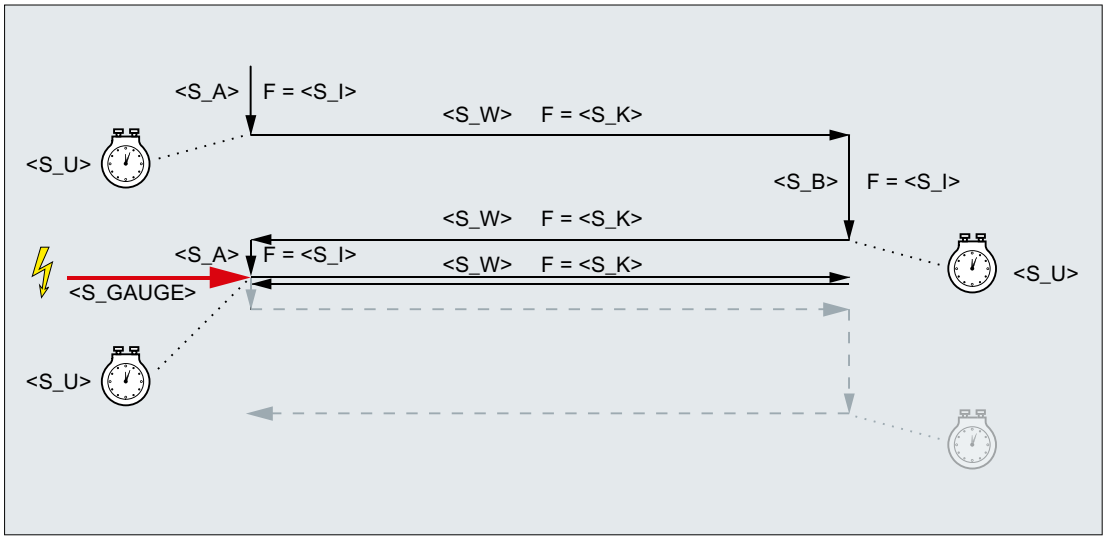
Lefutás

1. Ciklus Start az inga-tengely aktuális pozíciójában.
2. Fogásvétel tengely mozgatása fogásmélységre az elején P2 **<**<S_A>**>** a P6 <S_I> előtolással a fogásvételhez.
3. Kiszikráztatás P5 **<**<S_U>**>** kiszikráztatás idővel.
4. Ingat-tengely mozgatása P4 **<**<S_W>**>** köszörű szélességgel elmozdulási útként és P7 <S_K> kereszt-fogásvétel előtolással.
5. Fogásvétel tengely mozgatása fogásmélységre az elején P3 **<**<S_B>**>** a P6 <S_I> előtolással a fogásvételhez.
6. Kiszikráztatás P5 **<**<S_U>**>** kiszikráztatás idővel.
7. Ingat-tengely mozgatása P4 **<**<S_W>**>** köszörű szélességgel elmozdulási útként a kezdőpontra és P7 <S_K> kereszt-fogásvétel előtolással.
8. Megszakítás nélkül: A fent leírt lefutás a P7 (<S_H>) programozott ismétlés számnak eléréséig lesz ismételve.
   Megszakítással: A megmunkálás a következő kezdőpont elérésénél be lesz fejezve.

A lefutás egyes-mondattal nem szakítható meg.



Kép 20-1 Fogásvétel megszakítása a végén



Kép 20-2 Fogásvétel megszakítása az elején

Erőforrások

A ciklus erőforrásként egy mondatokat átfogó szinkronakciót és egy szinkronakció változót használ. A szinkronakció dinamikusan lesz megállapítva a szinkron-akció sáv szabad sávjából (CUS.DIR - 1 ..., CMA.DIR - 1000 ..., CST.DIR – 1199 ...). Szinkronakció változóként a SYG_IS[1] lesz használva.

Példák

Példa 1: Oszcillálás két lökettel

Ciklus paraméterek:

- fogás-mélység az elején: 0,02 mm
- fogás-mélység a végén: 0,01 mm
- löket: 100 mm
- kiszikrázási idő: 1 s
- fogásvétel előtolás: 1 mm/min
- kereszt előtolás: 1000 mm/perc
- ismétlések: 2
- inga- és fogásvétel-tengely: szabványos geometria-tengelyek

Megszakítás jel: gyors bemenet 1 ($A_IN[1])

Programkód

```
N10 T1 D1
N20 CYCLE4072("1",0.02,0.01,100,1,1,1000,2)
N30 M30
```

Példa 2: Oszcillálás két lökettel

Ciklus paraméterek:

- fogás-mélység az elején: 0,02 mm
- fogás-mélység a végén: 0,01 mm
- löket: 100 mm
- kiszikrázási idő: 1 s
- fogásvétel előtolás: 1 mm/min
- kereszt előtolás: 1000 mm/perc
- ismétlések: 2
- inga- és fogásvétel-tengely: szabványos geometria-tengelyek

Megszakítás jel: $A_DBR[20] változó < 0,01

Programkód

```
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 - Hossz-köszörülés folyamatos fogásvétellel

Szintaxis

```
CYCLE4073(<S_A>, <S_B>, <S_W>, <S_U>, <S_K>, <S_H>, <S_A1>, <S_A2>)
```

Paraméter

Nr.	Paraméter	Adattípus	Jelentés
1	<S_A>	REAL	fogás-mélység az elején
2	<S_B>	REAL	fogás-mélység a végén
3	<S_W>	REAL	köszörű szélesség
4	<S_U>	REAL	kiszikrázási idő
5	<S_K>	REAL	kereszt-fogásvétel előtolás
6	<S_H>	INT	ismétlések száma
7	<S_A1>	AXIS	fogásvétel-tengely (opcionális) ill. 1. geometria-tengely
8	<S_A2>	AXIS	ingázás-tengely (opcionális) ill. 2. geometria-tengely

Funkció

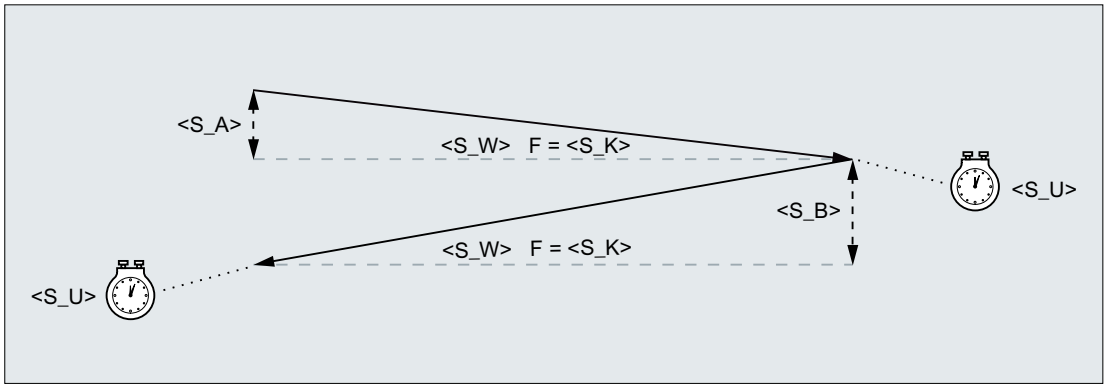
A ciklus <S_H> fogás megmunkálást szolgálja. Ennek során a fogásvétel az elejétől a végéig és a végétől az elejéig eltérő lehet.

Lefutás

1. Ciklus Start az inga-tengely aktuális pozíciójában 0 fogásvétel mélységgel.
2. Ingat-tengely mozgatása P3 <S_W> köszörű szélességgel elmozdulási útként és P5 <S_K> kereszt-fogásvétel előtolással a fogás-mélység folyamatos növelésével a P1 <S_A> fogás-mélység az elején-ig.
3. Kiszikráztatás P4 <S_U> kiszikráztatás idővel.
4. Ingat-tengely mozgatása P3 <S_W> köszörű szélességgel elmozdulási útként a kezdőponton és P5 S_K<S_B> kereszt-fogásvétel előtolással a fogás-mélység folyamatos növelésével a P2 <> fogás-mélység az elején-ig.
5. Kiszikráztatás P4 <S_U> kiszikráztatás idővel.

A lefutás egyes-mondattal nem szakítható meg.

A lefutás a P7 (<S_H>) programozott ismétlés számnak megfelelően lesz ismételve.



Példa

Oszcillálás két lökettel

Ciklus paraméterek:

- fogás-mélység az elején: 0,02 mm
- fogás-mélység a végén: 0,01 mm
- löket: 100 mm
- kiszikrázási idő: 1 s
- kereszt előtolás: 1000 mm/perc
- ismétlések: 2
- inga- és fogásvétel-tengely: szabványos geometria-tengelyek

Programkód

```
N10 T1 D1
N20 CYCLE4073(0.02,0.01,100,1,1000,2)
N30 M30
```

20.1.47 CYCLE4074 - Hossz-köszörülés folyamatos fogásvétellel és megszakítás jellel

Szintaxis

```
CYCLE4074(<S_GAUGE>, <S_A>, <S_B>, <S_W>, <S_U>, <S_K>, <S_H>,
<S_A1>, <S_A2>)
```

Paraméter

Nr.	Paraméter	Adattípus	Jelentés
1	<S_GAUGE>	STRING	fogásvétel megszakítás feltételek: 1. egy gyors bemenet száma 2. logikai kifejezés
2	<S_A>	REAL	fogás-mélység az elején
3	<S_B>	REAL	fogás-mélység a végén
4	<S_W>	REAL	köszörű szélesség
5	<S_U>	REAL	kiszikrázási idő
6	<S_K>	REAL	kereszt-fogásvétel előtolás
7	<S_H>	INT	ismétlések száma
8	<S_A1>	AXIS	fogásvétel-tengely (opcionális) ill. 1. geometria-tengely
9	<S_A2>	AXIS	ingázás-tengely (opcionális) ill. 2. geometria-tengely

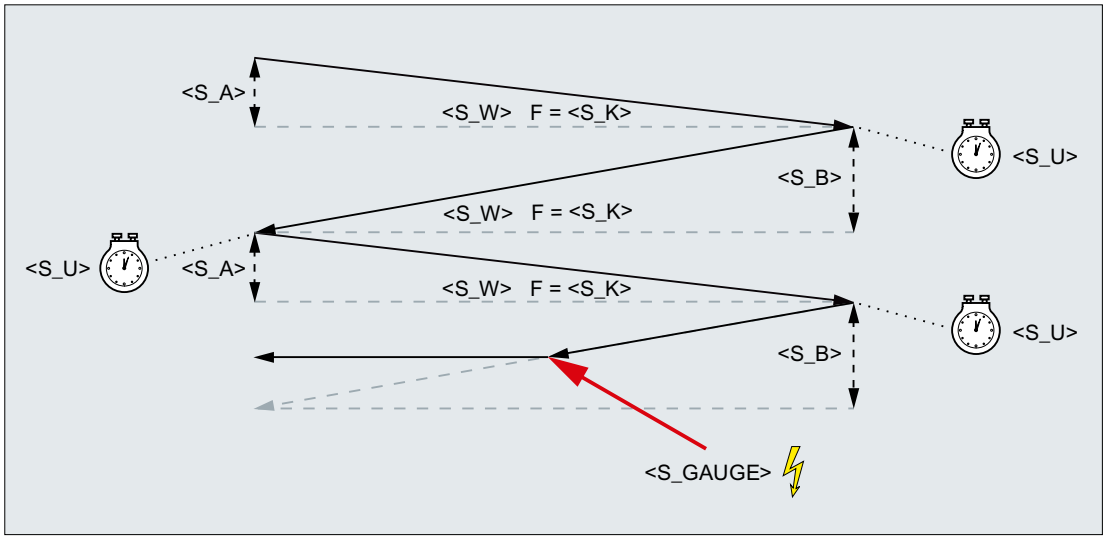
Funkció

A ciklus <S_H> fogás megmunkálást szolgálja pl. egy külső megszakítás jel figyelembe vételével. A fogás-mélység az elején és a végén különböző lehet. A mélységi fogásvétel meg lesz szakítva, ha a megszakítás feltétel teljesül. A mélységi fogásvétel megszakítása után mindig egy teljes löket lesz végrehajtva.

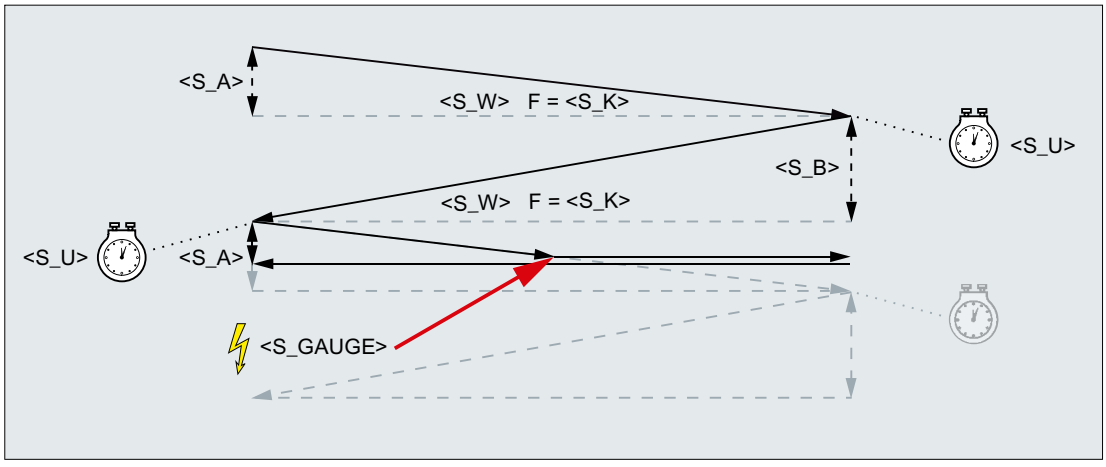
Lefutás

1. Ciklus Start az inga-tengely aktuális pozíciójában 0 fogásvétel mélységgel.
2. Ingat-tengely mozgatása P4 <S_W> köszörű szélességgel elmozdulási útként és P6 <S_K> kereszt-fogásvétel előtolással a fogás-mélység folyamatos növelésével a P2 <S_A> fogás-mélység az elején-ig.
3. Kiszikráztatás P5 <S_U> kiszikráztatás idővel.
4. Ingat-tengely mozgatása P4 <S_W> köszörű szélességgel elmozdulási útként a kezdőponton és P6 S_K<S_B> kereszt-fogásvétel előtolással a fogás-mélység folyamatos növelésével a P3 <> fogás-mélység az elején-ig.
5. Kiszikráztatás P5 <S_U> kiszikráztatás idővel.
6. Megszakítás nélkül: A fent leírt lefutás a P7 (<S_H>) programozott ismétlés számnak eléréséig lesz ismételve.
   Megszakítással: A mélységi fogásvétel meg lesz szakítva. A megmunkálás a következő kezdőpont elérésénél be lesz fejezve.

A lefutás egyes-mondattal nem szakítható meg.



Kép 20-3 Fogásvétel megszakítás a végétől az elejéhez



Kép 20-4 Fogásvétel megszakítás az elejétől a végéhez

Erőforrások

A ciklus erőforrásként egy mondatokat átfogó szinkronakciót és egy szinkronakció változót használ. A szinkronakció dinamikusan lesz megállapítva a szinkron-akció sáv szabad sávjából (CUS.DIR - 1 ..., CMA.DIR - 1000 ..., CST.DIR – 1199 ...). Szinkronakció változóként a SYG_IS[1] lesz használva.

Példák

Példa 1: Oszcillálás két lökettel

Ciklus paraméterek:

- fogás-mélység az elején: 0,02 mm
- fogás-mélység a végén: 0,01 mm

- löket: 100 mm
- kiszikrázási idő: 1 s
- kereszt előtolás: 1000 mm/perc
- ismétlések: 2
- inga- és fogásvétel-tengely: szabványos geometria-tengelyek

Megszakítás jel: gyors bemenet 1 ($A_IN[1])

Programkód

```
N10 T1 D1
N20 CYCLE4074("1",0.02,0.01,100,1,1000,2)
N30 M30
```

Példa 2: Oszcillálás két lökettel

Ciklus paraméterek:

- fogás-mélység az elején: 0,02 mm
- fogás-mélység a végén: 0,01 mm
- löket: 100 mm
- kiszikrázási idő: 1 s
- kereszt előtolás: 1000 mm/perc
- ismétlések: 2
- inga- és fogásvétel-tengely: szabványos geometria-tengelyek

Megszakítás jel: $A_DBR[20] változó < 0,01

Programkód

```
N10 T1 D1
N20 CYCLE4074("($A_DBR[20]<0.01)",0.02,0.01,100,1,1000,2)
N30 M30
```

20.1.48 CYCLE4075 - Sík-köszörülés fogással a fordulóponton

Szintaxis

```
CYCLE4075(<S_I>, <S_J>, <S_K>, <S_A>, <S_R>, <S_F>, <S_P>, <S_A1>,
<S_A2>)
```

Paraméter

Nr.	Paraméter	Adattípus	Jelentés
1	<S_I>	REAL	fogás-mélység az elején
2	<S_J>	REAL	fogás-mélység a végén

Nr.	Paraméter	Adattípus	Jelentés
3	<S_K>	REAL	össz fogás-mélység
4	<S_A>	REAL	köszörű szélesség
5	<S_R>	REAL	fogásvétel előtolás
6	<S_F>	REAL	kereszt-fogásvétel előtolás
7	<S_P>	REAL	kiszikrázási idő
8	<S_A1>	AXIS	fogásvétel-tengely (opcionális)
9	<S_A2>	AXIS	inga-tengely (opcionális)

Funkció

A ciklus P3 <S_K> össz fogás-mélység megmunkálást szolgálja fogás lépésekben. A fogásmélységek az elején P1 <S_I> és végén P2 <S_J> lehetnek eltérőek. A fogások között egy érintőleges mozgás történik.

A P1 ... P4 útadatok lehetnek negatívak vagy pozitívak.

A fogásvételi tengely P8 <S_A1> és/vagy inga-tengely P9 <S_A2> adatok opcionálisak. Ha egyik vagy mindkét paraméter nincs megadva, akkor a ciklus a csatorna első két geometria-tengelyét használja.

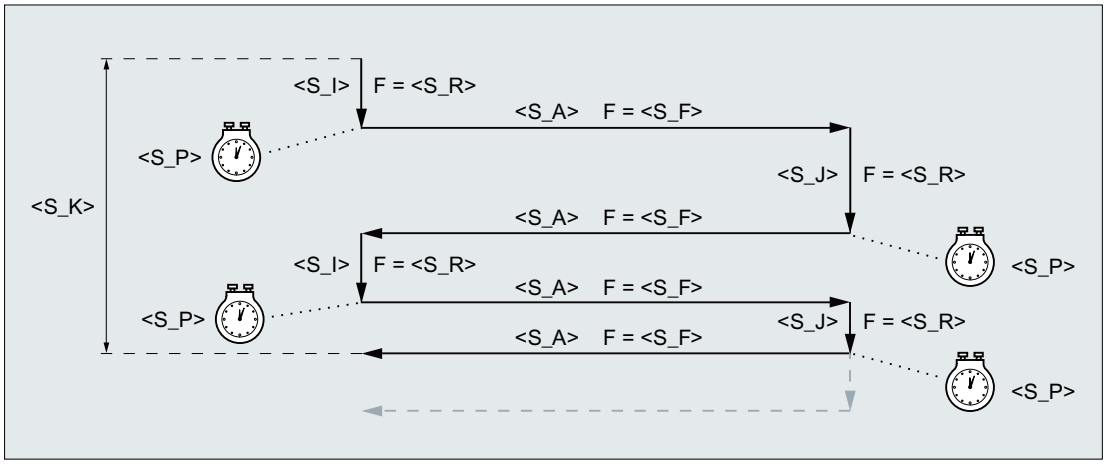
Ha a fogásmélység az elején P1 <S_I> és a végén P2 <S_J> összege 0 ill. az össz fogás-mélység P3 <S_K> 0, csak egy kiszikráztatás löket lesz végrehajtva.

Lefutás

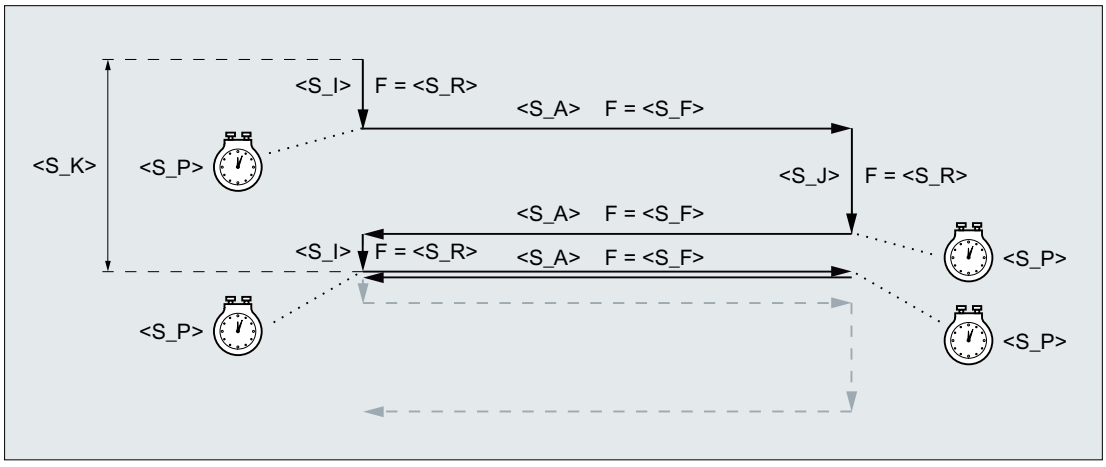
1. Ciklus Start az inga-tengely aktuális pozíciójában.
2. Fogásvétel tengely mozgatása fogásmélységre az elején P1 <S_I> a P5 <S_R> előtolással a fogásvételhez.
3. Kiszikráztatás P7 <S_P> kiszikráztatás idővel.
4. Ingat-tengely mozgatása P4 <S_A> köszörű szélességgel elmozdulási útként és P6 <S_F> kereszt-fogásvétel előtolással.
5. Fogásvétel tengely mozgatása fogásmélységre az elején P2 <S_J> a P5 <S_R> előtolással a fogásvételhez.
6. Kiszikráztatás P7 <S_P> kiszikráztatás idővel.
7. Ingat-tengely mozgatása P4 <S_A> köszörű szélességgel elmozdulási útként a kezdőpontra és P6 <S_F> kereszt-fogásvétel előtolással.

A lefutás egyes-mondattal nem szakítható meg.

A lefutás a P3 <S_K> össz fogás-mélység eléréséig lesz ismételve. Az utolsó löket egyenetlenül lesz felosztva.



Kép 20-5 Az össz fogás-mélységet a fogásvételnél a második fordulópontra érik el.



Kép 20-6 Az össz fogás-mélységet a fogásvételnél az első fordulóponton érik el.

Példa

Oszcillálás:

- 0,02 mm fogás-mélység az elején
- 0,01 mm fogás-mélység a végén
- össz fogás-mélység 1 mm
- löket 100 mm
- előtolás fogásra 1 mm/perc
- kereszt előtolás 1000 mm/perc
- kiszikrázási idő 1 másodperc
- szabványos geometria-tengelyek

Programkód

```
N10 T1 D1
```

Programkód

```
N20 CYCLE4075(0.02,0.01,1,100,1,1000,1)
N30 M30
```

20.1.49 CYCLE4077 - Sík-köszörülés fogással a fordulóponton és megszakítás jel

Szintaxis

```
CYCLE4077(<S_GAUGE>, <S_I>, <S_J>, <S_K>, <S_A>, <S_R>, <S_F>,
<S_P>, <S_A1>, <S_A2>)
```

Paraméter

Nr.	Paraméter	Adattípus	Jelentés
1	<S_GAUGE>	STRING	fogásvétel megszakítás feltétel: • egy gyors bemenet száma • logikai kifejezés
2	<S_I>	REAL	fogás-mélység az elején
3	<S_J>	REAL	fogás-mélység a végén
4	<S_K>	REAL	össz fogás-mélység
5	<S_A>	REAL	köszörű szélesség
6	<S_R>	REAL	fogásvétel előtolás
7	<S_F>	REAL	kereszt-fogásvétel előtolás
8	<S_P>	REAL	kiszikrázási idő
9	<S_A1>	AXIS	fogásvétel-tengely (opcionális)
10	<S_A2>	AXIS	inga-tengely (opcionális)

Funkció

A ciklus P4 <S_K> össz fogás-mélység megmunkálást szolgálja fogás lépésekben. A fogásmélységek az elején P2 <S_I> és végén P3 <S_J> lehetnek eltérőek. A fogások között egy érintőleges mozgás történik. A mélységi fogásvétel meg lesz szakítva, ha a gyors bemenet megszakítás jel 1 vagy ha a megszakítás feltétel teljesül. A megszakítás után mindig egy teljes löket lesz végrehajtva.

A P2 ... P5 útadatok lehetnek negatívak vagy pozitívak.

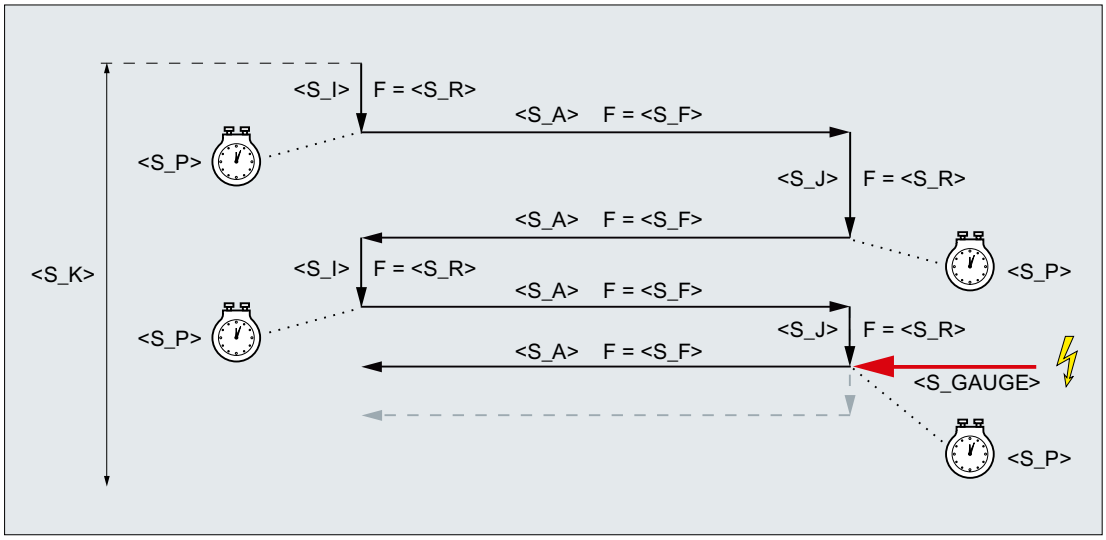
A fogásvételi tengely P9 <S_A1> és/vagy inga-tengely P10 <S_A2> adatok opcionálisak. Ha egyik vagy mindkét paraméter nincs megadva, akkor a ciklus a csatorna első két geometria-tengelyét használja.

Ha a fogásmélység az elején P2 <S_I> és a végén P3 <S_J> összege 0 ill. az össz fogás-mélység P4 <S_K> 0, csak egy kiszikráztatás löket lesz végrehajtva.

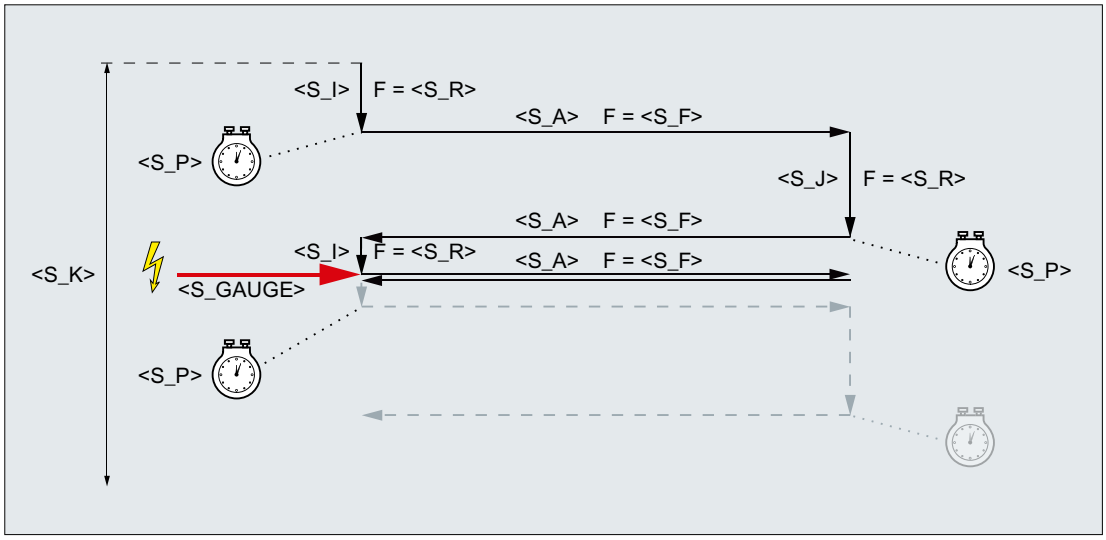
Lefutás

1. Ciklus Start az inga-tengely aktuális pozíciójában.
2. Fogásvétel tengely mozgatása fogásmélységre az elején P2 <S_I> a P6 <S_R> előtolással a fogásvételhez.
3. Kiszikráztatás P8 <S_P> kiszikráztatás idővel.
4. Ingat-tengely mozgatása P5 <S_A> köszörű szélességgel elmozdulási útként és P7 <S_F> kereszt-fogásvétel előtolással.
5. Fogásvétel tengely mozgatása fogásmélységre az elején P3 <S_J> a P6 <S_R> előtolással a fogásvételhez.
6. Kiszikráztatás P8 <S_P> kiszikráztatás idővel.
7. Ingat-tengely mozgatása P5 <S_A> köszörű szélességgel elmozdulási útként a kezdőpontra és P7 <S_F> kereszt-fogásvétel előtolással.
8. Megszakítás nélkül: A fent leírt lefutás a P4 <S_K> össz fogás-mélység eléréséig lesz ismételve. Az utolsó löket egyenetlenül lesz felosztva.
   Megszakítással: A megmunkálás a kezdőponton lesz befejezve.

A lefutás egyes-mondattal nem szakítható meg.



Kép 20-7 Fogásvétel megszakítása a végén



Kép 20-8 Fogásvétel megszakítása az elején

Erőforrások

A ciklus erőforrásként egy mondatokat átfogó szinkronakciót és egy szinkronakció változót használ. A szinkronakció dinamikusan lesz megállapítva a szinkron-akció sáv szabad sávjából (CUS.DIR - 1 ..., CMA.DIR - 1000 ..., CST.DIR – 1199 ...). Szinkronakció változóként a SYG_IS[1] lesz használva.

Példák

Példa 1

Oszcillálás:

- 0,02 mm fogás-mélység az elején
- 0,01 mm fogás-mélység a végén
- össz fogás-mélység 1 mm
- löket 100 mm
- előtolás fogásra 1 mm/perc
- kereszt előtolás 1000 mm/perc
- kiszikrázási idő 1 másodperc
- szabványos geometria-tengelyek

Megszakítás jel: gyors bemenet 1 ($A_IN[1])

Programkód

```
N10 T1 D1
N20 CYCLE4077("1",0.02,0.01,1,100,1,1000,1)
N30 M30
```

Példa 2

Oszcillálás:

- 0,02 mm fogás-mélység az elején
- 0,01 mm fogás-mélység a végén
- össz fogás-mélység 1 mm
- löket 100 mm
- előtolás fogásra 1 mm/perc
- kereszt előtolás 1000 mm/perc
- kiszikrázási idő 1 másodperc
- szabványos geometria-tengelyek

Megszakítás jel: Dualport-RAM változó 20 kisebb mint 0,01 ($A_DBR[20] < 0,01)

Programkód

```
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 - Sík-köszörülés folyamatos fogásvétellel

Szintaxis

```
CYCLE4078(<S_I>, <S_J>, <S_K>, <S_A>, <S_F>, <S_P>, <S_A1>, <S_A2>)
```

Paraméter

Nr.	Paraméter	Adattípus	Jelentés
1	<S_I>	REAL	fogásvétel az elejétől a végéhez
2	<S_J>	REAL	fogásvétel a végétől az elejéhez
3	<S_K>	REAL	össz fogás-mélység
4	<S_A>	REAL	köszörű szélesség
5	<S_F>	REAL	előtolás
6	<S_P>	REAL	kiszikrázási idő
7	<S_A1>	AXIS	fogásvétel-tengely (opcionális)
8	<S_A2>	AXIS	inga-tengely (opcionális)

Funkció

A ciklus P3 <S_K> össz fogás-mélység megmunkálást szolgálja folyamatos fogásvétellel. A fogásmélységek az elejétől a végéhez P1 <S_I> és végétől az elejéhez P2 <S_J> lehetnek eltérőek.

A P1 ... P4 útadatok lehetnek negatívak vagy pozitívak.

A fogásvételi tengely P8 <S_A1> és/vagy inga-tengely P9 <S_A2> adatok opcionálisak. Ha egyik vagy mindkét paraméter nincs megadva, akkor a ciklus a csatorna első két geometria-tengelyét használja.

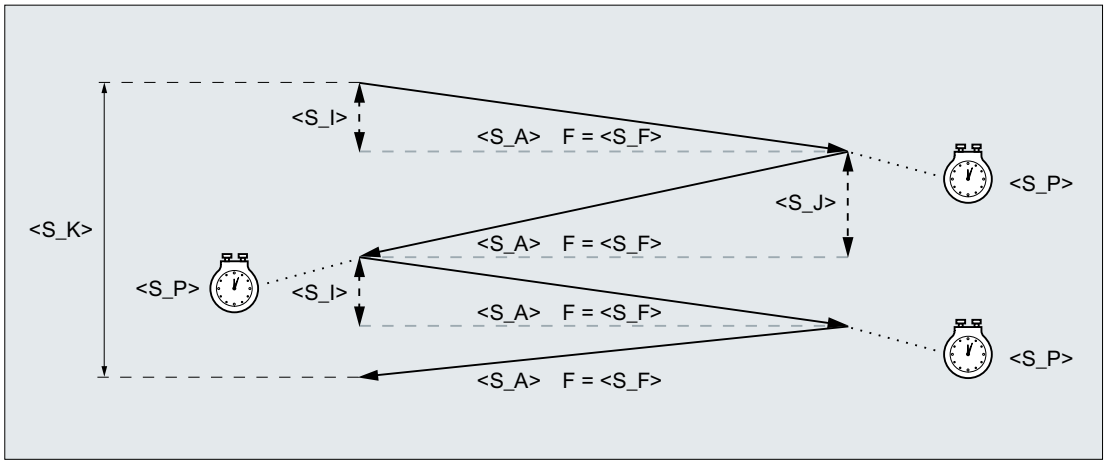
Ha a fogásmélység P1 <S_I> és P2 <S_J> összege 0 ill. az össz fogás-mélység P3 <S_K> 0, csak egy kiszikráztatás löket lesz végrehajtva.

Lefutás

1. Ciklus Start az inga-tengely aktuális pozíciójában 0 fogásvétel mélységgel.
2. Ingat-tengely mozgatása P4 <S_A> köszörű szélességgel elmozdulási útként és P5 <S_F> kereszt-fogásvétel előtolással a fogás-mélység folyamatos növelésével a P1 <S_I> fogás-mélység az elején-ig.
3. Kiszikráztatás P7 <S_P> kiszikráztatás idővel.
4. Ingat-tengely mozgatása P4 <S_A> köszörű szélességgel elmozdulási útként a kezdőponton és P5 S_F<S_J> előtolással a fogás-mélység folyamatos növelésével a P2 <> fogás-mélység az végén-ig.
5. Kiszikráztatás P7 <S_P> kiszikráztatás idővel.
6. Ingat-tengely mozgatása P4 <S_A> köszörű szélességgel elmozdulási útként a kezdőpontra és P5 <S_F> előtolással.

A lefutás egyes-mondattal nem szakítható meg.

A lefutás a P3 <S_K> össz fogás-mélység eléréséig lesz ismételve. Az utolsó löket egyenetlenül lesz felosztva.



Példa

Oszcillálás:

- 20 mm fogás-mélység az elején
- 10 mm fogás-mélység az elején
- össz fogás-mélység 100 mm
- löket 100 mm

- előtolás 1000 mm/perc
- kiszikrázási idő 1 másodperc
- szabványos geometria-tengelyek

Programkód

```
N10 T1 D1
N20 CYCLE4078(20,10,100,100,1000,1)
N30 M30
```

20.1.51 CYCLE4079 - Sík-köszörülés közbenső fogásvétellel

Szintaxis

```
CYCLE4079(<S_I>, <S_J>, <S_K>, <S_A>, <S_R>, <S_F>, <S_P>, <S_A1>,
<S_A2>)
```

Paraméter

Nr.	Paraméter	Adattípus	Jelentés
1	<S_I>	REAL	fogás-mélység az elején
2	<S_J>	REAL	fogás-mélység a végén
3	<S_K>	REAL	össz fogás-mélység
4	<S_A>	REAL	köszörű szélesség
5	<S_R>	REAL	fogásvétel előtolás
6	<S_F>	REAL	kereszt-fogásvétel előtolás
7	<S_P>	REAL	kiszikrázási idő
8	<S_A1>	AXIS	fogásvétel-tengely (opcionális)
9	<S_A2>	AXIS	inga-tengely (opcionális)

Funkció

A ciklus P3 <S_K> össz fogás-mélység megmunkálást szolgálja fogás lépésekben. A fogásmélységek az elején P1 <S_I> és végén P2 <S_J> lehetnek eltérőek. A fogások között egy érintőleges mozgás történik.

A P1 ... P4 útadatok lehetnek negatívak vagy pozitívak.

A fogásvételi tengely P8 <S_A1> és/vagy inga-tengely P9 <S_A2> adatok opcionálisak. Ha egyik vagy mindkét paraméter nincs megadva, akkor a ciklus a csatorna első két geometria-tengelyét használja.

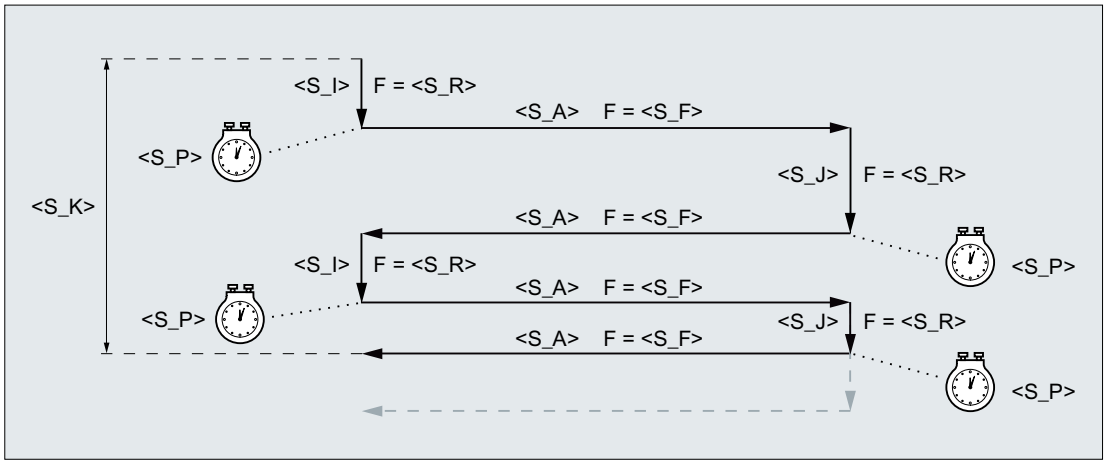
Ha a fogásmélység az elején P1 <S_I> és a végén P2 <S_J> összege 0 ill. az össz fogás-mélység P3 <S_K> 0, csak egy kiszikráztatás löket lesz végrehajtva.

Lefutás

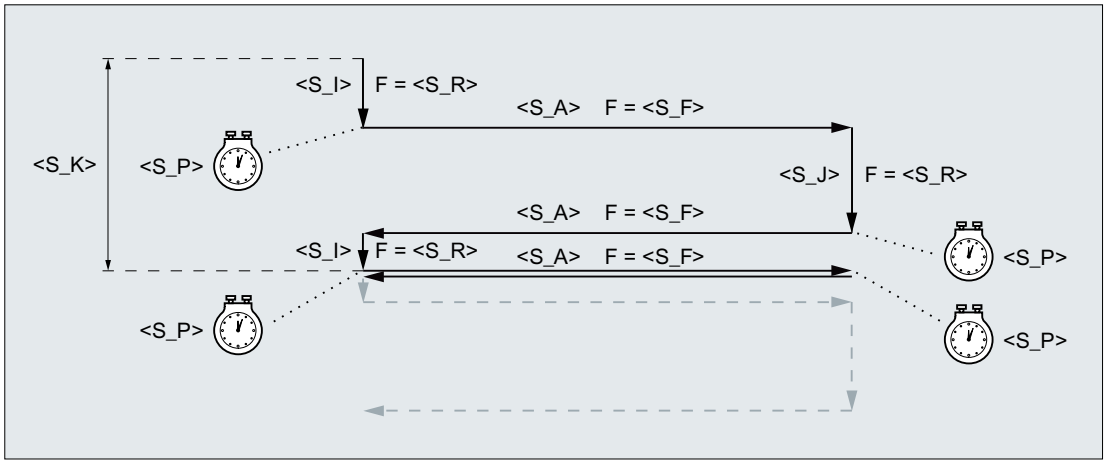
1. Ciklus Start az inga-tengely aktuális pozíciójában.
2. Fogásvétel tengely mozgatása fogásmélységre az elején P1 <S_I> a P5 <S_R> előtolással a fogásvételhez.
3. Kiszikráztatás P7 <S_P> kiszikráztatás idővel.
4. Ingat-tengely mozgatása P4 <S_A> köszörű szélességgel elmozdulási útként és P6 <S_F> kereszt-fogásvétel előtolással.
5. Fogásvétel tengely mozgatása fogásmélységre az elején P2 <S_J> a P5 <S_R> előtolással a fogásvételhez.
6. Kiszikráztatás P7 <S_P> kiszikráztatás idővel.
7. Ingat-tengely mozgatása P4 <S_A> köszörű szélességgel elmozdulási útként a kezdőpontra és P6 <S_F> kereszt-fogásvétel előtolással.

A lefutás egyes-mondattal nem szakítható meg.

A lefutás a P3 <S_K> össz fogás-mélység eléréséig lesz ismételve. Az utolsó löket egyenetlenül lesz felosztva.



Kép 20-9 Az össz fogás-mélységet a fogásvételnél a második fordulópontra érik el.



Kép 20-10 Az össz fogás-mélységet a fogásvételnél az első fordulóponton érik el.

Példa

Oszcillálás:

- 0,02 mm fogás-mélység az elején
- 0,01 mm fogás-mélység a végén
- össz fogás-mélység 1 mm
- löket 100 mm
- előtolás fogásra 1 mm/perc
- kereszt előtolás 1000 mm/perc
- kiszikrázási idő 1 másodperc
- szabványos geometria-tengelyek

Programkód

```
N10 T1 D1
N20 CYCLE4079(0.02,0.01,1,100,1,1000,1)
N30 M30
```

20.1.52 GROUP_BEGIN - programblokk kezdete

Szintaxis

```
GROUP_BEGIN(<_LEVEL>, <_NAME>, <_SP>, <_MODE>, <S_ICON>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés		
1		<_LEVEL>	INT	sík		
				0 =	fő-szint	
				1 =	1. al-szint	
2		<_NAME>	STRING[128]	blokk név		
3		<_SP>	INT	orsó		
				0 =	nincs orsó	
				1 =	főorsó	
				2 =	ellenorsó	
4		<_MODE>	INT	módus		
				bit 0	= 1	GROUP_ADDEND van
				bit 1	= 1	ShopTurn: automatikus visszahúzás (szerszám-csere-pontra menni)
				bit 12	foglalt	
				bit 13	foglalt	
5		<S_ICON>	STRING[32]	ikon neve (csak kezelőfelülethez)		

20.1.53 GROUP_END - programblokk vége

Szintaxis

```
GROUP_END(<_LEVEL>, <_SP>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés	
1		<_LEVEL>	INT	sík	
				0 =	fő-szint
				1 =	1. al-szint
2		<_SP>	INT	orsó	
				0 =	nincs orsó
				1 =	főorsó
				2 =	ellenorsó

20.1.54 GROUP_ADDEND - bejáratos kiegészítés vége

Szintaxis

```
GROUP_ADDEND(<_LEVEL>, <_SP>)
```

Paraméter

Nr.	Paraméter maszk	Paraméter belül	Adattípus	Jelentés	
1		<_LEVEL>	INT	sík	
				0 =	fő-szint
				1 =	1. al-szint
2		<_SP>	INT	orsó	
				0 =	nincs orsó
				1 =	főorsó
				2 =	ellenorsó

20.1.55 Peremfeltételek

20.1.55.1 Technológia skálázás ciklus maszkokon belül

Ha a technológia skálázás aktív, a különböző ciklus maszkoknál lehet választani az egyszerűsített beadást, amelynél csak a lényeges ciklus paraméterek lesznek kijelezve.

A következő ciklus maszkoknál lehet pl. az egyszerűsített beadást választani:

Technológia	Ciklus maszk
fúrás	mélylyuk fúrása
	menetfúrás
marás	négyszög-zseb
	kontúrmarás: zseb
esztergálás	menet esztergálás: hossz
	kontúresztergálás: leforgácsolás
	kontúresztergálás: leszúrás
	kontúresztergálás: leszúró esztergálás

Az érintett ciklus maszkoknál a kezelő felületen fennáll a választási lehetőség "beadás: **egyszerű**" és "beadás: **teljes**".

Nem kijelzett ciklus paraméterek

A leegyszerűsített beadásnál nem kijelzett ciklus paraméterek vagy fix, technológiailag értelmes, de nem változtatható értékekkel vannak előre kitöltve. vagy a ciklus paraméterekhez csatorna-specifikus ciklus beállítási-adatokkal lehet paraméterezhető értékeket hozzárendelni. Lásd az "Üzembehelyezés" > "Csatorna-specifikus ciklus beállítási-adatok" bekezdésben

Átkapcsolás "beadás: teljes" > "beadás: egyszerű"

Ha egy ciklus maszk a "beadás teljes" beállítással lesz kitöltve és utána "beadás egyszerű"-re lesz átkapcsolva, a ciklus felhívás generálásánál az akkor már nem kijelezett paraméterekhez az alap vagy beállítási adatok lesznek használva.

Üzembehelyezés

Csatorna-specifikus konfigurációs gépadatok

A gépadatokkal a Technológia skálázás lehetőség ciklus maszkokon belül lesz aktiválva:

MD52210 $MCS_FUNCTION_MASK_DISP, Bit 9 = 1 ("beadás egyszerű" választást kijelezni)

Csatorna-specifikus ciklus beállítási-adatok

Ha az egyszerű beadás a ciklus maszkon belül aktív, bizonyos ciklus paraméterek értékeit a következő beállítási adatokkal lehet előre megadni:

Szám	Jelölő	Jelentés
SD55300	$SCS_EASY_SAFETY_CLEARANCE	biztonsági távolság
SD55301	$SCS_EASY_DWELL_TIME	várakozási idő
SD55305	$SCS_EASY_DRILL_DEEP_FD1	mélylyuk fúrása: százalék: 1. előtolás
SD55306	$SCS_EASY_DRILL_DEEP_DF	mélylyuk fúrása: százalék: fogás
SD55307	$SCS_EASY_DRILL_DEEP_V1	mélylyuk fúrása: minimális mélységi fogás
SD55308	$SCS_EASY_DRILL_DEEP_V2	mélylyuk fúrása: visszahúzási érték
SD55309	$SCS_EASY_THREAD_RETURN_DIST	menet esztergálás: visszafutási távolság

20.2 Mérőciklusok

A mérőciklusok speciálisan a Siemens által rendelkezésre bocsájtott alprogramok megadott mérési feladatok megoldására. Amint a ciklusoknál általánosan szokás, a mérőciklusokat is lehet paraméterekkel a konkrét problémához illeszteni.

A mérőciklusok a mérésekhez a következő tartományokban és technológiáknál állnak rendelkezésre:

- szerszám mérések esztergálásnál / marásnál
- munkadarab mérések esztergálásnál / marásnál

Irodalom

A mérőciklusok részletes leírás itt található:

Mérő ciklusok programozási kézikönyv

Táblázatok

21

21.1 utasítások

Megjegyzés

Ciklusok

Az utasítások listája tartalmazza az összes ciklust, amelyek az NC programban előfordulhatnak (G utasítás), azaz a program szerkesztőben a maszkokkal programozhatóak ill. a köszörülésnél a programozási támogatás nélkül programozhatók kell legyenek. Nem lettek figyelembe véve azok a ciklusok, amelyek kompatibilitási okokból még megvannak, de a SINUMERIK Operate program szerkesztőjével már nem szerkeszthetőek ("kompatibilitás ciklusok").

Utasítások A ... C

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
:	O	NC főmondatszám, ugrásjelző lezárás, láncolás operátor		+		PGAsl
*	O	Szorzás műveleti jel		+		PGAsl
+	O	Összeadás műveleti jel		+		PGAsl
-	O	Osztás műveleti jel		+		PGAsl
<	O	Összehasonlítás műveleti jel, kisebb		+		PGAsl
<<	O	Láncolás műveleti jel stringekhez		+		PGAsl
<=	O	Összehasonlítás műveleti jel, kisebb egyenlő		+		PGAsl
=	O	Hozzárendelés műveleti jel		+		PGAsl
>=	O	Összehasonlítás műveleti jel, nagyobb egyenlő		+		PGAsl
/	O	Osztás műveleti jel		+		PGAsl
/0 … … /7		Mondat kihagyása (1. kihagyási szint)° ... ... mondat ki lesz hagyva (8. kihagyási szint)		+		PGsl
A	A	Tengelynév	m/s	+		PGAsl
A2	A	Szerszámtájolás: RPY- vagy Euler-szög	s	+		PGAsl
A3	A	Szerszámtájolás: 1. komponense az irányvektornak	s	+		PGAsl
A4	A	Szerszámtájolás: 1. komponense a felületnormálvektornak a mondat elején	s	+		PGAsl
A5	A	Szerszámtájolás: 1. komponense a felületnormálvektornak a mondat végén	s	+		PGAsl

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
A6	A	Szerszámtájolás: 1. komponense az irányvektornak a kúp forgástengelyéhez	s	+		PGAsl
A7	A	Szerszámtájolás: 1. vektor-komponens a közbenső tájoláshoz a kúppalást felületen	s	+		PGAsl
ABS	F	Abszolútérték (érték)		+	+	PGAsl
AC	K	Koordináták/pozíciók abszolút méretmegadás	s	+		PGsl
ACC	K	Aktuális tengely-gyorsítás befolyásolása	m	+	+	PGsl
ACCLIMA	K	Aktuális maximális tengely-gyorsítás befolyásolása	m	+	+	PGAsl
ACN	K	Abszolút méretmegadás körtengelyekre, pozícióra rámenet negatív irányban	s	+		PGsl
ACOS	F	Arcus-Cosinus (trigon. függvény)		+	+	PGAsl
ACP	K	Abszolút méretmegadás körtengelyekre, pozícióra rámenet pozitív irányban	s	+		PGsl
ACTBLOCNO	P	Egy vészjelzés-mondat aktuális mondatszámának kiadása, még ha "aktuális mondatkijelzés elnyomva" (DISPLOF) aktív is!		+		PGAsl
ADDFRAME	F	Egy mért frame beszámítása és esetleg aktiválása		+	-	PGAsl, FB1sl (K2)
ADIS	A	Átsimítási távolság a G1, G2, G3, ... pályafunkciókra	m	+		PGsl
ADISPOS	A	Átsimítási távolság a G0 gyorsmenetre	m	+		PGsl
ADISPOSA	P	Tűrésablak nagysága IPOBRKA-hoz	m	+	+	PGAsl
ALF	A	Gyorsleemelési szög	m	+		PGAsl
AMIRROR	G	Programozható tükrözés	s	+		PGsl
AND	K	Logikai ÉS		+		PGAsl
ANG	A	Kontúrvonal-szög	s	+		PGsl
AP	A	Polárszög	m/s	+		PGsl
APR	K	Hozzáférési védelem fokát olvasni / kijelezni		+		PGAsl
APRB	K	Hozzáférési jogot olvasni, BTSS		+		PGAsl
APRP	K	Hozzáférési jogot olvasni, munkadarabprogram		+		PGAsl
APW	K	Hozzáférési jogot írni		+		PGAsl
APWB	K	Hozzáférési jogot írni, BTSS		+		PGAsl
APWP	K	Hozzáférési jogot írni, munkadarabprogram		+		PGAsl
APX	K	Megadott nyelvi elem végrehajtásának hozzáférési jogát definiálni		+		PGAsl
AR	A	Nyílásszög	m/s	+		PGsl
AROT	G	Programozható forgatás	s	+		PGsl
AROTS	G	Programozható frame-forgatás térszöggel	s	+		PGsl
AS	K	Makró-definíció		+		PGAsl
ASCALE	G	Programozható skálázás	s	+		PGsl
ASIN	F	Arcussinus függvény		+	+	PGAsl

Utasítás	Fajta 1)	Jelentés	W 2)	TP 3)	SA 4)	Leírást lásd 5)
1) 2) 3) 4) 5) Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
ASPLINE	G	Akima-Spline	m	+		PGAsl
ATAN2	F	Arcus-Tangens2		+	+	PGAsl
ATOL	A	Kompresszor funkciók, tájolás-simítások és átsimítási módok tengely-specifikus tűrése	m	+		PGAsl
ATRANS	G	Hozzáadódóan programozható nullapontelto-lás	s	+		PGsl
AUXFUDEL	P	Segédfunkciót csatorna-specifikusan törölni a globális listából		+	-	FB1sl (H2)
AUXFUDELG	P	Egy segédfunkció-csoport összes segédfunk-cióját csatorna-specifikusan törölni a globális listából		+	-	FB1sl (H2)
AUXFUMSEQ	P	M segédfunkciók kiadási sorrendjét megálla-pítani		+	-	FB1sl (H2)
AUXFUSYNC	P	A segédfunkciók globális listájából egy teljes munkadarabprogram mondatot generálni stringként a csatorna-specifikus SERUPRO-Vége-ASUP számára.		+	-	FB1sl (H2)
AX	K	Változó tengelyjelölő	m/s	+		PGAsl
AXCTSWE	P	Konténer tengely forgatás		+	-	PGAsl
AXCTSWEC	P	Tengely-konténer forgatás engedélyt vissza-venni		+	+	PGAsl
AXCTSWED	P	Tengely-konténer forgatás (utasítás változat üzembehelyezéshez)		+	-	PGAsl
AXIS	K	Tengelyjelölő, tengelycím		+		PGAsl
AXNAME	F	Bemeneti stringet átalakít tengelyjelölőre		+	-	PGAsl
AXSTRING	F	Stringet orsószámra alakít		+	-	PGAsl
AXTOCHAN	P	Tengelyt egy adott csatornához igényelni NC programból és szinkron-akcióból lehetséges.		+	+	PGAsl
AXTOSPI	F	Tengely-jelölőt átalakít orsóindexre		+	-	PGAsl
B	A	Tengelynév	m/s	+		PGAsl
B2	A	Szerszámtájolás: RPY- vagy Euler-szög	s	+		PGAsl
B3	A	Szerszámtájolás: . komponense az irányvek-tornak	s	+		PGAsl
B4	A	Szerszámtájolás: 2. komponense a felület-normálvektornak a mondat elején	s	+		PGAsl
B5	A	Szerszámtájolás: 2. komponense a felület-normálvektornak a mondat végén	s	+		PGAsl
B6	A	Szerszámtájolás: 2. komponense az irányvek-tornak a kúp forgástengelyéhez	s	+		PGAsl
B7	A	Szerszámtájolás: 2. vektor-komponens a köz-benső tájoláshoz a kúppalást felületen	s	+		PGAsl
B_AND	O	Bitenként ÉS		+		PGAsl
B_OR	O	Bitenként VAGY		+		PGAsl
B_NOT	O	Bitenként negálás		+		PGAsl
B_XOR	O	Bitenként kizáró-VAGY		+		PGAsl

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
BAUTO	G	Az első Spline-rész definiálása a következő 3 ponton keresztül	m	+		PGAsl
BLOCK	K	A feldolgozandó programrész definíciója a TO kulcsszóval együtt egy közvetett alprogram-hívásban		+		PGAsl
BLSYNC	K	Az interrupt-rutin feldolgozása csak a következő mondatváltás után kezdődjön		+		PGAsl
BNAT [6]	G	Természetes átmenet az első Spline-mondathoz	m	+		PGAsl
BOOL	K	adattípus: igazságérték TRUE / FALSE ill. 1 / 0		+		PGAsl
BOUND	F	Megvizsgálja, hogy az érték a definiált tartományon belül van-e. Egyenlőségnél a vizsgált értéket adja vissza.		+	+	PGAsl
BRISK [6]	G	Ugrásszerű pályagyorsítás	m	+		PGAsl
BRISKA	P	Ugrásszerű pályagyorsítás bekapcsolása a programozott tengelyekre		+	-	PGAsl
BSPLINE	G	B-Spline	m	+		PGAsl
BTAN	G	Érintőleges átmenet az első Spline-mondathoz	m	+		PGAsl
C	A	Tengelynév	m/s	+		PGAsl
C2	A	Szerszámtájolás: RPY- vagy Euler-szög	s	+		PGAsl
C3	A	Szerszámtájolás: 3. komponense az irányvektornak	s	+		PGAsl
C4	A	Szerszámtájolás: 3. komponense a felületnormálvektornak a mondat elején	s	+		PGAsl
C5	A	Szerszámtájolás: 3. komponense a felületnormálvektornak a mondat végén	s	+		PGAsl
C6	A	Szerszámtájolás: 3. komponense az irányvektornak a kúp forgástengelyéhez	s	+		PGAsl
C7	A	Szerszámtájolás: 3. vektor-komponens a közbenső tájoláshoz a kúppalást felületen	s	+		PGAsl
CAC	K	Mozgás egy abszolút pozícióra		+		PGAsl
CACN	K	A táblázatban megadott abszolút értékre mozgás negatív irányban		+		PGAsl
CACP	K	A táblázatban megadott abszolút értékre mozgás pozitív irányban		+		PGAsl
CALCDAT	F	Egy kör sugarát és középpontját számítja ki 3 vagy 4 pontjából		+	-	PGAsl
CALCPOSI	F	Védőtartomány-sértés, munkatér-határolás és szoftver-végállások vizsgálata		+	-	PGAsl
CALL	K	Közvetett alprogram hívás		+		PGAsl
CALLPATH	P	Programozható keresőág alprogramhívásnál		+	-	PGAsl
CANCEL	P	Modális szinkronakció megszakítása		+	-	FBSYsl
CASE	K	Feltételes program elágazás		+		PGAsl

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
CDC	K	Közvetlen mozgás egy pozícióra		+		PGAsl
CDOF [6]	G	Ütközés-felügyelet kikapcsolása	m	+		PGsl
CDOF2	G	Ütközés-felügyelet kikapcsolása, 3D kerületi marásnál	m	+		PGsl
CDON	G	Ütközés-felügyelet bekapcsolása	m	+		PGsl
CFC [6]	G	Állandó előtolás a kontúron	m	+		PGsl
CFIN	G	Állandó előtolás csak belső görbületnél, nem külső görbületnél	m	+		PGsl
CFINE	F	Finom-eltolás hozzárendelés egy FRAME változóhoz		+	-	PGAsl
CFTCP	G	Állandó előtolás a szerszám vágóél vonatkoztatási ponton, középpont-pálya	m	+		PGsl
CHAN	K	Adatok érvényességi tartományának megadása		+		PGAsl
CHANDATA	P	Csatornaszám beállítása a csatorna-adat hozzáféréshez		+	-	PGAsl
CHAR	K	adattípus: ASCII karakter		+		PGAsl
CHF	A	Letörés; érték = letörés hossza	s	+		PGsl
CHKDM	F	Egyértelműség vizsgálata egy táron belül		+	-	FBWsl
CHKDNO	F	D-számok egyértelműség vizsgálata		+	-	PGAsl
CHR	A	Letörés; érték = letörés szélessége mozgásirányban		+		PGsl
CIC	K	Növekményes mozgás egy pozícióra		+		PGAsl
CIP	G	Kör-interpoláció közbenső-ponton keresztül	m	+		PGsl
CLEARM	P	Csatorna-koordinálásnál egy vagy több jelölő törlése		+	+	PGAsl
CLRINT	P	Interrupt kikapcsolás		+	-	PGAsl
CMIRROR	F	Tükrözés egy koordinátatengelyen		+	-	PGAsl
COARSEA	K	Mozgás vége "Pontos-állj durva" eléréséné l	m	+		PGAsl
COLLPAIR	F	Egy ütközés pár összetartozásának vizsgálata		+		PGAsl
COMPCAD	G	COMPCAD kompresszor funkció bekapcsolása	m	+		PGAsl
COMPCURV	G	COMPCURV kompresszor funkció bekapcsolása	m	+		PGAsl
COMPLETE		Vezérlési utasítás adatok ki- és beolvasására		+		PGAsl
COMPOF [6]	G	NC-mondat kompresszor kikapcsolása	m	+		PGAsl
COMPON	G	COMPON kompresszor funkció bekapcsolása	m	+		PGAsl
COMPSURF	G	COMPSURF kompresszor funkció bekapcsolása	m	+		PGAsl
CONTDCON	P	Kontúr-dekódolást táblázati formában bekapcsolni		+	-	PGAsl
CONTPRON	P	Referencia feldolgozást bekapcsolni		+	-	PGAsl

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
CORROF	P	Az összes aktív átlapoló mozgás ki lesz kapcsolva.		+	-	PGsl
CORRTRAFO	F	Tájolás tengelyek offset-vektorait vagy irányvektorait a gép kinematikai modelljében módosítani		+	-	PGAsl
COS	F	Cosinus (trigon. függvény)		+	+	PGAsl
COUPDEF	P	ELG-egyesülés / szinkronorsó-egyesülés definíció		+	-	PGAsl
COUPDEL	P	ELG-egyesülés törlés		+	-	PGAsl
COUPOF	P	ELG-egyesülés / szinkronorsó-pár bekapcsolása		+	-	PGAsl
COUPOFS	P	ELG-egyesülés / szinkronorsó-pár kikapcsolás követő orsó állj-jal		+	-	PGAsl
COUPON	P	ELG-egyesülés / szinkronorsó-pár bekapcsolása		+	-	PGAsl
COUPONC	P	ELG-egyesülés / szinkronorsó-pár bekapcsolás megelőző programozás átvételével		+	-	PGAsl
COUPRES	P	ELG-egyesülés törlés		+	-	PGAsl
CP [6]	G	Pályamozgás	m	+		PGAsl
CPBC	K	Generátoros csatolás: mondatváltás kritérium		+	+	FB3sl (M3)
CPDEF	K	Generátoros csatolás: egy csatolási mód létrehozása		+	+	FB3sl (M3)
CPDEL	K	Generátoros csatolás: egy csatolási mód törlése		+	+	FB3sl (M3)
CPFMOF	K	Generátoros csatolás: követő tengely viselkedése teljes kikapcsolásnál		+	+	FB3sl (M3)
CPFMON	K	Generátoros csatolás: követő tengely viselkedése teljes bekapcsolásnál		+	+	FB3sl (M3)
CPFMSON	K	Generátoros csatolás: szinkronizációs módus		+	+	FB3sl (M3)
CPFPOS	K	Generátoros csatolás: követő tengely szinkron-pozíciója		+	+	FB3sl (M3)
CPFRS	K	Generátoros csatolás: koordináta vonatkoztatási rendszer		+	+	FB3sl (M3)
CPLA	K	Generátoros csatolás: egy vezető tengely definíciója		+	-	FB3sl (M3)
CPLCTID	K	Generátoros csatolás: görbe-táblázatok száma		+	+	FB3sl (M3)
CPLDEF	K	Generátoros csatolás: egy vezető tengely definíciója egy csatolási módus létrehozásához		+	+	FB3sl (M3)
CPLDEL	K	Generátoros csatolás: egy csatolási módus egy vezető tengelyének törlése		+	+	FB3sl (M3)
CPLDEN	K	Generátoros csatolás: csatolási tényező nevezője		+	+	FB3sl (M3)
CPLINSC	K	Generátoros csatolás: skálázási tényező egy vezető tengely bemenet értékéhez		+	+	FB3sl (M3)

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
CPLINTR	K	Generátoros csatolás: eltolási érték egy vezető tengely bemenet értékéhez		+	+	FB3sl (M3)
CPLNUM	K	Generátoros csatolás: csatolási tényező számlálója		+	+	FB3sl (M3)
CPLOF	K	Generátoros csatolás: egy csatolási módus egy vezető tengelyének kikapcsolása		+	+	FB3sl (M3)
CPLON	K	Generátoros csatolás: egy csatolási módus egy vezető tengelyének bekapcsolása		+	+	FB3sl (M3)
CPLOUTSC	K	Generátoros csatolás: skálázási tényező egy vezető tengely kimenet értékéhez		+	+	FB3sl (M3)
CPLOUTTR	K	Generátoros csatolás: eltolási érték egy csatolás kimenet értékéhez		+	+	FB3sl (M3)
CPLPOS	K	Generátoros csatolás: vezető tengely szinkron-pozíciója		+	+	FB3sl (M3)
CPLSETVAL	K	Generátoros csatolás: csatolás vonatkoztatás		+	+	FB3sl (M3)
CPMALARM	K	Generátoros csatolás: speciális csatolás-vonatkozású vészjelzések kiadásának elnyomása		+	+	FB3sl (M3)
CPMBRAKE	K	Generátoros csatolás: követő tengely viselkedése adott állj-jeleknél és parancsoknál		+	-	FB3sl (M3)
CPMPRT	K	Generátoros csatolás: Csatolás viselkedése munkadarabprogram indításnál programteszt általi mondat-keresés alatt		+	+	FB3sl (M3)
CPMRESET	K	Generátoros csatolás: csatolás viselkedése RESET-nél		+	+	FB3sl (M3)
CPMSTART	K	Generátoros csatolás: csatolás viselkedése munkadarabprogram indításánál		+	+	FB3sl (M3)
CPMVDI	K	Generátoros csatolás: követő tengely viselkedése adott NC/PLC interfész jelekre		+	+	FB3sl (M3)
CPOF	K	Generátoros csatolás: egy csatolási mód kikapcsolása		+	+	FB3sl (M3)
CPON	K	Generátoros csatolás: egy csatolási mód bekapcsolása		+	+	FB3sl (M3)
CPRECOF [6]	G	Programozható kontúrpontosság kikapcsolása	m	+		PGAsl
CPRECON	G	Programozható kontúrpontosság bekapcsolása	m	+		PGAsl
CPRES	K	Generátoros csatolás: aktiválja a szinkron-orsó csatolás beállított adatait		+	-	FB3sl (M3)
CPROT	P	Csatorna-specifikus védelmi tartomány be-/kikapcsolása		+	-	PGAsl
CPROTDEF	P	Egy csatorna-specifikus védőtartomány definíciója		+	-	PGAsl
CPSETTYPE	K	Generátoros csatolás: csatolás típus		+	+	FB3sl (M3)
CPSYNCOP	K	Generátoros csatolás: küszöbérték a "durva" pozíció-szinkron futásra		+	+	FB3sl (M3)

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
CPSYNCOP2	K	Generátoros csatolás: küszöbérték a "durva" 2 pozíció-szinkron futásra		+	+	FB3sl (M3)
CPSYNCOV	K	Generátoros csatolás: küszöbérték a "durva" sebesség-szinkron futásra		+	+	FB3sl (M3)
CPSYNFIP	K	Generátoros csatolás: küszöbérték a "finom" pozíció-szinkron futásra		+	+	FB3sl (M3)
CPSYNFIP2	K	Generátoros csatolás: küszöbérték a "finom" 2 pozíció-szinkron futásra		+	+	FB3sl (M3)
CPSYNFIV	K	Generátoros csatolás: küszöbérték a "finom" sebesség-szinkron futásra		+	+	FB3sl (M3)
CR	A	Körsugár	s	+		PGsl
CROT	F	Aktuális koordinátarendszer forgatása		+	-	PGAsl
CROTS	F	Programozható frame-forgatás térszöggel (forgatás a megadott tengelyekre)	s	+	-	PGsl
CRPL	F	Frame-forgatás egy tetszőleges síkban		+	-	FB1sl (K2)
CSCALE	F	Mértéktényező több tengelyre		+	-	PGAsl
CSPLINE	F	Köbös Spline	m	+		PGAsl
CT	G	Kör érintőleges átmenettel	m	+		PGsl
CTAB	F	Megállapítja a követő tengely pozícióját a vezető tengely pozíciója alapján a görbe-táblázatból		+	+	PGAsl
CTABDEF	P	Táblázat definíció bekapcsolása		+	-	PGAsl
CTABDEL	P	Görbe-táblázat törlése		+	-	PGAsl
CTABEND	P	Táblázat definíció kikapcsolása		+	-	PGAsl
CTABEXISTS	F	Megvizsgálja az n számú görbe-táblázatot		+	+	PGAsl
CTABFNO	F	A még lehetséges görbe-táblázatok száma a tárolóban		+	+	PGAsl
CTABFPOL	F	A még lehetséges polinomok száma a tárolóban		+	+	PGAsl
CTABFSEG	F	A még lehetséges görbe-szegmensek száma a tárolóban		+	+	PGAsl
CTABID	F	Visszaadja a n. görbe-táblázat táblázatszámát		+	+	PGAsl
CTABINV	F	Megállapítja a vezető tengely pozícióját a követő tengely pozíciója alapján a görbe-táblázatból		+	+	PGAsl
CTABISLOCK	F	Visszaadja az n számú görbe-táblázat tiltási állapotát		+	+	PGAsl
CTABLOCK	P	Törlés és átírás elleni tiltás beállítása		+	+	PGAsl
CTABMEMTYP	F	Visszaadja a tárolót, amelyben az n számú görbe-táblázat van.		+	+	PGAsl
CTABMPOL	F	A maximálisan lehetséges polinomok száma a tárolóban		+	+	PGAsl
CTABMSEG	F	A maximálisan lehetséges görbe-szegmensek száma a tárolóban		+	+	PGAsl

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
CTABNO	F	A definiált görbe-táblázatok száma az SRAM vagy DRAM tárolóban		+	+	FB3sl (M3)
CTABNOMEM	F	A definiált görbe-táblázatok száma az SRAM vagy DRAM tárolóban		+	+	PGAsl
CTABPERIOD	F	Visszaadja a táblázat periodicitását az n számmal		+	+	PGAsl
CTABPOL	F	A már használt polinomok száma a tárolóban		+	+	PGAsl
CTABPOLID	F	Az n számú görbe-táblázat által használt görbe-polinomok száma		+	+	PGAsl
CTABSEG	F	A már használt görbe-szegmensek száma a tárolóban		+	+	PGAsl
CTABSEGID	F	Az n számú görbe-táblázat által használt görbe-szegmensek száma		+	+	PGAsl
CTABSEV	F	Visszaadja a görbe-táblázat egy szegmensének követő tengely végértékét		+	+	PGAsl
CTABSSV	F	Visszaadja a görbe-táblázat egy szegmensének követő tengely kezdőértékét		+	+	PGAsl
CTABTEP	F	Visszaadja a vezető tengely értékét a görbe-táblázat végén		+	+	PGAsl
CTABTEV	F	Visszaadja a követő tengely értékét a görbe-táblázat végén		+	+	PGAsl
CTABTMAX	F	Visszaadja a követő tengely maximális értékét a görbe-táblázatban		+	+	PGAsl
CTABTMIN	F	Visszaadja a követő tengely minimális értékét a görbe-táblázatban		+	+	PGAsl
CTABTSP	F	Visszaadja a vezető tengely értékét a görbe-táblázat kezdetén		+	+	PGAsl
CTABTSV	F	Visszaadja a követő tengely értékét a görbe-táblázat kezdetén		+	+	PGAsl
CTABUNLOCK	P	Törlés és átírás elleni tiltás feloldása		+	+	PGAsl
CTOL	A	Kompresszor funkciók, tájolás-simítások és átsimítási módok kontúr-tűrése	m	+		PGAsl
CTRANS	F	Nullaponteltolás több tengelyre		+	-	PGAsl
CUT2D [6]	G	2D-s SZSK	m	+		PGsl
CUT2DD	G	Eltérés-szerszám vonatkoztatott 2½ D-s SZSK	m	+		PGsl
CUT2DF	G	2D-s SZSK, az aktuális frame-hez relatívan (ferde sík)	m	+		PGsl
CUT2DFD	G	Eltérés-szerszám vonatkoztatott 2½ D-s SZSK, az aktuális frame-hez relatívan (ferde sík)	m	+		PGsl
CUT3DC	G	3D-s SZSK kerületi marással	m	+		PGAsl
CUT3DCC	G	3D-s SZSK kerületi maráshoz egy határoló felület figyelembe vételével 3D-s sugárkorrekcióval: kontúr a megmunkálási felületen	m	+		PGAsl

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
CUT3DCCD	G	3D-s SZSK kerületi maráshoz egy határoló felület figyelembe vételével eltérés szerszámmal a szerszám középpont pályán: Ráállás a határoló felületre	m	+		PGAsl
CUT3DCD	G	Eltérés-szerszám vonatkoztatott 3D-s SZSK homlokmarásra	m	+		PGAsl
CUT3DF	G	3D-s SZSK homlokmarásra tájolás változásra	m	+		PGAsl
CUT3DFD	G	Eltérés-szerszám vonatkoztatott 3D-s SZSK homlokmarásra tájolás változással	m	+		PGAsl
CUT3DFF	G	3D-s SZSK homlokmarásra állandó tájolással. A szerszámtájolást a G17 - G19 határozza meg és adott esetben egy frame által elforgatott irány.	m	+		PGAsl
CUT3DFS	G	3D-s SZSK homlokmarásra állandó tájolással. A szerszámtájolást a G17 - G19 határozza meg és ezt framek nem befolyásolják.	m	+		PGAsl
CUTCONOF [6]	G	Állandó sugár-korrekció bekapcsolása	m	+		PGsl
CUTCONON	G	Állandó sugár-korrekció bekapcsolása	m	+		PGsl
CUTMOD	A	Korrekcióadatok módosítását forgatható szerszámoknál bekapcsolni (tájolható szerszámtartókkal kapcsolatban)	m	+		PGAsl
CUTMODK	A	Korrekcióadatok módosítását forgatható szerszámoknál bekapcsolni (tájolás transzformációkkal kapcsolatban, amelyek kinematikai láncként lettek definiálva)	m	+		PGAsl
CYCLE60	C (T)	Gravírozás ciklus		+		PGAsl
CYCLE61	C (T)	Síkmarás		+		PGAsl
CYCLE62	C (T)	Kontúrhívás		+		PGAsl
CYCLE63	C (T)	Kontúrzseb marása		+		PGAsl
CYCLE64	C (T)	Kontúrzseb előfúrása		+		PGAsl
CYCLE70	C (T)	Menetmarás		+		PGAsl
CYCLE72	C (T)	Pályamarás		+		PGAsl
CYCLE76	C (T)	Négyszögcsap marása		+		PGAsl
CYCLE77	C (T)	Körcsap marása		+		PGAsl
CYCLE78	C (T)	Furatmenet marás		+		PGAsl
CYCLE79	C (T)	Sokszög		+		PGAsl
CYCLE81	C (T)	Fúrás, központozás		+		PGAsl
CYCLE82	C (T)	Fúrás, sík-süllyesztés		+		PGAsl
CYCLE83	C (T)	Mélylyuk fúrása		+		PGAsl
CYCLE84	C (T)	Menetfúrás kiegyenlítő tokmány nélkül		+		PGAsl
CYCLE85	C (T)	Dörzsölés		+		PGAsl
CYCLE86	C (T)	Kiesztergálás		+		PGAsl
CYCLE92	C (T)	Leszúrás		+		PGAsl
CYCLE95	C (T)	Kontúr leforgácsolás		+		PGAsl

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
CYCLE98	C (T)	Menetlánc		+		PGAsl
CYCLE99	C (T)	Menet esztergálás		+		PGAsl
CYCLE150	C (M)	Mérési eredmények kijelzése / jegyzőkönyvezése		+		BNMsl
CYCLE435	C (T)	Lehúzás pozíciót kiszámítani		+		PGAsl
CYCLE495	C (T)	Profilozás		+		PGAsl
CYCLE750	C (A)	Belső munka-ciklus CYCLE751 ... CYCLE759-hez (tartalmazza az MMC utasítás a tulajdonképpeni funkció felhíváshoz)		-		FB3sl (T4)
CYCLE751	C (A)	Optimalizálási eljárás megnyitás / végrehajtás / lezárás		M		FB3sl (T4)
CYCLE752	C (A)	Tengelyt optimalizálási eljáráshoz hozzáadni		M		FB3sl(T4)
CYCLE753	C (A)	Optimalizálási módot kiválasztani		M		FB3sl (T4)
CYCLE754	C (A)	Adatkészlet hozzáadás / eltávolítás		M		FB3sl (T4)
CYCLE755	C (A)	Adatkészlet mentés / helyreállítás		M		FB3sl (T4)
CYCLE756	C (A)	Optimalizálási eredményeket aktiválni		M		FB3sl (T4)
CYCLE757	C (A)	Optimalizálási adatokat tárolni		M		FB3sl (T4)
CYCLE758	C (A)	Paraméter értékét változtatni		M		FB3sl (T4)
CYCLE759	C (A)	Paraméter értékét olvasni		M		FB3sl (T4)
CYCLE800	C (T)	Billentés		+		PGAsl
CYCLE801	C (T)	Rács vagy keret		+		PGAsl
CYCLE802	C (T)	Tetszőleges pozíciók		+		PGAsl
CYCLE830	C (T)	Mélylyuk fúrása 2		+		PGAsl
CYCLE832	C (T)	High Speed Settings		+		PGAsl
CYCLE840	C (T)	Menetfúrás kiegyenlítő tokmánnyal		+		PGAsl
CYCLE899	C (T)	Nyitott vájat marása		+		PGAsl
CYCLE930	C (T)	Beszúrás		+		PGAsl
CYCLE940	C (T)	Szabadra szúrás formák		+		PGAsl
CYCLE951	C (T)	Leforgácsolás		+		PGAsl
CYCLE952	C (T)	Kontúr-szúrás		+		PGAsl
CYCLE961	C (M)	Egy munkadarab sarok (belső vagy külső) helyzetét meghatározni és nullaponteltolásként használni.		+		BNMsl
CYCLE971	C (M)	Szerszám-mérőtapintót kalibrálni, szerszámhosszat és/vagy szerszámsugarat mérni (csak marás technológiához)		+		BNMsl
CYCLE973	C (M)	Munkadarab-mérőtapintót egy szerszám felületén vagy egy vájatban kalibrálni (csak esztergálás technológiához)		+		BNMsl
CYCLE974	C (M)	Munkadarab nullapontot a kiválasztott mérőtengelyen 1-pontos méréssel meghatározni (csak esztergálás technológiához)		+		BNMsl

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
CYCLE976	C (M)	Munkadarab-mérőtapintót egy kalibráló gyűrűben ill. egy kalibráló golyón a munkasíkban teljesen vagy egy élen egy megadott tengelyre és irányra kalibrálni		+		BNMsl
CYCLE977	C (M)	Középpontot a síkban és szélességet ill. átmérőt meghatározni		+		BNMsl
CYCLE978	C (M)	Egy él pozícióját megmérni a munkadarab koordinátarendszerben		+		BNMsl
CYCLE979	C (M)	Középpontot a síkban meghatározni, körszegmensek sugarát mérni		+		BNMsl
CYCLE982	C (M)	Szerszám-mérőtapintót kalibrálni, eszterga-, fúró- és marószerszámokat bemérni (csak esztergálás technológiához)		+		BNMsl
CYCLE994	C (M)	Munkadarab nullapontot a kiválasztott mérőtengelyen 2-pontos méréssel meghatározni (csak esztergálás technológiához)		+		BNMsl
CYCLE995	C (M)	Orsó merőlegességét a szerszámgépen megmérni		+		BNMsl
CYCLE996	C (M)	Transzformáció-releváns adatok meghatározása a kinematikus transzformációkhoz körtengelyekkel		+		BNMsl
CYCLE997	C (M)	Egy golyó középpontját és átmérőjét meghatározni, három elosztott golyó középpontjait mérni		+		BNMsl
CYCLE998	C (M)	Egy felület (sík) szöghelyzetét a munkasíkra vonatkoztatva meghatározni és az élek helyzetét a munkadarab koordináta-rendszerben meghatározni		+		BNMsl
CYCLE4071	C (T)	Hossz-köszörülés fogással a fordulóponton		+		PGAsl
CYCLE4072	C (T)	Hossz-köszörülés fogással a fordulóponton és megszakítás jel		+		PGAsl
CYCLE4073	C (T)	Hossz-köszörülés folyamatos fogásvétellel		+		PGAsl
CYCLE4074	C (T)	Hossz-köszörülés folyamatos fogásvétellel és megszakítás jellel		+		PGAsl
CYCLE4075	C (T)	Sík-köszörülés fogással a fordulóponton		+		PGAsl
CYCLE4077	C (T)	Sík-köszörülés fogással a fordulóponton és megszakítás jel		+		PGAsl
CYCLE4078	C (T)	Sík-köszörülés folyamatos fogásvétellel		+		PGAsl
CYCLE4079	C (T)	Sík-köszörülés középső fogással		+		PGAsl

Utasítások D ... F

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1][2][3][4][5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
D	A	Szerszámkorrekció-szám		+		PGsl
D0	A	D0-nál a korrekciók nem hatásosak a szerszámra		+		PGsl
DAC	K	Abszolút mondatonkénti tengely-specifikus átmérő-programozás	s	+		PGsl
DC	K	Abszolút méretmegadás körtengelyekre, pozícióra közvetlen rámenetel	s	+		PGsl
DCI	K	Adat-osztály I (= egyéni) hozzárendelni (csak SINUMERIK 828D!)		+		PGAsl
DCM	K	Adat-osztály M (= gépgyártó) hozzárendelni (csak SINUMERIK 828D!)		+		PGAsl
DCU	K	Adat-osztály M (= felhasználó) hozzárendelni (csak SINUMERIK 828D!)		+		PGAsl
DEF	K	Változó definíció		+		PGAsl
DEFAULT	K	CASE elágazásban ág		+		PGAsl
DEFINE	K	Kulcsszó makró-definíciókhoz		+		PGAsl
DELAYFSTOF	P	Egy Stopp-Delay tartomány végét definiálni	m	+	-	PGAsl
DELAYFSTON	P	Egy Stopp-Delay tartomány elejét definiálni	m	+	-	PGAsl
DELDL	F	Additív korrekciót törölni		+	-	PGAsl
DELDTG	P	Maradékút törlés		-	+	FBSYsl
DELETE	P	A megadott fájlt törölni. A fájlnevet ággal és fájl-jelölővel lehet megadni.		+	-	PGAsl
DELMLOWNER	F	A szerszám tulajdonosi tárhelyét törölni		+	-	FBWsl
DELMLRES	F	Tárhely foglalást törölni		+	-	FBWsl
DELMT	P	Multitool törlése		+	-	FBWsl
DELOBJ	F	Kinematikai láncok elemei, védelmi tartományok és védelmi tartomány elemei, ütközési párok és transzformációs adatok törlése		+		PGAsl
DELT	P	Szerszámot törölni		+	-	FBWsl
DELTC	P	Szerszámtartó adatkészletet törölni		+	-	FBWsl
DELTOOLENV	F	Szerszám-környezeteket leíró adatkészleteket törölni		+	-	PGAsl
DIACYCOFA	K	Tengely-specifikus modális átmérő-programozás: KI a ciklusokban	m	+		FB1sl (P1)
DIAM90	G	Átmérő-programozás G90-nél, sugár-programozás G91-nél	m	+		PGAsl
DIAM90A	K	Tengely-specifikus modális átmérő- programozás G90-nél és AC, sugár-programozás G91-nél és IC	m	+		PGsl
DIAMCHAN	K	Összes tengely átvétele a tengelyfunkciók gépadatból az átmérő-programozás csatorna állapotába		+		PGsl

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1][2][3][4][5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
DIAMCHANA	K	Átmérő-programozás csatorna-állapot átvétele		+		PGsl
DIAMCYCOF	G	Csatorna-specifikus átmérő-programozás KI a ciklusokban	m	+		FB1sl (P1)
DIAMOF [6]	G	Átmérő-programozás KI alaphelyzetet lásd gépgyártónál	m	+		PGsl
DIAMOFA	K	Tengely-specifikus modális átmérő-programozás: KI alaphelyzetet lásd gépgyártónál	m	+		PGsl
DIAMON	G	Átmérő-programozás BE	m	+		PGsl
DIAMONA	K	Tengely-specifikus modális átmérő-programozás: BE engedélyezést lásd gépgyártónál	m	+		PGsl
DIC	K	Relatív mondatonkénti tengely-specifikus átmérő-programozás	s	+		PGsl
DILF	A	Visszahúzási út (hossz)	m	+		PGsl
DISABLE	P	Interrupt KI		+	-	PGAsl
DISC	A	Átmeneti kör megnövelés szerszám-sugárkorrekció	m	+		PGsl
DISCL	A	Gyors ráállás mozgás végpontjának távolsága a megmunkálási síktól		+		PGsl
DISPLOF	PA	Aktuális mondat kijelzést elnyomni		+		PGAsl
DISPLON	PA	Aktuális mondat kijelzést elnyomást megszüntetni		+		PGAsl
DISPR	A	Repos pályakülönbség	s	+		PGAsl
DISR	A	Repos távolság	s	+		PGAsl
DISRP	A	Visszahúzási sík távolsága a megmunkálási síktól puha rá- és lemenetnél		+		PGsl
DITE	A	Menet kifutási út	m	+		PGsl
DITS	A	Menet bekezdési út	m	+		PGsl
DIV	K	Egész osztás		+		PGAsl
DL	A	Helyfüggő additív szerszámkorrekciót kiválasztani (DL, összeg beállítási korrekció)	m	+		PGAsl
DO	A	Kulcsszó szinkron-akciókhoz, teljesült feltételnél vált ki akciót		-	+	FBSYsl
DRFOF	P	Kézikerék-eltolások kikapcsolása (DRF)	m	+	-	PGsl
DRIVE	G	Sebességtől függő pályagyorsítás	m	+		PGAsl
DRIVEA	P	Megtört gyorsulási jelleggörbét a programozott tengelyekre bekapcsolni		+	-	PGAsl
DYNFINISH	G	Finom-simítás dinamika	m	+		PGAsl
DYNNORM [6]	G	Normál dinamika	m	+		PGAsl
DYNPOS	G	Dinamika pozícionáló üzemre, menetfúrásra	m	+		PGAsl
DYNROUGH	G	Nagyolás dinamika	m	+		PGAsl
DYNSEMIFIN	G	Simítás dinamika	m	+		PGAsl

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1][2][3][4][5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
DZERO	P	A TO egység összes D-számát érvénytelennek jelöli		+	-	PGAsl
EAUTO	G	Utolsó Spline szakasz megadása a következő 3 ponttal	m	+		PGAsl
EGDEF	P	Elektronikus hajtómű definíció		+	-	PGAsl
EGDEL	P	Követő tengely csatolás definíciót törölni		+	-	PGAsl
EGOFC	P	Elektronikus hajtóművet folyamatosan kikapcsolni		+	-	PGAsl
EGOFS	P	Elektronikus hajtóművet szelektíven kikapcsolni		+	-	PGAsl
EGON	P	Elektronikus hajtóművet bekapcsolni		+	-	PGAsl
EGONSYN	P	Elektronikus hajtóművet bekapcsolni		+	-	PGAsl
EGONSYNE	P	Elektronikus hajtóművet bekapcsolni, a rámeneti módus megadásával		+	-	PGAsl
ELSE	K	Program elágazás, ha IF-feltétel nem teljesült		+		PGAsl
ENABLE	P	Interrupt BE		+	-	PGAsl
ENAT [6]	G	Természetes görbe-átmenet a következő mozgás-mondathoz	m	+		PGAsl
ENDFOR	K	FOR-számlálóhurok végsora		+		PGAsl
ENDIF	K	IF-elágazás végsora		+		PGAsl
ENDLABEL	K	Vég-jelölő munkadarabprogram ismétlések-nél REPEAT-tel		+		PGAsl, FB1sl (K1)
ENDLOOP	K	LOOP végtelen programhurok végsora		+		PGAsl
ENDPROC	K	PROC kezdősorú program végsor		+		
ENDWHILE	K	WHILE-hurok végsora		+		PGAsl
ESRR	P	Hajtásalapú ESR-visszahúzást a hajtásban paraméterezni		+		PGAsl
ESRS	P	Hajtásalapú ESR-leállítást a hajtásban paraméterezni		+		PGAsl
ETAN	G	Érintőleges görbe-átmenet a következő elmozdulási mondathoz Spline-kezdetnél	m	+		PGAsl
EVERY	K	Szinkron-akciót végrehajtani, ha a feltétel FALSE-ról TRUE-ra változik		-	+	FBSYsl
EX	K	Kulcsszó érték hozzárendeléshez exponenciális írásmódban		+		PGAsl
EXECSTRING	P	Egy string-változó átadása a végrehajtandó programsorral		+	-	PGAsl
EXECTAB	P	Mozgás-táblázat egy elemét feldolgozni		+	-	PGAsl
EXECUTE	P	Program végrehajtás BE		+	-	PGAsl
EXP	F	Exponenciális függvény ex		+	+	PGAsl
EXTCALL	A	Külső alprogramot feldolgozni		+	+	PGAsl
EXTCLOSE	P	Írásra megnyitott külső készüléket/fájlt lezárni		+	-	PGAsl
EXTERN	K	Egy paraméter-átadásos alprogram ismertté tétele		+		PGAsl

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1][2][3][4][5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
EXTOPEN	P	Külső készüléket/fájlt a csatornára íráshoz megnyitni		+	-	PGAsl
F	A	Előtolás érték (G4-gyel kapcsolatosan F-fel a várakozási időt is programozzuk)		+	+	PGsl
FA	K	Tengely előtolás	m	+	+	PGsl
FAD	A	Fogásvétel előtolás puha rá- és lemenethez		+		PGsl
FALSE	K	Logikai állandó: hamis		+	+	PGAsl
FB	A	Mondatonkénti előtolás		+		PGsl
FCTDEF	P	Polinom függvény definiálása		+	-	PGAsl
FCUB	G	Előtolás köbös Spline szerint változtatható	m	+		PGAsl
FD	A	Pályaelőtolás kézikerék-átlapolódásra	s	+		PGsl
FDA	K	Tengely előtolás kézikerék-átlapolódásra	s	+		PGsl
FENDNORM [6]	G	Sarok-késleltetés KI	m	+		PGAsl
FFWOF [6]	G	Elővezérlés KI	m	+		PGAsl
FFWON	G	Elővezérlés BE	m	+		PGAsl
FGREF	K	Vonatkozási sugár kör-tengelyeknél vagy pálya-vonatkozási tényező tájolótengelyeknél (vektor-interpoláció)	m	+		PGsl
FGROUP	P	Pálya-előtolásos tengely(ek) meghatározása		+	-	PGsl
FI	K	Frame-adatok hozzáférés paraméter: finom-eltolás		+		PGAsl
FIFOCTRL	G	Előrefutás-puffer vezérlése	m	+		PGAsl
FILEDATE	P	Visszaadja a fájl utolsó írásának dátumát		+	-	PGAsl
FILEINFO	P	Visszaadja a FILEDATE, FILESIZE, FILESTAT és FILETIME összegét		+	-	PGAsl
FILESIZE	P	Visszaadja a fájl aktuális méretét		+	-	PGAsl
FILESTAT	P	Visszaadja az olvasás, írás, végrehajtás, kijelzés, törlés (rwxsd) jogok fájl-állapotokat		+	-	PGAsl
FILETIME	P	Visszaadja a fájl utolsó írásának időpontját		+	-	PGAsl
FINEA	K	Mozgás vége "Pontos-állj finom" eléréséné l	m	+		PGAsl
FL	K	Határsebesség szinkrontengelyekre	m	+		PGsl
FLIN	G	Előtolás lineárisan változtatható	m	+		PGAsl
FMA	K	Több tengely előtolás	m	+		PGsl
FNORM [6]	G	Normál előtolás DIN66025 szerint	m	+		PGAsl
FOC	K	Mondatonként hatásos nyomaték/erő-határolás	s	-	+	FBSYsl
FOCOF	K	Modális nyomaték/erő-határolást kikapcsolni	m	-	+	FBSYsl
FOCON	K	Modális nyomaték/erő-határolást bekapcsolni	m	-	+	FBSYsl
FOR	K	Számláló hurok fix számú átfutással		+		PGAsl
FP	A	Fixpont: az elérendő fixpont száma	s	+		PGsl
FPO	K	Egy polinommal programozott előtolás lefutás		+		PGAsl
FPR	P	Körtengely jelölése		+	-	PGsl

Utasítás	Fajta 1)	Jelentés	W 2)	TP 3)	SA 4)	Leírást lásd 5)
1)2)3)4)5) Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
FPRAOF	P	Fordulati előtolás kikapcsolása		+	-	PGsl
FPRAON	P	Fordulati előtolás bekapcsolása		+	-	PGsl
FRAME	K	Adattípus a koordinátarendszer megadásához		+		PGAsl
FRC	A	Előtolás sugárnál és letörésnél	s	+		PGsl
FRCM	A	Előtolás sugárnál és letörésnél modális	m	+		PGsl
FROM	K	Az akció végre lesz hajtva, ha a feltétel egyszer teljesült és amíg a szinkron-akció aktív.		-	+	FBSYsl
FTOC	P	Szerszám finomkorrekciót változtatni		-	+	FBSYsl
FTOCOF 6)	G	Online hatásos szerszám-finomkorrekció KI	m	+		PGAsl
FTOCON	G	Online hatásos szerszám-finomkorrekció BE	m	+		PGAsl
FXS	K	Fix-ütközőre menet be	m	+	+	PGsl
FXST	K	Fix-ütközőre menet nyomatékhatár	m	+	+	PGsl
FXSW	K	Fix-ütközőre menet ellenőrzőablak		+	+	PGsl
FZ	K	Fog-előtolás	m	+		PGsl

Utasítások G ... L

Utasítás	Fajta 1)	Jelentés	W 2)	TP 3)	SA 4)	Leírást lásd 5)
1) 2) 3) 4) 5) Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
G0	G	Egyenes interpoláció gyorsmenettel (gyorsmeneti mozgás)	m	+		PGsl
G1 6)	G	Egyenes interpoláció előtolással (egyenes interpoláció)	m	+		PGsl
G2	G	Kör-interpoláció órajárás irányában	m	+		PGsl
G3	G	Kör-interpoláció órajárás irányával szemben	m	+		PGsl
G4	G	Várakozási idő, időben előre-meghatározott	s	+		PGsl
G5	G	Ferde beszúrás köszörülés	s	+		PGAsl
G7	G	Kiegyenlítő mozgás ferde beszúrás köszörülésnél	s	+		PGAsl
G9	G	Pontos állj - sebesség-csökkenés	s	+		PGsl
G17 6)	G	Munkasík kiválasztás X/Y	m	+		PGsl
G18	G	Munkasík kiválasztás Z/X	m	+		PGsl
G19	G	Munkasík kiválasztás Y/Z	m	+		PGsl
G25	G	Alsó munkatér határolás	s	+		PGsl
G26	G	Felső munkatér határolás	s	+		PGsl
G33	G	Menetvágás állandó emelkedéssel	m	+		PGsl
G34	G	Menetvágás lineárisan növekvő emelkedéssel:	m	+		PGsl
G35	G	Menetvágás lineárisan csökkenő emelkedéssel:	m	+		PGsl

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
G40 [6]	G	Szerszámsugár-korrekció KI	m	+		PGsl
G41	G	Szerszámsugár-korrekció a kontúrtól balra	m	+		PGsl
G42	G	Szerszámsugár-korrekció a kontúrtól jobbra	m	+		PGsl
G53	G	Aktuális nullaponteltolás elnyomása (mondatonként)	s	+		PGsl
G54	G	1. beállítható nullaponteltolás	m	+		PGsl
G55	G	2. beállítható nullaponteltolás	m	+		PGsl
G56	G	3. beállítható nullaponteltolás	m	+		PGsl
G57	G	4. beállítható nullaponteltolás	m	+		PGsl
G58 (840D sl)	G	Abszolút programozható nullaponteltolás (durva eltolás)	s	+		PGsl
G58 (828D)	G	5. beállítható nullaponteltolás	m	+		PGsl
G59 (840D sl)	G	Hozzáadódóan programozható nullaponteltolás (finom eltolás)	s	+		PGsl
G59 (828D)	G	6. beállítható nullaponteltolás	m	+		PGsl
G60 [6]	G	Pontos állj - sebesség-csökkenés	m	+		PGsl
G62	G	Sarok-késleltetés belső sarkokon aktív szerszámsugár-korrekciónál (G41, G42)	m	+		PGAsl
G63	G	Menetfúrás kiegyenlítő tokmánnyal	s	+		PGsl
G64	G	Pályavezérlő-üzem	m	+		PGsl
G70	G	Geometriai adatok méretmegadás hüvelykben (hosszak)	m	+	+	PGsl
G71 [6]	G	Geometriai adatok méretmegadás metrikus (hosszak)	m	+	+	PGsl
G74	G	Referenciapontra menet	s	+		PGsl
G75	G	Fix-pontra menet	s	+		PGsl
G90 [6]	G	Méretmegadás abszolút	m/s	+		PGsl
G91	G	Lánc méretadat	m/s	+		PGsl
G93	G	Idő-reciprok előtolás 1/perc	m	+		PGsl
G94 [6]	G	Lineáris előtolás F mm/perc vagy hüvelyk/perc-ben és fok/perc-ben	m	+		PGsl
G95	G	Fordulati előtolás F mm/ford vagy hüvelyk/ford-ban	m	+		PGsl
G96	G	Fordulati előtolás (mint G95-nél) és állandó vágássebesség	m	+		PGsl
G97	G	Fordulati előtolás és állandó orsó-fordulatszám (állandó vágósebesség KI)	m	+		PGsl
G110	G	Pólus programozás relatív az utolsó programozott parancspozícióhoz	s	+		PGsl
G111	G	Pólus programozás relatív az aktuális munkadarab-koordinátarendszer nullapontjához	s	+		PGsl
G112	G	Pólus programozás relatív az utolsó érvényes pólushoz	s	+		PGsl
G140 [6]	G	Rámenetel irány WAB megadás G41/G42-vel	m	+		PGsl

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
G141	G	Ráállás irány WAB balra a kontúrtól	m	+		PGsl
G142	G	Ráállás irány WAB jobbra a kontúrtól	m	+		PGsl
G143	G	Rámenetel irány WAB érintő függő	m	+		PGsl
G147	G	Puha ráállás egyenessel	s	+		PGsl
G148	G	Puha lemenet egyenessel	s	+		PGsl
G153	G	Aktuális frame-k elnyomása, beleértve az alap-frame-t	s	+		PGsl
G247	G	Puha ráállás körrel	s	+		PGsl
G248	G	Puha lemenet negyedkörrel	s	+		PGsl
G290 [6]	G	Átkapcsolás SINUMERIK-módra BE	m	+		FBWsl
G291	G	Átkapcsolás ISO2/3-módra BE	m	+		FBWsl
G331	G	Menetfúrás kiegyenlítő tokmány nélkül, pozitív emelkedés, jobbra	m	+		PGsl
G332	G	Menetfúrás kiegyenlítő tokmány nélkül, negatív emelkedés, balra	m	+		PGsl
G335	G	Egy íves menet esztergálása órajárás irányában	m	+		PGsl
G336	G	Egy íves menet esztergálása órajárással szemben	m	+		PGsl
G340 [6]	G	Térbeli rámenetel (mélységben és síkban egyidőben (helix))	m	+		PGsl
G341	G	Először a függőleges tengelyen (z) ráállni, azután a síkban	m	+		PGsl
G347	G	Puha rámenetel félkörrel	s	+		PGsl
G348	G	Puha elmenetel félkörrel	s	+		PGsl
G450 [6]	G	Átmenet-kör	m	+		PGsl
G451	G	Ekvidisztánsok metszéspontja	m	+		PGsl
G460 [6]	G	Ütközés-felügyelet bekapcsolása rá- és elmeneti mondatra	m	+		PGsl
G461	G	Egy kör beszúrása a szerszámsugár-korrekciós mondatba	m	+		PGsl
G462	G	Egy egyenes beszúrása a szerszámsugár-korrekciós mondatba	m	+		PGsl
G500 [6]	G	Az összes beállítható frame kikapcsolása, alap-frame-k aktívak	m	+		PGsl
G505 ... G599	G	5. ... 99. beállítható nullaponteltolás	m	+		PGsl
G601 [6]	G	Mondatváltás pontos állj finomnál	m	+		PGsl
G602	G	Mondatváltás pontos állj durvánál	m	+		PGsl
G603	G	Mondatváltás IPO mondatvégnél	m	+		PGsl
G621	G	Sarok-késleltetés minden sarkokon	m	+		PGAsl
G641	G	Pályavezérlő-üzem átsimítással út-kritérium szerint (= programozható átsimítási távolság)	m	+		PGsl
G642	G	Pályavezérlő-üzem átsimítással megadott tűrések betartásával	m	+		PGsl

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
G643	G	Pályavezérlő-üzem átsimítással megadott tűrések betartásával (mondaton belül)	m	+		PGsl
G644	G	pályavezérlő-üzem átsimítással maximális lehetséges dinamikával	m	+		PGsl
G645	G	Pályavezérlő-üzem átsimítással a sarkokon és érintőleges mondat-átmenetek a megadott tűrések betartásával	m	+		PGsl
G700	G	Geometriai és technológiai adatok méretmegadása hüvelykben (hosszak, előtolás)	m	+	+	PGsl
G710 [6]	G	Geometriai és technológiai adatok méretmegadása metrikusan (hosszak, előtolás)	m	+	+	PGsl
G810 [6], ..., G819	G	Az OEM-alkalmazók számára foglalt G-csoport		+		PGAsl
G820 [6], ..., G829	G	Az OEM-alkalmazók számára foglalt G-csoport		+		PGAsl
G931	G	Előtolás megadása elmozdulási idővel, állandó pályasebességet kikapcsolni	m	+		
G942	G	Lineáris előtolás és állandó vágósebesség vagy orsó-fordulatszámot befagyasztani	m	+		
G952	G	Fordulati előtolás és állandó vágósebesség vagy orsó-fordulatszámot befagyasztani	m	+		
G961	G	Lineáris előtolás (mint G94-nél) és állandó vágássebesség	m	+		PGsl
G962	G	Lineáris előtolás vagy fordulati előtolás és állandó vágósebesség	m	+		PGsl
G971	G	Lineáris előtolás és állandó orsó-fordulatszám (állandó vágósebesség KI)	m	+		PGsl
G972	G	Lineáris előtolás vagy fordulati előtolás és állandó orsó-fordulatszám (állandó vágósebesség KI)	m	+		PGsl
G973	G	Fordulati előtolás orsó-fordulatszám határolás nélkül és állandó orsó-fordulatszám (G97 LIMS nélkül ISO-módushoz)	m	+		PGsl
GEOAX	P	Geometria-tengelyekhez 1 - 3 új csatorna-tengelyt hozzárendelni		+	-	PGAsl
GET	P	Szabaddá tett tengelyeket a csatornák között cserélni		+	+	PGAsl
GETACTT	F	Aktív szerszámot egy csoport hasonló nevű szerszámból meghatározni		+	-	FBWsl
GETACTTD	F	Egy abszolút D-számhoz meghatározza a hozzátartozó T-számot		+	-	PGAsl
GETD	P	Tengelyt a csatornák között közvetlenül cserélni		+	-	PGAsl
GETDNO	F	Egy szerszám (T) egy vágóélének (CE) D-számát adja		+	-	PGAsl
GETEXET	P	A beváltott T-szám olvasása		+	-	FBWsl

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
GETFREELOC	P	Egy megadott szerszámnak egy üres helyet keresni a tárakban		+	-	FBWsl
GETSELT	P	Előválasztott T-számot adja		+	-	FBWsl
GETT	F	T-számot a szerszámnévhez meghatározni		+	-	FBWsl
GETTCOR	F	Szerszámhosszakat ill. szerszámhossz-komponenseket kiolvasni		+	-	PGAsl
GETTENV	F	T-, D-, és DL-számokat olvasni		+	-	PGAsl
GETVARAP	F	Egy rendszer/felhasználói változót olvasni		+	-	PGAsl
GETVARDFT	F	Egy rendszer/felhasználói változó alapértékét olvasni		+	-	PGAsl
GETVARLIM	F	Egy rendszer/felhasználói változó határértékét olvasni		+	-	PGAsl
GETVARPHU	F	Egy rendszer/felhasználói változó fizikai egységét olvasni		+	-	PGAsl
GETVARTYP	F	Egy rendszer/felhasználói változó adattípusát olvasni		+	-	PGAsl
GFRAME0 ... GFRAME100	G	Csatorna adattárolás <n> köszörű frame aktiválása	m	+		PGsl
GOTO	K	Ugrás utasítás először előre azután hátra (először a program vége azután a program eleje irányába)		+		PGAsl
GOTOB	K	Ugrás utasítás hátra (program eleje irányába)		+		PGAsl
GOTOC	K	Mint GOTO, de 14080 "Ugráscélt nem találta" vészjelzés elnyomása		+		PGAsl
GOTOF	K	Ugrás utasítás előre (program vége irányába)		+		PGAsl
GOTOS	K	Visszaugrás a programkezdetre		+		PGAsl
GP	K	Kulcsszó a pozíció jellemzők közvetett programozására		+		PGAsl
GROUP_ ADDEND	C (T)	Bejáratás vége		+		PGAsl
GROUP_BEGIN	C (T)	Programblokk kezdete		+		PGAsl
GROUP_END	C (T)	Programblokk vége		+		PGAsl
GWPSOF	P	Állandó tárcsakerületi sebesség (SUG) lekapcsolása	s	+	-	PGsl
GWPSON	P	Állandó tárcsakerületi sebesség (SUG) bekapcsolása	s	+	-	PGsl
H...	A	Segédfunkció kiadása a PLC-re		+	+	PGsl/FB1sl (H2)
HOLES1	C (T)	Lyuksor		+		PGAsl
HOLES2	C (T)	Lyukkör		+		PGAsl
I	A	Interpolációs paraméter	s	+		PGsl
I1	A	Közbenső pont koordináták	s	+		PGsl
IC	K	Lánc méret-beadás	s	+		PGsl
ICYCOF	P	Egy technológiai ciklus összes mondatát az ICYCOF után egy IPO-ütemben feldolgozni		+	+	FBSYsl

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
ICYCON	P	Egy technológiai ciklus minden egyes mondatát az ICYCON után egy külön IPO-ütemben feldolgozni		+	+	FBSYsl
ID	K	Modális szinkron-akciók jelölése	m	-	+	FBSYsl
IDS	K	Modális statikus szinkron-akciók jelölése		-	+	FBSYsl
IF	K	Egy feltételes ugrás bevezetése egy munkadarabprogramban / technológiai ciklusban		+	+	PGAsl
INDEX	F	Egy karakter indexének meghatározása a bemeneti stringben		+	-	PGAsl
INICF	K	Változók inicializálása NEWCONF-nál		+		PGAsl
INIPO	K	Változók inicializálása Power On-nál		+		PGAsl
INIRE	K	Változók inicializálása Reset-nél		+		PGAsl
INIT	P	Egy adott NC programot egy adott csatornában feldolgozáshoz kiválasztani		+	-	PGAsl
INITIAL		Egy INI fájl létrehozása az összes tartományról		+		PGAsl
INT	K	Adattípus: egész-szám előjellel		+		PGAsl
INTERSEC	F	Két kontúrelem metszéspontját kiszámítani		+	-	PGAsl
INVCCW	G	Mozgás egy evolvensen az óramutató járásával ellentétes irányban	m	+		PGsl
INVCW	G	Mozgás egy evolvensen az óramutató járása irányában	m	+		PGsl
INVFRAME	F	Egy frame-ből az inverz frame-t kiszámítani		+	-	FB1sl (K2)
IP	K	Változtatható interpolációs paraméter		+		PGAsl
IPOBRKA	P	Mozgási kritérium a fékrámpa kezdőpontjától	m	+	+	
IPOENDA	K	Mozgás vége "IPO-Stop" elérésénél	m	+		PGAsl
IPTRLOCK	P	A keresésre alkalmatlan programszakasz kezdetének befagyasztása a következő gépi funkciós mondatra.	m	+	-	PGAsl
IPTRUNLOCK	P	A keresésre alkalmatlan programszakasz végét a megszakítás időpontjában aktuális mondatra beállítani.	m	+	-	PGAsl
IR	A	Kör középpont koordináták (X irány) egy íves menet esztergálásánál		+		PGsl
ISAXIS	F	Megvizsgálja, hogy a paraméterként megadott geometria-tengely 1-e		+	-	PGAsl
ISD	A	Bemerülés-mélység	m	+		PGAsl
ISFILE	F	Megvizsgálja, hogy a egy fájl az NCK felhasználói tárolóban létezik-e		+	-	PGAsl
ISNUMBER	F	Megvizsgálja, hogy a bemeneti stringet át lehet-e alakítani számmá		+	-	PGAsl
ISOCALL	K	Egy ISO nyelven programozott program közvetett felhívása		+		PGAsl
ISVAR	F	Megvizsgálja, hogy az átadási paraméter egy, az NC-ben ismert változót tartalmaz-e		+	-	PGAsl

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
J	A	Interpolációs paraméter	s	+		PGsl
J1	A	Közbenső pont koordináták	s	+		PGsl
JERKA	P	Az MD-vel beállított gyorsulási viselkedést a programozott tengelyekre aktiválni		+	-	
JERKLIM	K	Maximális tengely-rándítás csökkentése vagy növelése	m	+		PGAsl
JERKLIMA	K	Maximális tengely-rándítás csökkentése vagy növelése	m	+	+	PGAsl
JR	A	Kör középpont koordináták (Y irány) egy íves menet esztergálásánál		+		PGsl
K	A	Interpolációs paraméter	s	+		PGsl
K1	A	Közbenső pont koordináták	s	+		PGsl
KONT	G	Kontúr megkerülése szerszámkorrekciónál	m	+		PGsl
KONTC	G	Állandó görbületű polinommal rá-/lemenni	m	+		PGsl
KONTT	G	Állandó érintőjű polinommal rá-/lemenni	m	+		PGsl
KR	A	Kör középpont koordináták (Z irány) egy íves menet esztergálásánál		+		PGsl
L	A	Alprogram-szám	s	+	+	PGAsl
LEAD	A	Előre-sietési szög 1. szerszámtájolás 2. tájolási polinom	m	+		PGAsl
LEADOF	P	Tengely vezetőérték-csatolás KI		+	+	PGAsl
LEADON	P	Tengely vezetőérték-csatolás BE		+	+	PGAsl
LENTOAX	F	Információkat ad az aktív szerszám L1, L2 és L3 szerszámhosszainak a hozzárendeléséről az abszcisszához, ordinátához és applikátá-hoz		+	-	PGAsl
LFOF [6]	G	Menetvágás gyors visszahúzás KI	m	+		PGsl
LFON	G	Menetvágás gyors visszahúzás BE	m	+		PGsl
LFPOS	G	Az POLFMASK vagy POLFMLIN által ismert-té tett tengelyek visszahúzása a POLF-fal programozott abszolút tengelypozícióra	m	+		PGsl
LFTXT [6]	G	Gyors leemelésnél a visszahúzási mozgás síkja a pálya-érintőből és az aktuális szerszá-mirányból lesz meghatározva	m	+		PGsl
LFWP	G	Gyors leemelésnél a visszahúzási mozgás síkja az aktuális munkasíkkal (G17/G18/G19) lesz meghatározva	m	+		PGsl
LIFTFAST	K	Gyors leemelés		+		PGsl
LIMS	K	Fordulatszám-határolás G96/G961 és G97 esetén	m	+		PGsl
LLI	K	Változók alsó határértéke		+		PGAsl
LN	F	Természetes logaritmus		+	+	PGAsl
LOCK	P	Szinkron-akció ID-vel tiltani (technológiai ciklust megállítani)		-	+	FBSYsl

Utasítás	Fajta 1)	Jelentés	W 2)	TP 3)	SA 4)	Leírást lásd 5)
1) 2) 3) 4) 5) Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
LONGHOLE	C (T)	Hosszlyuk		+		PGAsl
LOOP	K	Végtelen hurok bevezetése		+		PGAsl

Utasítások M ... R

Utasítás	Fajta 1)	Jelentés	W 2)	TP 3)	SA 4)	Leírást lásd 5)
1) 2) 3) 4) 5) Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
M0		Programozott állj		+	+	PGsl
M1		Választható állj		+	+	PGsl
M2		Főprogram programvég (mint M30)		+	+	PGsl
M3		Orsó forgásirány jobbra		+	+	PGsl
M4		Orsó forgásirány balra		+	+	PGsl
M5		Orsó állj		+	+	PGsl
M6		Szerszámcsere		+	+	PGsl
M17		Alprogram vége		+	+	PGsl
M19		Orsó pozicionálása az SD43240-be beadott pozícióra		+	+	PGsl
M30		Főprogram programvég (mint M2)		+	+	PGsl
M40		Automatikus hajtómű-kapcsolás		+	+	PGsl
M41 ... M45		Hajtóműfokozat 1 ... 5		+	+	PGsl
M70		Átmenet tengelyüzembe		+	+	PGsl
MASLDEF	P	Master/Slave tengely-egyesülés definiálás		+	+	PGAsl
MASLDEL	P	Master/Slave tengely-egyesülés felbontása és az egyesülés definíciójának törlése		+	+	PGAsl
MASLOF	P	Egy ideiglenes csatolás kikapcsolása		+	+	PGAsl
MASLOFS	P	Egy ideiglenes csatolás kikapcsolása a Slave-tengely automatikus leállításával		+	+	PGAsl
MASLON	P	Egy ideiglenes csatolás bekapcsolása		+	+	PGAsl
MATCH	F	Egy string keresése stringben		+	-	PGAsl
MAXVAL	F	Nagyobb érték két változóból (aritmetikai függvény)		+	+	PGAsl
MCALL	K	Modális alprogram hívás		+		PGAsl
MEAC	K	Tengely folytonos mérés maradékút törlés nélkül	s	+	+	PGAsl
MEAFRAME	F	Frame-számítás mérőpontokból		+	-	PGAsl
MEAS	A	Mérés maradékút törléssel	s	+		PGAsl
MEASA	K	Tengely mérés maradékút törléssel	s	+	+	PGAsl
MEASURE	F	Számítási módszer a munkadarab és a szerszám méréshez		+	-	FB1sl (M5)
MEAW	A	Mérés maradékút-törlés nélkül	s	+		PGAsl
MEAWA	K	Tengely mérés maradékút törlés nélkül	s	+	+	PGAsl

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
MI	K	Hozzáférés Frame-adatokhoz: Tükrözés		+		PGAsl
MINDEX	F	Egy karakter indexének meghatározása a bemeneti stringben		+	-	PGAsl
MINVAL	F	Kisebb érték két változóból (aritmetikai függvény)		+	+	PGAsl
MIRROR	G	Programozható tükrözés	s	+		PGAsl
MMC	P	Munkadarab-programból interaktív dialógusablakot a HMI-n felhívni		+	-	PGAsl
MOD	K	Modulo osztás		+		PGAsl
MODAXVAL	F	Egy modulo körtengely modulo pozícióját megállapítani		+	-	PGAsl
MOV	K	Pozícionáló tengely indítása		-	+	FBSYsl
MOVT	A	Egy elmozdulás végpontját szerszámirányban megadni				FB1(K2)
MSG	P	Programozható jelentések	m	+	-	PGsl
MVTOOL	P	Programutasítás egy szerszám mozgatására		+	-	FBWsl
N	A	NC mellékmondat-szám		+		PGsl
NAMETOINT	F	Rendszerváltozó index megállapítása		+		PGAsl
NC	K	Adatok érvényességi tartományának megadása		+		PGAsl
NEWCONF	P	Megváltoztatott gépadatokat átvenni (megfelel a "Gépadatokat hatásossá tenni"-nek)		+	-	PGAsl
NEWMT	F	Új Multitool-t létrehozni		+	-	FBWsl
NEWT	F	Új szerszámot létrehozni		+	-	FBWsl
NORM [6]	G	Normál-beállítás a szerszámkorrekció kezdő-, végpontjában	m	+		PGsl
NOT	K	Logikai NEM (negálás)		+		PGAsl
NPROT	P	Gép-specifikus védőtartomány BE/KI		+	-	PGAsl
NPROTDEF	P	Egy gép-specifikus védőtartomány definíciója		+	-	PGAsl
NUMBER	F	Bemenet stringet számmá alakítani		+	-	PGAsl
OEMIPO1	G	OEM interpoláció 1	m	+		PGAsl
OEMIPO2	G	OEM interpoláció 2	m	+		PGAsl
OF	K	CASE elágazás kulcsszó		+		PGAsl
OFFN	A	Ráhagyás a programozott kontúrra	m	+		PGsl
OMA1	A	OEM cím 1	m	+		PGAsl
OMA2	A	OEM cím 2	m	+		PGAsl
OMA3	A	OEM cím 3	m	+		PGAsl
OMA4	A	OEM cím 4	m	+		PGAsl
OMA5	A	OEM cím 5	m	+		PGAsl
OR	K	Logikai művelet, VAGY kapcsolat		+		PGAsl
ORIAXES	G	Gép- vagy tájoló-tengelyek lineáris interpolációja	m	+		PGAsl

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
ORIAXPOS	G	Tájolási szög virtuális tájolási tengellyel kör-tengely-pozíciókkal	m	+		PGAsl
ORIC [6]	G	Tájolás-változások a külső sarkokon a betoldandó kör-mondattal átlapolódnak	m	+		PGAsl
ORICONCCW	G	Interpoláció egy körpalást-felületen az óramutató járásával ellentétes irányban	m	+		PGAsl/FB3sl (F3)
ORICONCW	G	Interpoláció egy körpalást-felületen az óramutató járásának irányában	m	+		PGAsl/FB3sl (F4)
ORICONIO	G	Interpoláció egy körpalást-felületen egy közbenső tájolás megadásával	m	+		PGAsl/FB3sl (F4)
ORICONTO	G	Interpoláció egy körpalást-felületen érintőleges átmenetben (végtájolás megadása)	m	+		PGAsl/FB3sl (F5)
ORICURVE	G	Tájolás interpolációja a szerszám két érintőpontja mozgásának megadásával	m	+		PGAsl/FB3sl (F6)
ORID	G	Tájolás-változások a kör-mondat előtt végrehajtódnak	m	+		PGAsl
ORIEULER [6]	G	Tájolás szög Euler-szöggel	m	+		PGAsl
ORIMKS	G	Szerszámtájolás gép-koordinátarendszerben	m	+		PGAsl
ORIPATH	G	Szerszámtájolás a pályára vonatkoztatva	m	+		PGAsl
ORIPATHS	G	Szerszámtájolás a pályára vonatkoztatva, a tájolás lefutásában egy törés simítva lesz	m	+		PGAsl
ORIPLANE	G	Interpoláció egy síkban (megfelel ORIVECT-nek) nagy-kör interpoláció	m	+		PGAsl
ORIRESET	P	Szerszámtájolás alaphelyzete max. 3 tájolótengelyig		+	-	PGAsl
ORIROTA [6]	G	Forgásszög egy abszolút megadott forgásirányhoz	m	+		PGAsl
ORIROTC	G	Érintőleges forgásvektor a pályaérintőhöz	m	+		PGAsl
ORIROTR	G	Forgásszög a kezdő- és végtájolás síkjához vonatkoztatva	m	+		PGAsl
ORIROTT	G	Forgásszög a tájolásvektor változásához viszonyítva	m	+		PGAsl
ORIRPY	G	Tájolás-szög RPY-szöggel (XYZ)	m	+		PGAsl
ORIRPY2	G	Tájolás-szög RPY-szöggel (ZYX)	m	+		PGAsl
ORIS	A	Tájolás változás	m	+		PGAsl
ORISOF [6]	G	Tájolás lefutás simítása KI	m	+		PGAsl
ORISOLH	F	Tájolásokat kiszámítani		+		PGAsl
ORISON	G	Tájolás lefutás simítása BE	m	+		PGAsl
ORIVECT [6]	G	Nagy-kör interpoláció (azonos ORIPLANE-nel)	m	+		PGAsl
ORIVIRT1	G	Tájolás-szög virtuális tájolási tengellyel (definíció 1)	m	+		PGAsl

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
ORIVIRT2	G	Tájolás-szög virtuális tájolási tengellyel (definíció 1)	m	+		PGAsl
ORIWKS [6]	G	Szerszámtájolás munkadarab-koordináta-rendszerben	m	+		PGAsl
OS	K	Ingázás be/ki		+		PGAsl
OSB	K	Ingázás: kezdőpont	m	+		FB1sl (P5)
OSC	G	Szerszám-tájolás állandó simítás	m	+		PGAsl
OSCILL	K	Tengely: 1 - 3 fogásvételi tengely	m	+		PGAsl
OSCTRL	K	Ingázás opciók	m	+		PGAsl
OSD	G	Szerszámtájolás átsimítása az átsimítási utak megadásával SD	m	+		PGAsl
OSE	K	Ingázás végpont	m	+		PGAsl
OSNSC	K	Ingázás: kiszikráztatás száma	m	+		PGAsl
OSOF [6]	G	Szerszámtájolás simítása KI	m	+		PGAsl
OSP1	K	Ingázás: bal fordulópont	m	+		PGAsl
OSP2	K	Ingázás jobb fordulópont	m	+		PGAsl
OSS	G	Szerszámtájolás simítása mondatvégen	m	+		PGAsl
OSSE	G	Szerszámtájolás simítása mondatkezdeten és mondatvégen	m	+		PGAsl
OST	G	Szerszámtájolás átsimítása a szögtűrés megadásával fokban SD-vel (maximális eltérés a programozott tájolás-lefutástól)	m	+		PGAsl
OST1	K	Ingázás: megálláspont a bal fordulópontban	m	+		PGAsl
OST2	K	Ingázás: megálláspont a jobb fordulópontban	m	+		PGAsl
OTOL	A	Kompresszor funkciók, tájolás-simítások és átsimítási módok tájolás-tűrése	m	+		PGAsl
OVR	K	Fordulatszám korrekció	m	+		PGAsl
OVRA	K	Tengely fordulatszám korrekció	m	+	+	PGAsl
OVRRAP	K	Gyorsmenet korrekció	m	+		PGAsl
P	A	Alprogram átfutások száma		+		PGAsl
PAROT	G	Munkadarab-koordinátarendszert a munkadarabon beállítani	m	+		PGsl
PAROTOF [6]	G	Munkadarabra vonatkoztatott frame-forgatást kikapcsolni	m	+		PGsl
PCALL	K	Alprogram felhívása abszolút ág-megadással és paraméter-átadással		+		PGAsl
PDELAYOF	G	Késleltetés lyukasztásnál KI	m	+		PGAsl
PDELAYON [6]	G	Késleltetés lyukasztásnál BE	m	+		PGAsl
PHI	K	Tájolás forgásszöge a kúp iránytengelye körül		+		PGAsl
PHU	K	Egy változó fizikai egysége		+		PGAsl
PL	A	1. B-Spline: csomópont távolság 2. polinom interpoláció: paraméter-intervallum hossza polinom interpolációnál	s	+		PGAsl

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
PM	K	percenként		+		PGsl
PO	K	Polinom együttható polinom interpolációnál	s	+		PGAsl
POCKET3	C (T)	Négyszögzseb marása		+		PGAsl
POCKET4	C (T)	Körzseb marása		+		PGAsl
POLF	K	LIFTFAST visszahúzási pozíció	m	+		PGsl/PGAsl
POLFA	P	Egyes-tengelyek visszahúzási pozícióját $AA_ESR_TRIGGER-rel indítani	m	+	+	PGsl
POLFMASK	P	Tengelyeket a visszahúzásra a tengelyek közötti összefüggés nélkül engedélyezni	m	+	-	PGsl
POLFMLIN	P	Tengelyeket a visszahúzásra a tengelyek közötti összefüggéssel engedélyezni	m	+	-	PGsl
POLY	G	Polinom-interpoláció	m	+		PGAsl
POLYPATH	P	Polinom interpoláció választható az AXIS vagy VECT tengelycsoportokra	m	+	-	PGAsl
PON	G	Lyukasztás BE	m	+		PGAsl
PONS	G	Lyukasztás BE IPO ütemben	m	+		PGAsl
POS	K	Tengely pozicionálása		+	+	PGsl
POSA	K	Tengely pozicionálása mondathatáron át		+	+	PGsl
POSM	P	Tárat pozicionálni		+	-	FBWsl
POSMT	P	Multitool-t szerszámtartón helyszámra pozicionálni		+	-	FBWsl
POSP	K	Pozícionálás szakaszonként (ingázás)		+		PGsl
POSRANGE	F	Megállapítani, hogy egy tengely aktuálisan interpolált parancs-pozíciója egy megadott referencia-pozíció körül egy ablakon belül van-e		+	+	FBSYsl
POT	F	Négyzet (aritmetikai függvény)		+	+	PGAsl
PR	K	fordulatonként		+		PGsl
PREPRO	PA	Alprogramokat előkészítéssel megjelölni		+		PGAsl
PRESETON	P	Valósérték beállítása a referencia állapot elvesztésével		+	+	PGAsl
PRESETONS	P	Valósérték beállítása a referencia állapot elvesztése nélkül		+	+	PGAsl
PRIO	K	Kulcsszó az interruptok prioritásainak beállítására		+		PGAsl
PRLOC	K	Változók inicializálása Reset-nél csak helyi változások után		+		PGAsl
PROC	K	Egy program első utasítása		+		PGAsl
PROTA	P	Ütközés modell újra számítását igényelni		+		PGAsl
PROTD	F	Két védőtartomány távolságát kiszámítani		+		PGAsl
PROTS	P	Védelmi tartomány állapotának beállítása		+		PGAsl
PSI	K	Kúp nyílásszöge		+		PGAsl
PTP	G	Pontról pontra mozgás (PTP mozgás)	m	+		PGAsl

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
PTPG0	G	Pontról pontra mozgás csak G0-nál, különben CP pályamozgás	m	+		PGAsl
PTPWOC	G	Pontról pontra mozgás kiegyenlítő mozgások nélkül, amelyeket a tájolás változások okoznak	m	+		PGAsl
PUNCHACC	P	Útfüggő gyorsítás sapkázásnál		+	-	PGAsl
PUTFTOC	P	Szerszám finomkorrekció párhuzamos lehúzáshoz		+	-	PGAsl
PUTFTOCF	P	Szerszám finomkorrekció egy FCtDEF-fel megadott funkciótól függően párhuzamos lehúzáshoz		+	-	PGAsl
PW	A	B-Spline, pont-súly	s	+		PGAsl
QU	K	Gyors kiegészítő-(segéd-)funkció kiadás		+		PGsl
R...	A	Számítási paraméter beállítható tengelyjelölőként is és numerikus bővítéssel		+		PGAsl
RAC	K	Abszolút mondatonkénti tengely-specifikus sugár-programozás	s	+		PGsl
RDISABLE	P	Beolvasás tiltás		-	+	FBSYsl
READ	P	Beolvas a megadott fájlba egy vagy több sort és az olvasott információkat egy mezőben teszi el		+	-	PGAsl
REAL	K	Adattípus: lebegőpontos változó előjellel (valós szám)		+		PGAsl
REDEF	K	Rendszerváltozók, felhasználói változók és NC utasítások újra definiálása		+		PGAsl
RELEASE	P	Géptengelyek felszabadítása tengelycseréhez		+	+	PGAsl
REP	K	Kulcsszó egy mező összes elemének inicializálásához ugyanazzal az értékkel		+		PGAsl
REPEAT	K	Egy programhurok ismétlése		+		PGAsl
REPEATB	K	Egy programsor ismétlése		+		PGAsl
REPOSA	G	Újra rámenet a kontúrra lineárisan az összes tengellyel	s	+		PGAsl
REPOSH	G	Újra rámenet a kontúrra félkörben	s	+		PGAsl
REPOSHA	G	Újra rámenet a kontúrra az összes tengellyel; geometria-tengelyek félkörben	s	+		PGAsl
REPOSL	G	Újra rámenet a kontúrra lineárisan	s	+		PGAsl
REPOSQ	G	Újra rámenet a kontúrra negyedkörben	s	+		PGAsl
REPOSQA	G	Újra rámenet a kontúrra egyenesen az összes tengellyel; geometria-tengelyek negyedkörben	s	+		PGAsl
RESETMON	P	Programutasítás parancsérték aktiváláshoz		+	-	FBWsl
RET	P	Alprogramvég		+	+	PGAsl
RETB	P	Alprogramvég		+	+	PGAsl

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
RIC	K	Relatív mondatonkénti tengely-specifikus sugár-programozás	s	+		PGsl
RINDEX	F	Egy karakter indexének meghatározása a bemeneti stringben		+	-	PGAsl
RMB	G	Újra rámenetel a mondat kezdőpontra	m	+		PGAsl
RMBBL	G	Újra rámenetel a mondat kezdőpontra	s	+		PGAsl
RME	G	Újra rámenetel a mondat végpontra	m	+		PGAsl
RMEBL	G	Újra rámenetel a mondat végpontra	s	+		PGAsl
RMI [6]	G	Újra rámenetel a megszakítási pontra	m	+		PGAsl
RMIBL [6]	G	Újra rámenetel a megszakítási pontra	s	+		PGAsl
RMN	G	Újra rámenetel a legközelebbi pályapontra	m	+		PGAsl
RMNBL	G	Újra rámenetel a legközelebbi pályapontra	s	+		PGAsl
RND	A	Kontúr-sarok lekerekítés	s	+		PGsl
RNDM	A	Modális lekerekítés	m	+		PGsl
ROT	G	Programozható forgatás	s	+		PGsl
ROTS	G	Programozható frame-forgatás térszöggel	s	+		PGsl
ROUND	F	Tizedesjegyek kerekítése		+	+	PGAsl
ROUNDUP	F	Beadási érték felkerekítése		+	+	PGAsl
RP	A	Polársugár	m/s	+		PGsl
RPL	A	Forgatás a síkban	s	+		PGsl
RT	K	Frame-adatok hozzáférés paraméter: forgatás		+		PGAsl
RTLIOF	G	G0 lineáris egyenes nélkül (egyes-tengely interpoláció)	m	+		PGsl
RTLION [6]	G	G0 egyenes interpolációval	m	+		PGsl

Utasítások S ... Z

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
S	A	Orsó-fordulatszám (G4, G96/G961 esetében más jelentés)	m/s	+	+	PGsl
SAVE	PA	Jellemző az alprogram hívásoknál az információ mentésére		+		PGAsl
SBLOF	P	Egyes-mondat elnyomás		+	-	PGAsl
SBLON	P	Egyes-mondat elnyomás megszüntetése		+	-	PGAsl
SC	K	Frame-adatok hozzáférés paraméter: skálázás		+		PGAsl
SCALE	G	Programozható skálázás	s	+		PGsl
SCC	K	Egy síktengely szelektív hozzárendelése G96/G961/G962-höz. A tengelyjelölő lehet geometriai-, csatorna vagy géptengely.		+		PGsl
SCPARA	K	Szervo paraméterkészlet programozása		+	+	PGAsl

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
SD	A	Spline-fok	s	+		PGAsl
SET	K	Kulcsszó egy mező összes elemének inicializálásához a felsorolt értékkel		+		PGAsl
SETAL	P	Vészjelzés beállítás		+	+	PGAsl
SETDNO	F	Egy szerszám (T) egy vágóélének (CE) D-szám hozzárendelése		+	-	PGAsl
SETINT	K	Annak megállapítása, hogy melyik interrupt-rutint kell aktiválni, ha egy NC bemenet igaz		+		PGAsl
SETM	P	Jelölő beállítása a saját csatornában		+	+	PGAsl
SETMS	P	Visszakapcsolás a gépadatban meghatározott mester-orsóra		+	-	PGsl
SETMS(n)	P	Orsó n-nek mester-orsóként kell hatni		+		PGsl
SETMTH	P	Mester-szerszámtartó számát beállítani		+	-	FBWsl
SETPIECE	P	Darabszámot minden szerszámra figyelembe venni, amelyek az orsóhoz hozzá vannak rendelve		+	-	FBWsl
SETTA	P	Szerszámot a kopás-egyesülésből aktívvá tenni		+	-	FBWsl
SETTCOR	F	Szerszám-komponensek megváltoztatása az összes peremfeltétel figyelembe vételével		+	-	PGAsl
SETTIA	P	Szerszámot a kopás-egyesülésből nem aktívvá tenni		+	-	FBWsl
SF	A	Indításpont-eltolás menetvágásra	m	+		PGsl
SIN	F	Sinus (trigon. függvény)		+	+	PGAsl
SIRELAY	F	A SIRELIN, SIRELOUT és SIRELTIME által paraméterezett biztonsági funkciókat aktiválni		-	+	FBSIsl
SIRELIN	P	Funkciómodul bemeneti mértékeket inicializálni		+	-	FBSIsl
SIRELOUT	P	Funkciómodul kimeneti mértékeket inicializálni		+	-	FBSIsl
SIRELTIME	P	Funkciómodul időzítést inicializálni		+	-	FBSIsl
SLOT1	C (T)	Hosszvájat		+		PGAsl
SLOT2	C (T)	Körvájat		+		PGAsl
SOFT	G	Rándítás-határolt pályagyorsítás	m	+		PGAsl
SOFTA	P	Rándítás-határolt pályagyorsítást a programozott tengelyekre bekapcsolni		+	-	PGAsl
SON	G	Sapkázás BE	m	+		PGAsl
SONS	G	Sapkázás BE IPO ütemben	m	+		PGAsl
SPATH [6]	G	Pálya-vonatkoztatás FGROUP-tengelyekre ívhossz	m	+		PGAsl
SPCOF	P	Mester-orsó vagy orsó(k) átkapcsolása helyzet-szabályozásról fordulatszám-szabályozásra	m	+	-	PGsl

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
SPCON	P	Mester-orsó vagy orsó(k) átkapcsolása fordulatszám-szabályozásból helyzetszabályozásra	m	+	-	PGAsl
SPI	F	Orsószámot átalakít tengelyjelölőre		+	-	PGAsl
SPIF1 [6]	G	Gyors NC be-/kimenetek lyukasztásra/sapkázásra bájt 1	m	+		FB2sl (N4)
SPIF2	G	Gyors NC be-/kimenetek lyukasztásra/sapkázásra bájt 2	m	+		FB2sl (N4)
SPLINEPATH	P	Spline-egyesülés meghatározása		+	-	PGAsl
SPN	A	Rész-szakaszok mondatonkénti száma	s	+		PGAsl
SPOF [6]	G	Löket KI, lyukasztás, sapkázás KI	m	+		PGAsl
SPOS	K	Orsópozíció	m	+	+	PGsl
SPOSA	K	Orsópozíció a mondathatárokon túl	m	+		PGsl
SPP	A	Egy rész-szakasz hossza	m	+		PGAsl
SPRINT	F	Egy formattált bemeneti stringet ad vissza.		+		PGAsl
SQRT	F	Négyzetgyökvonás (aritmetikai funkció) (square root)		+	+	PGAsl
SR	A	Ingázás visszahúzás út szinkronakciókra	s	+		PGsl
SRA	K	Ingázás visszahúzás út külső bemenetnél axiálisan szinkronakciókra	m	+		PGsl
ST	A	Ingázás kiszikrázási idő szinkron-akciókra	s	+		PGsl
STA	K	Ingázás kiszikrázási idő axiális szinkron-akciókra	m	+		PGsl
START	P	Kiválasztott programok egyidejű indítása több csatornában a futó programból		+	-	PGAsl
STARTFIFO [6]	G	Feldolgozás; ezzel párhuzamosan az előrefutás-puffer feltöltése	m	+		PGAsl
STAT		Csukló állása	s	+		PGAsl
STOLF	K	G0 tűréstényező	m	+		PGAsl
STOPFIFO	G	A megmunkálás leállítása; Az előfutás-puffer feltöltése addig, amíg STARTFIFO felismerésre kerül, előfutás-puffer teli vagy programvég	m	+		PGAsl
STOPRE	P	Előrefutás-állj, amíg az összes előkészített mondat a főfutamban feldolgozásra kerül		+	-	PGAsl
STOPREOF	P	Előrefutás-álljt megszünteti		-	+	FBSYsl
STRING	K	Adattípus: karakterlánc		+		PGAsl
STRINGIS	F	Megvizsgálja az NC-nyelvi terjedelmet és speciálisan ehhez az utasításhoz tartozó NC-ciklus neveket, felhasználói változókat, makrókat és címkeneveket, hogy ezek léteznek, érvényesek, definiáltak vagy aktívak-e.		+	-	PGAsl

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
STRLEN	F	Egy string hosszának meghatározása		+	-	PGAsl
SUBSTR	F	Egy karakter indexének meghatározása a bemeneti stringben		+	-	PGAsl
SUPA	G	Az aktuális nullaponteltolás elnyomása, beleértve a programozott eltolásokat, rendszerframe-eket, kézikerék-eltolásokat (DRF), külső nullapont-eltolást és átlapolt mozgást	s	+		PGsl
SVC	K	Szerszám vágósebesség	m	+		PGsl
SYNFCT	P	Egy polinom kiértékelése egy feltételtől függően a mozgás-szinkron akcióban		-	+	FBSYsl
SYNR	K	Változó szinkron olvasása, vagyis a megmunkálási időpontban		+		PGAsl
SYNRW	K	Változó szinkron írása és olvasása, vagyis a megmunkálási időpontban		+		PGAsl
SYNW	K	Változó szinkron írása, vagyis a megmunkálási időpontban		+		PGAsl
T	A	Szerszám felhívás (csere csak akkor, ha ez a gépadatban meghatározott; különben M6 utasítás szükséges)		+		PGsl
TAN	F	Tangens (trigon. függvény)		+	+	PGAsl
TANG	P	Érintőleges vezérlés: csatolást definiálni		+	-	PGAsl
TANGDEL	P	Érintőleges vezérlés: csatolást törölni		+	-	PGAsl
TANGOF	P	Érintőleges vezérlés: csatolást kikapcsolni		+	-	PGAsl
TANGON	P	Érintőleges vezérlés: csatolást bekapcsolni		+	-	PGAsl
TCA (828D: _TCA)	P	Szerszámválasztás / szerszámváltás a szerszám állapotától független		+	-	FBWsl
TCARR	A	Szerszámtartó ("m" szám) felhívás		+		PGAsl
TCI	P	Szerszámcsere a tár közbenső tárolójából		+	-	FBWsl
TCOABS [6]	G	Szerszámhossz-komponensek kiszámítása az aktuális szerszámtájolásból meghatározni	m	+		PGAsl
TCOFR	G	Szerszámhossz-komponensek meghatározása az aktív frame orientálásból	m	+		PGAsl
TCOFRX	G	Egy aktív frame tájolását a szerszám választásnál meghatározni, szerszám X irányba mutat	m	+		PGAsl
TCOFRY	G	Egy aktív frame tájolását a szerszám választásnál meghatározni, szerszám Y irányba mutat	m	+		PGAsl
TCOFRZ	G	Egy aktív frame tájolását a szerszám választásnál meghatározni, szerszám Z irányba mutat	m	+		PGAsl
THETA	A	Forgásszög	s	+		PGAsl
TILT	A	Oldalszög	m	+		PGAsl
TLIFT	P	Érintőleges vezérlés: közbenső mondat generálást bekapcsolni		+	-	PGAsl

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
TML	P	Szerszám választás tár helyszámmal		+	-	FBWsl
TMOF	P	Szerszám-felügyelet lekapcsolás		+	-	PGAsl
TMON	P	Szerszám-felügyelet bekapcsolás		+	-	PGAsl
TO	K	FOR számláló-hurok végértékét jelöli		+		PGAsl
TOFF	K	Szerszámhossz-offset a szerszámhossz-komponensek irányában, ami az indexben megadott geometria-tengellyel párhuzamosan hat.	m	+		PGsl
TOFFL	K	Szerszámhossz-offset az L1, L2 ill. L3 szerszámhossz-komponensek irányában	m	+		PGsl
TOFFOF	P	Online szerszámhossz-korrekció kikapcsolása		+	-	PGAsl
TOFFON	P	Online szerszámhossz-korrekció aktiválása		+	-	PGAsl
TOFFR	A	Szerszámsugár-offset	m	+		PGsl
TOFRAME	G	MKR Z tengelyét frame forgatással a szerszámtájolással párhuzamosan beállítani	m	+		PGsl
TOFRAMEX	G	MKR X tengelyét frame forgatással a szerszámtájolással párhuzamosan beállítani	m	+		PGsl
TOFRAMEY	G	MKR Y tengelyét frame forgatással a szerszámtájolással párhuzamosan beállítani	m	+		PGsl
TOFRAMEZ	G	mint TOFRAME	m	+		PGsl
TOLOWER	F	Egy string betűit kisbetűvé alakítani		+	-	PGAsl
TOOLENV	F	Minden aktuális állapotot tárolni, amelyeknek a tárolóban eltárolt szerszámadatok kiértékelésénél jelentésük van		+	-	PGAsl
TOOLGNT	F	Egy szerszámcsoport szerszámainak számát megállapítani		+	-	FBWsl
TOOLGT	F	Egy szerszám T-számát egy szerszámcsoportból megállapítani		+	-	FBWsl
TOROT	G	MKR Z tengelyét frame forgatással a szerszámtájolással párhuzamosan beállítani	m	+		PGsl
TOROTOF [6]	G	Frame-forgatások szerszámirányban KI	m	+		PGsl
TOROTX	G	MKR X tengelyét frame forgatással a szerszámtájolással párhuzamosan beállítani	m	+		PGsl
TOROTY	G	MKR Y tengelyét frame forgatással a szerszámtájolással párhuzamosan beállítani	m	+		PGsl
TOROTZ	G	mint TOROT	m	+		PGsl
TOUPPER	F	Egy string betűit nagybetűvé alakítani		+	-	PGAsl
TOWBCS	G	kopásértékek alap-koordinátarendszerben (AKR)	m	+		PGAsl
TOWKCS	G	Kopásértékek a szerszámfej koordináta-rendszerében kinetikai transzformációnál (a GKR-től a szerszámforgatásban különbözik)	m	+		PGAsl
TOWMCS	G	Kopásértékek gép-koordinátarendszerben (GKR)	m	+		PGAsl

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
TOWSTD [6]	G	alapérték korrekciókra szerszámhosszban	m	+		PGAsl
TOWTCS	G	Kopásértékek szerszám-koordinátarendszerben (szerszámtartó vonatkoztatási pont T a szerszámtartó felvevőn)	m	+		PGAsl
TOWWCS	G	Kopásértékek munkadarab-koordinátarendszerben (MKR)	m	+		PGAsl
TR	K	Egy frame változó eltolás komponense		+		PGAsl
TRAANG	P	Ferde tengely transzformáció		+	-	PGAsl
TRACON	P	Kaszkád transzformáció		+	-	PGAsl
TRACYL	P	Henger: palástfelület-transzformáció		+	-	PGAsl
TRAFOOF	P	Csatornában aktív transzformáció kikapcsolása		+	-	PGAsl
TRAFOON	P	Kinematikai láncokkal definiált transzformációt bekapcsolni		+	-	PGAsl
TRAILOF	P	Aszinkron vontatás KI		+	+	PGAsl
TRAILON	P	Aszinkron vontatás BE		+	+	PGAsl
TRANS	G	Abszolút programozható nullaponteltolás	s	+		PGsl
TRANSMIT	P	Polár transzformáció (homlokfelület megmunkálás)		+	-	PGAsl
TRAORI	P	4-, 5-tengelyes transzformáció, általános transzformáció		+	-	PGAsl
TRUE	K	Logikai állandó: igaz		+		PGAsl
TRUNC	F	Tizedesjegyek levágása		+	+	PGAsl
TU		tengelyszög	s	+		PGAsl
TURN	A	Menetszám a csavarvonalra	s	+		PGsl
ULI	K	Változók felső határértéke		+		PGAsl
UNLOCK	P	ID szinkronakciót engedélyezni (technológia ciklust megállítani)		-	+	FBSYsl
UNTIL	K	REPEAT hurok befejezési feltétele		+		PGAsl
UPATH	G	Pálya-vonatkoztatás FGROUP-tengelyekre görbeparaméter	m	+		PGAsl
VAR	K	Kulcsszó: paraméter átadás módja		+		PGAsl
VELOLIM	K	Maximális tengelysebesség csökkentése	m	+		PGAsl
VELOLIMA	K	Követő tengely maximális tengelysebesség csökkentése vagy növelése	m	+	+	PGAsl
WAITC	P	Várakozás, amíg a csatolási mondatváltás kritériuma a tengelyekre/orsókra teljesül		+	-	PGAsl
WAITE	P	Várakozás program végére egy másik csatornában		+	-	PGAsl
WAITENC	P	Várakozás a szinkronizált ill. helyreállított tengelypozíciókra		+	-	PGAsl
WAITM	P	Jelölőre vár a megadott csatornában; előző mondatot pontos állj-jal befejezni.		+	-	PGAsl

Utasítás	Fajta [1]	Jelentés	W [2]	TP [3]	SA [4]	Leírást lásd [5]
[1] [2] [3] [4] [5] Magyarázatokat lásd Jelmagyarázat (Oldal 857).						
WAITMC	P	Jelölőre vár a megadott csatornában; pontos állj csak akkor, ha a többi csatorna a jelölőt még nem érte el.		+	-	PGAsl
WAITP	P	Várakozás pozícionáló tengelyek mozgás végére		+	-	PGsl
WAITS	P	Várakozás az orsópozíció elérésére		+	-	PGsl
WALCS0 [6]	G	MKR munkatér-határolás kikapcsolva	m	+	-	PGsl
WALCS1	G	MKR munkatér-határolás csoport 1 aktív	m	+	-	PGsl
WALCS2	G	MKR munkatér-határolás csoport 2 aktív	m	+	-	PGsl
WALCS3	G	MKR munkatér-határolás csoport 3 aktív	m	+	-	PGsl
WALCS4	G	MKR munkatér-határolás csoport 4 aktív	m	+	-	PGsl
WALCS5	G	MKR munkatér-határolás csoport 5 aktív	m	+	-	PGsl
WALCS6	G	MKR munkatér-határolás csoport 6 aktív	m	+	-	PGsl
WALCS7	G	MKR munkatér-határolás csoport 7 aktív	m	+	-	PGsl
WALCS8	G	MKR munkatér-határolás csoport 8 aktív	m	+	-	PGsl
WALCS9	G	MKR munkatér-határolás csoport 9 aktív	m	+	-	PGsl
WALCS10	G	MKR munkatér-határolás csoport 10 aktív	m	+	-	PGsl
WALIMOF	G	AKR munkatér-határolás KI	m	+	-	PGsl
WALIMON [6]	G	AKR munkatér-határolás BE	m	+	-	PGsl
WHEN	K	Akció egyszer végre lesz hajtva, ha a feltétel egyszer teljesül.		-	+	FBSYsl
WHENEVER	K	Az akció ciklikusan végre lesz hajtva minden interpolációs ütemben, ha a feltétel teljesül.		-	+	FBSYsl
WHILE	K	WHILE programhurok kezdete		+		PGAsl
WRITE	P	Mondatot a fájl-rendszerbe írni. Hozzáfűz egy mondatot a megadott fájl végéhez.		+	-	PGAsl
WRTPR	P	String írása BTSS változóba		+	-	PGsl
X	A	Tengelynév	m/s	+	+	PGsl
XOR	O	Logikai kizáró-VAGY		+		PGAsl
Y	A	Tengelynév	m/s	+	+	PGsl
Z	A	Tengelynév	m/s	+	+	PGsl

jelmagyarázat

1) Utasítás fajtája:

A Cím

Jelölő. amihez egy érték lesz hozzárendelve (pl. OVR=10). Van néhány cím, amelyek érték hozzárendelés nélkül egy funkciót be- vagy kikapcsolnak (pl. CPLON és CPLOF).

C (A) AST ciklus

Előre definiált NC program automatikus utó-optimalizáláshoz AST-vel (= automatikus szervó optimalizálás) Az illesztés a konkrét optimalizálási helyzethez a paraméterekkel történik, amelyek a ciklusnak a felhívásakor kerülnek átadásra.

C (M) Mérőciklus

Előre definiált NC program, amelyben egy adott mérési folyamat, mint pl. egy hengeres munkadarab belső átmérőjének meghatározása, általánosan van programozva. Az illesztés a konkrét mérési helyzethez a paraméterekkel történik, amelyek a ciklusnak a felhívásakor kerülnek átadásra.

C (T) Technológiai ciklus

Előre definiált NC program, amelyben egy adott megmunkálási folyamat, mint pl. egy menet fúrása vagy egy zseb marása általánosan van programozva. Az illesztés a konkrét megmunkálási helyzethez a paraméterekkel történik, amelyek a ciklusnak a felhívásakor kerülnek átadásra.

F Előre definiált függvény (visszatérési értéket ad)

Az előre definiált függvény felhívása lehet operandus egy kifejezésben.

G G utasítás

A G utasítások csoportokba vannak beosztva. Egy mondatban egy csoportból csak egy G utasítás írható. Egy G utasítás lehet modálisan hatásos (a visszahívásáig az azonos csoport egy másik funkciója által), vagy csak arra a mondatra hatásos, amelyikben áll (mondatonként hatásos).

K Kulcsszó

Jelölő. ami meghatározza egy mondat szintaxisát. Egy kulcsszóhoz nincs érték hozzárendelve és egy kulcsszóval nem lehet NC funkciót be-/kikapcsolni.

Példák: Vezérlő szekezetek (IF, ELSE, ENDIF, WHEN, ...), Program lefutás (GOTOB, GOTO, RET …)

O Operator

Az operátor egy matematikai, összehasonlító vagy logikai művelet

P Előre definiált eljárás (visszatérési értéket ad)

PA program jellemző

A program jellemzők egy alprogram definíciós sorának a végén állnak.

```
PROC <programnév>(...) <program jellemző>
```

Ezek határozzák meg az alprogrom lefutás viselkedését.

2) Az utasítások hatásossága:

m modális

s mondatonként

3) Programozhatóság a munkadarabprogramokban:

+ programozható

- nem programozható

M csak a gépgyártó által programozható

[4] Programozhatóság szinkronakciókban:

+	programozható
-	nem programozható
T	csak a technológiai ciklusokban programozható

[5] Utalás a dokumentumra, amely az utasítás részletes leírását tartalmazza:

PGsl	Alapok programozási kézikönyv
PGAsl	Munka-előkészítés programozási kézikönyv
BNMsl	Mérő ciklusok programozási kézikönyv
BHDsl	Esztergálás kezelési kézikönyv
BHFsl	Marás kezelési kézikönyv
FB1sl ( )	Működési kézikönyv Alapfunkciók (az érintett funkció leírás alfanumerikus rövidítése zárójelben)
FB2sl ( )	Működési kézikönyv Bővítő funkciók (az érintett funkció leírás alfanumerikus rövidítése zárójelben)
FB3sl ( )	Működési kézikönyv Különleges funkciók (az érintett funkció leírás alfanumerikus rövidítése zárójelben)
FBSIsl	Működési kézikönyv Safety Integrated
FBSYsl	Működési kézikönyv szinkronakciók
FBWsl	Működési kézikönyv Szerszámkezelés

[6] alapbeállítás programkezdetnél (a vezérlés kiszállítási állapotában, ha másként nincs programozva)

Kép 21-1 Lábjegyzetek jelentése az utasítások táblázatban

21.2 Utasítások: Rendelkezésre állás SINUMERIK 828D esetén

Megjegyzés

Ciklusok

A ciklusok "opcionális"-ként vannak megjelölve, ha liszensz-köteles opcióktól függnek:

- bővített technológia funkciók (cikkszám: 6FC5800-0AP58-0YB0)
- mérőciklusok (cikkszám: 6FC5800-0AP28-0YB0)
- kinematika bemérése (cikkszám: 6FC5800-0AP18-0YB0)
- SINUMERIK köszörülés Advanced (cikkszám: 6FC5800-0AS35-0YB0)

Nem lett megjelölve, ha a ciklusok a "Bővített technológiai funkciók" opcióval bizonyos rendszereken csak rész funkcionalitásúak.

21.2.1 Esztergálás / marás vezérlés változatok

Utasítások A ... C

Utasítás	SINUMERIK 828D							
• alapkivitel • opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
:	•	•	•	•	•	•	•	•
*	•	•	•	•	•	•	•	•
+	•	•	•	•	•	•	•	•
-	•	•	•	•	•	•	•	•
<	•	•	•	•	•	•	•	•
<<	•	•	•	•	•	•	•	•
<=	•	•	•	•	•	•	•	•
=	•	•	•	•	•	•	•	•
>=	•	•	•	•	•	•	•	•
/	•	•	•	•	•	•	•	•
/0 … /7	•	•	•	•	•	•	•	•
A	•	•	•	•	•	•	•	•
A2	-	-	-	-	-	-	-	-
A3	-	-	-	-	-	-	-	-
A4	-	-	-	-	-	-	-	-
A5	-	-	-	-	-	-	-	-
A6	-	-	-	-	-	-	-	-
A7	-	-	-	-	-	-	-	-

Utasítás	SINUMERIK 828D							
● alapkivitel ● opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
ABS	●	●	●	●	●	●	●	●
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	●	●	●	●	●	●	●	●

Utasítás	SINUMERIK 828D							
● alapkivitel ○ opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
AX	●	●	●	●	●	●	●	●
AXCTSWE	-	-	-	-	-	-	-	-
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	-	-	-	-	-	-	csatorna-tengely név	csatorna-tengely név
C3	-	-	-	-	-	-	-	-
C4	-	-	-	-	-	-	-	-
C5	-	-	-	-	-	-	-	-
C6	-	-	-	-	-	-	-	-

Utasítás	SINUMERIK 828D							
● alapkivitel ● opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
C7	-	-	-	-	-	-	-	-
CAC	●	●	●	●	●	●	●	●
CACN	●	●	●	●	●	●	●	●
CACP	●	●	●	●	●	●	●	●
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	-	-	○	-	○	○	-	-

Utasítás	SINUMERIK 828D							
● alapkivitel ○ opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
CONTDCON	●	●	●	●	●	●	●	●
CONTPRON	●	●	●	●	●	●	●	●
CORROF	●	●	●	●	●	●	●	●
CORRTRAFO	-	-	-	-	-	-	-	-
COS	●	●	●	●	●	●	●	●
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	○	○	○	○	○	○	○	○

Utasítás	SINUMERIK 828D							
● alapkivitel ● opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
CPMRESET	○	○	○	○	○	○	○	○
CPMSTART	○	○	○	○	○	○	○	○
CPMVDI	○	○	○	○	○	○	○	○
CPOF	○	○	○	○	○	○	○	○
CPON	○	○	○	○	○	○	○	○
CPRECOF	●	●	●	●	●	●	●	●
CPRECON	●	●	●	●	●	●	●	●
CPRES	○	○	○	○	○	○	○	○
CPROT	●	●	●	●	●	●	●	●
CPROTDEF	●	●	●	●	●	●	●	●
CPSETTYPE	○	○	○	○	○	○	○	○
CPSYNCOP	○	○	○	○	○	○	○	○
CPSYNCOP2	○	○	○	○	○	○	○	○
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	-	-	-	-	-	-	-	-

Utasítás	SINUMERIK 828D							
● alapkivitel ● opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
CTABMSEG	-	-	-	-	-	-	-	-
CTABNO	-	-	-	-	-	-	-	-
CTABNOMEM	-	-	-	-	-	-	-	-
CTABPERIOD	-	-	-	-	-	-	-	-
CTABPOL	-	-	-	-	-	-	-	-
CTABPOLID	-	-	-	-	-	-	-	-
CTABSEG	-	-	-	-	-	-	-	-
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	●	●	●	●	●	●	●	●

Utasítás	SINUMERIK 828D							
● alapkivitel ● opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
CYCLE63	○	○	●	●	●	●	●	●
CYCLE64	○	○	●	●	●	●	●	●
CYCLE70	○	○	●	●	●	●	●	●
CYCLE72	●	●	●	●	●	●	●	●
CYCLE76	●	●	●	●	●	●	●	●
CYCLE77	●	●	●	●	●	●	●	●
CYCLE78	○	○	●	●	●	●	●	●
CYCLE79	○	○	●	●	●	●	●	●
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	○	○	●	●	●	●	●	●

Utasítás	SINUMERIK 828D							
● alapkivitel ● opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
CYCLE930	●	●	●	●	●	●	●	●
CYCLE940	●	●	●	●	●	●	●	●
CYCLE951	●	●	●	●	●	●	●	●
CYCLE952	○	○	●	●	●	●	●	●
CYCLE961	○	○	○	○	○	○	○	○
CYCLE971	○	○	○	○	○	○	○	○
CYCLE973	○	○	○	○	○	○	○	○
CYCLE974	○	○	○	○	○	○	○	○
CYCLE976	○	○	○	○	○	○	○	○
CYCLE977	○	○	○	○	○	○	○	○
CYCLE978	○	○	○	○	○	○	○	○
CYCLE979	○	○	○	○	○	○	○	○
CYCLE982	○	○	○	○	○	○	○	○
CYCLE994	○	○	○	○	○	○	○	○
CYCLE995	○	○	○	○	○	○	○	○
CYCLE996	○	-	○	-	○	○	-	-
CYCLE997	○	○	○	○	○	○	○	○
CYCLE998	○	○	○	○	○	○	○	○
CYCLE4071	-	-	-	-	-	-	-	-
CYCLE4072	-	-	-	-	-	-	-	-
CYCLE4073	-	-	-	-	-	-	-	-
CYCLE4074	-	-	-	-	-	-	-	-
CYCLE4075	-	-	-	-	-	-	-	-
CYCLE4077	-	-	-	-	-	-	-	-
CYCLE4078	-	-	-	-	-	-	-	-
CYCLE4079	-	-	-	-	-	-	-	-

Utasítások D ... F

Utasítás	SINUMERIK 828D							
● alapkivitel ● opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
D	●	●	●	●	●	●	●	●
D0	●	●	●	●	●	●	●	●
DAC	●	●	●	●	●	●	●	●

Utasítás	SINUMERIK 828D							
• alapkivitel • opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
DC	•	•	•	•	•	•	•	•
DCI	•	•	•	•	•	•	•	•
DCM	•	•	•	•	•	•	•	•
DCU	•	•	•	•	•	•	•	•
DEF	•	•	•	•	•	•	•	•
DEFINE	•	•	•	•	•	•	•	•
DEFAULT	•	•	•	•	•	•	•	•
DELAYFSTON	•	•	•	•	•	•	•	•
DELAYFSTOF	•	•	•	•	•	•	•	•
DELDL	•	•	•	•	•	•	•	•
DELDTG	•	•	•	•	•	•	•	•
DELETE	•	•	•	•	•	•	•	•
DELMLOWNER	•	•	•	•	•	•	•	•
DELMLRES	•	•	•	•	•	•	•	•
DELMT	•	•	•	•	•	•	•	•
DELOBJ	-	-	-	-	-	-	-	-
DELT	•	•	•	•	•	•	•	•
DELTC	•	•	•	•	•	•	•	•
DELTOOLENV	•	•	•	•	•	•	•	•
DIACYCOFA	•	•	•	•	•	•	•	•
DIAM90	•	•	•	•	•	•	•	•
DIAM90A	•	•	•	•	•	•	•	•
DIAMCHAN	•	•	•	•	•	•	•	•
DIAMCHANA	•	•	•	•	•	•	•	•
DIAMCYCOF	•	•	•	•	•	•	•	•
DIAMOF	•	•	•	•	•	•	•	•
DIAMOFA	•	•	•	•	•	•	•	•
DIAMON	•	•	•	•	•	•	•	•
DIAMONA	•	•	•	•	•	•	•	•
DIC	•	•	•	•	•	•	•	•
DILF	•	•	•	•	•	•	•	•
DISABLE	•	•	•	•	•	•	•	•
DISC	•	•	•	•	•	•	•	•
DISCL	•	•	•	•	•	•	•	•
DISPLOF	•	•	•	•	•	•	•	•
DISPLON	•	•	•	•	•	•	•	•
DISPR	•	•	•	•	•	•	•	•
DISR	•	•	•	•	•	•	•	•

Utasítás	SINUMERIK 828D							
● alapkivitel ● opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
DISRP	●	●	●	●	●	●	●	●
DITE	●	●	●	●	●	●	●	●
DITS	●	●	●	●	●	●	●	●
DIV	●	●	●	●	●	●	●	●
DL	-	-	-	-	-	-	-	-
DO	●	●	●	●	●	●	●	●
DRFOF	●	●	●	●	●	●	●	●
DRIVE	●	●	●	●	●	●	●	●
DRIVEA	●	●	●	●	●	●	●	●
DYNFINISH	●	●	●	●	●	●	●	●
DYNNORM	●	●	●	●	●	●	●	●
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	●	●	●	●	●	●	●	●

Utasítás	SINUMERIK 828D							
● alapkivitel ● opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
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	●	●	●	●	●	●	●	●
FILETIME	●	●	●	●	●	●	●	●
FINEA	●	●	●	●	●	●	●	●
FL	●	●	●	●	●	●	●	●
FLIN	●	●	●	●	●	●	●	●
FMA	●	●	●	●	●	●	●	●
FNORM	●	●	●	●	●	●	●	●
FOCOF	○	○	○	○	○	○	○	○
FOCON	○	○	○	○	○	○	○	○
FOR	●	●	●	●	●	●	●	●
FP	●	●	●	●	●	●	●	●
FPO	-	-	-	-	-	-	-	-

Utasítás	SINUMERIK 828D							
• alapkivitel • opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
FPR	•	•	•	•	•	•	•	•
FPRAOF	•	•	•	•	•	•	•	•
FPRAON	•	•	•	•	•	•	•	•
FRAME	•	•	•	•	•	•	•	•
FRC	•	•	•	•	•	•	•	•
FRCM	•	•	•	•	•	•	•	•
FROM	•	•	•	•	•	•	•	•
FTOC	•	•	•	•	•	•	•	•
FTOCOF	•	•	•	•	•	•	•	•
FTOCON	•	•	•	•	•	•	•	•
FXS	•	•	•	•	•	•	•	•
FXST	•	•	•	•	•	•	•	•
FXSW	•	•	•	•	•	•	•	•
FZ	•	•	•	•	•	•	•	•

Utasítások G ... L

Utasítás	SINUMERIK 828D							
• alapkivitel ○ opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
G0	•	•	•	•	•	•	•	•
G1	•	•	•	•	•	•	•	•
G2	•	•	•	•	•	•	•	•
G3	•	•	•	•	•	•	•	•
G4	•	•	•	•	•	•	•	•
G5	•	•	•	•	•	•	•	•
G7	•	•	•	•	•	•	•	•
G9	•	•	•	•	•	•	•	•
G17	•	•	•	•	•	•	•	•
G18	•	•	•	•	•	•	•	•
G19	•	•	•	•	•	•	•	•
G25	•	•	•	•	•	•	•	•
G26	•	•	•	•	•	•	•	•
G33	•	•	•	•	•	•	•	•
G34	•	•	•	•	•	•	•	•

Utasítás	SINUMERIK 828D							
● alapkivitel ○ opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
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	●	●	●	●	●	●	●	●
G140	●	●	●	●	●	●	●	●
G141	●	●	●	●	●	●	●	●
G142	●	●	●	●	●	●	●	●
G143	●	●	●	●	●	●	●	●
G147	●	●	●	●	●	●	●	●
G148	●	●	●	●	●	●	●	●
G153	●	●	●	●	●	●	●	●
G247	●	●	●	●	●	●	●	●
G248	●	●	●	●	●	●	●	●

Utasítás	SINUMERIK 828D							
● alapkivitel ○ opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
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	-	-	-	-	-	-	-	-
G931	●	●	●	●	●	●	●	●
G942	●	●	●	●	●	●	●	●
G952	●	●	●	●	●	●	●	●
G961	●	●	●	●	●	●	●	●
G962	●	●	●	●	●	●	●	●
G971	●	●	●	●	●	●	●	●
G972	●	●	●	●	●	●	●	●
G973	●	●	●	●	●	●	●	●

Utasítás	SINUMERIK 828D							
● alapkivitel ○ opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
GEOAX	●	●	●	●	●	●	●	●
GET	●	●	●	●	●	●	●	●
GETACTT	●	●	●	●	●	●	●	●
GETACTTD	●	●	●	●	●	●	●	●
GETD	-	-	-	-	-	○	-	●
GETDNO	●	●	●	●	●	●	●	●
GETEXET	●	●	●	●	●	●	●	●
GETFREELOC	●	●	●	●	●	●	●	●
GETSELT	●	●	●	●	●	●	●	●
GETT	●	●	●	●	●	●	●	●
GETTCOR	●	●	●	●	●	●	●	●
GETTENV	●	●	●	●	●	●	●	●
GETVARAP	●	●	●	●	●	●	●	●
GETVARDFT	●	●	●	●	●	●	●	●
GETVARLIM	●	●	●	●	●	●	●	●
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	●	●	●	●	●	●	●	●

Utasítás	SINUMERIK 828D							
● alapkivitel ○ opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
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	●	●	●	●	●	●	●	●
K	●	●	●	●	●	●	●	●
K1	●	●	●	●	●	●	●	●
KONT	●	●	●	●	●	●	●	●
KONTC	●	●	●	●	●	●	●	●
KONTT	●	●	●	●	●	●	●	●

Utasítás	SINUMERIK 828D							
● alapkivitel ○ opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
KR	●	●	●	●	●	●	●	●
L	●	●	●	●	●	●	●	●
LEAD szerszámtájolás tájolási polinom	- -	- -	- -	- -	- -	- -	- -	- -
LEADOF	-	-	-	●	-	-	●	●
LEADON	-	-	-	●	-	-	●	●
LENTOAX	●	●	●	●	●	●	●	●
LFOF	●	●	●	●	●	●	●	●
LFON	●	●	●	●	●	●	●	●
LFPOS	●	●	●	●	●	●	●	●
LFTXT	●	●	●	●	●	●	●	●
LFWP	●	●	●	●	●	●	●	●
LIFTFAST	●	●	●	●	●	●	●	●
LIMS	●	●	●	●	●	●	●	●
LLI	●	●	●	●	●	●	●	●
LN	●	●	●	●	●	●	●	●
LOCK	●	●	●	●	●	●	●	●
LONGHOLE	●	●	●	●	●	●	●	●
LOOP	●	●	●	●	●	●	●	●

Utasítások M ... R

Utasítás	SINUMERIK 828D							
● alapkivitel ● opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
M0	●	●	●	●	●	●	●	●
M1	●	●	●	●	●	●	●	●
M2	●	●	●	●	●	●	●	●
M3	●	●	●	●	●	●	●	●
M4	●	●	●	●	●	●	●	●
M5	●	●	●	●	●	●	●	●
M6	●	●	●	●	●	●	●	●
M17	●	●	●	●	●	●	●	●

Utasítás	SINUMERIK 828D							
• alapkivitel • opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
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	•	•	•	•	•	•	•	•
NC	•	•	•	•	•	•	•	•
NEWCONF	•	•	•	•	•	•	•	•
NEWMT	•	•	•	•	•	•	•	•
NEWT	•	•	•	•	•	•	•	•
NORM	•	•	•	•	•	•	•	•

Utasítás	SINUMERIK 828D							
• alapkivitel • opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
NOT	•	•	•	•	•	•	•	•
NPROT	•	•	•	•	•	•	•	•
NPROTDEF	•	•	•	•	•	•	•	•
NUMBER	•	•	•	•	•	•	•	•
OEMIPO1	-	-	-	-	-	-	-	-
OEMIPO2	-	-	-	-	-	-	-	-
OF	•	•	•	•	•	•	•	•
OFFN	•	•	•	•	•	•	•	•
OMA1	-	-	-	-	-	-	-	-
OMA2	-	-	-	-	-	-	-	-
OMA3	-	-	-	-	-	-	-	-
OMA4	-	-	-	-	-	-	-	-
OMA5	-	-	-	-	-	-	-	-
OR	•	•	•	•	•	•	•	•
ORIAXES	-	-	-	-	-	-	-	-
ORIAXPOS	-	-	-	-	-	-	-	-
ORIC	-	-	-	-	-	-	-	-
ORICONCCW	-	-	-	-	-	-	-	-
ORICONCW	-	-	-	-	-	-	-	-
ORICONIO	-	-	-	-	-	-	-	-
ORICONTO	-	-	-	-	-	-	-	-
ORICURVE	-	-	-	-	-	-	-	-
ORID	-	-	-	-	-	-	-	-
ORIEULER	-	-	-	-	-	-	-	-
ORIMKS	-	-	-	-	-	-	-	-
ORIPATH	-	-	-	-	-	-	-	-
ORIPATHS	-	-	-	-	-	-	-	-
ORIPLANE	-	-	-	-	-	-	-	-
ORIRESET	-	-	-	-	-	-	-	-
ORIROTA	-	-	-	-	-	-	-	-
ORIROTC	-	-	-	-	-	-	-	-
ORIROTR	-	-	-	-	-	-	-	-
ORIROTT	-	-	-	-	-	-	-	-
ORIRPY	-	-	-	-	-	-	-	-
ORIRPY2	-	-	-	-	-	-	-	-
ORIS	-	-	-	-	-	-	-	-
ORISOF	-	-	-	-	-	-	-	-
ORISOLH	-	-	-	-	-	-	-	-

Utasítás	SINUMERIK 828D							
● alapkivitel ● opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
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	●	●	●	●	●	●	●	●

Utasítás	SINUMERIK 828D							
● alapkivitel ● opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
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	●	●	●	●	●	●	●	●

Utasítás	SINUMERIK 828D							
• alapkivitel • opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
REAL	•	•	•	•	•	•	•	•
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	•	•	•	•	•	•	•	•

Utasítások S ... Z

Utasítás	SINUMERIK 828D							
● alapkivitel ○ opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
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	●	●	●	●	●	●	●	●

Utasítás	SINUMERIK 828D							
● alapkivitel ○ opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
SPI	●	●	●	●	●	●	●	●
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	-	-	-	-	-	-	-	-

Utasítás	SINUMERIK 828D							
● alapkivitel ○ opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
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	●	●	●	●	●	●	●	●
TOFRAMEY	●	●	●	●	●	●	●	●
TOFRAMEZ	●	●	●	●	●	●	●	●
TOLOWER	●	●	●	●	●	●	●	●
TOOLENV	●	●	●	●	●	●	●	●
TOOLGNT	●	●	●	●	●	●	●	●
TOOLGT	●	●	●	●	●	●	●	●
TOROT	●	●	●	●	●	●	●	●
TOROTOF	●	●	●	●	●	●	●	●
TOROTX	●	●	●	●	●	●	●	●
TOROTY	●	●	●	●	●	●	●	●
TOROTZ	●	●	●	●	●	●	●	●
TOUPPER	●	●	●	●	●	●	●	●
TOWBCS	●	-	●	-	●	●	-	-
TOWKCS	●	-	●	-	●	●	-	-

Utasítás	SINUMERIK 828D							
● alapkivitel ○ opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
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	●	●	●	●	●	●	●	●
VELOLIMA	●	●	●	●	●	●	●	●
WAITC	●	●	●	●	●	●	●	●
WAITE	-	-	-	-	-	○	-	●
WAITENC	●	●	●	●	●	●	●	●
WAITM	-	-	-	-	-	○	-	●
WAITMC	-	-	-	-	-	○	-	●
WAITP	●	●	●	●	●	●	●	●
WAITS	●	●	●	●	●	●	●	●
WALCS0	●	●	●	●	●	●	●	●
WALCS1	●	●	●	●	●	●	●	●
WALCS2	●	●	●	●	●	●	●	●
WALCS3	●	●	●	●	●	●	●	●
WALCS4	●	●	●	●	●	●	●	●

Utasítás	SINUMERIK 828D							
● alapkivitel ○ opció - nem áll rendelkezésre	SW24x(5) CNC-SW marás export (me42)	SW24x(5) CNC-SW esztergálás export (te42)	SW26x(3) CNC-SW marás export (me62)	SW26x(3) CNC-SW esztergálás export (te62)	SW28x(2) CNC-SW marás export (me821)	SW28x(1) CNC-SW marás Adv. export (me822)	SW28x(2) CNC-SW esztergálás export (te821)	SW28x(1) CNC-SW esztergálás Adv. export (te822)
WALCS5	●	●	●	●	●	●	●	●
WALCS6	●	●	●	●	●	●	●	●
WALCS7	●	●	●	●	●	●	●	●
WALCS8	●	●	●	●	●	●	●	●
WALCS9	●	●	●	●	●	●	●	●
WALCS10	●	●	●	●	●	●	●	●
WALIMOF	●	●	●	●	●	●	●	●
WALIMON	●	●	●	●	●	●	●	●
WHEN	●	●	●	●	●	●	●	●
WHENEVER	●	●	●	●	●	●	●	●
WHILE	●	●	●	●	●	●	●	●
WORKPIECE	●	●	●	●	●	●	●	●
WRITE	●	●	●	●	●	●	●	●
WRTPR	●	●	●	●	●	●	●	●
X	●	●	●	●	●	●	●	●
XOR	●	●	●	●	●	●	●	●
Y	●	●	●	●	●	●	●	●
Z	●	●	●	●	●	●	●	●

21.2.2 Köszörülés vezérlés változat

Utasítások A ... C

Utasítás	SINUMERIK 828D					
● alapkivitel ● opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
:	●	●	●	●	●	●
*	●	●	●	●	●	●
+	●	●	●	●	●	●
-	●	●	●	●	●	●
<	●	●	●	●	●	●
<<	●	●	●	●	●	●
<=	●	●	●	●	●	●

Utasítás	SINUMERIK 828D					
• alapkivitel • opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech 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	•	•	•	•	•	•
AP	•	•	•	•	•	•
APR	•	•	•	•	•	•
APRB	•	•	•	•	•	•
APRP	•	•	•	•	•	•
APW	•	•	•	•	•	•
APWB	•	•	•	•	•	•
APWP	•	•	•	•	•	•
APX	•	•	•	•	•	•
AR	•	•	•	•	•	•
AROT	•	•	•	•	•	•
AROTS	•	•	•	•	•	•
AS	•	•	•	•	•	•

Utasítás	SINUMERIK 828D					
● alapkivitel ● opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
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	●	●	●	●	●	●
B_OR	●	●	●	●	●	●
B_NOT	●	●	●	●	●	●
B_XOR	●	●	●	●	●	●
BAUTO	○	○	○	○	○	○
BLOCK	●	●	●	●	●	●
BLSYNC	●	●	●	●	●	●
BNAT	○	○	○	○	○	○
BOOL	●	●	●	●	●	●
BOUND	●	●	●	●	●	●
BRISK	●	●	●	●	●	●
BRISKA	●	●	●	●	●	●
BSPLINE	○	○	○	○	○	○

Utasítás	SINUMERIK 828D					
● alapkivitel ● opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
BTAN	○	○	○	○	○	○
C	●	●	●	●	●	●
C2	-	-	csatorna-ten-gely név	-	-	-
C3	-	-	-	-	-	-
C4	-	-	-	-	-	-
C5	-	-	-	-	-	-
C6	-	-	-	-	-	-
C7	-	-	-	-	-	-
CAC	●	●	●	●	●	●
CACN	●	●	●	●	●	●
CACP	●	●	●	●	●	●
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	●	●	●	●	●	●

Utasítás	SINUMERIK 828D					
● alapkivitel ● opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
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	○	○	○	○	○	○
CPFRS	○	○	○	○	○	○
CPLA	○	○	○	○	○	○
CPLCTID	○	○	○	○	○	○
CPLDEF	○	○	○	○	○	○
CPLDEL	○	○	○	○	○	○
CPLDEN	○	○	○	○	○	○
CPLINSC	○	○	○	○	○	○
CPLINTR	○	○	○	○	○	○
CPLNUM	○	○	○	○	○	○
CPLOF	○	○	○	○	○	○
CPLON	○	○	○	○	○	○
CPLOUTSC	○	○	○	○	○	○

Utasítás	SINUMERIK 828D					
● alapkivitel ○ opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
CPLOUTTR	○	○	○	○	○	○
CPLPOS	○	○	○	○	○	○
CPLSETVAL	○	○	○	○	○	○
CPMALARM	○	○	○	○	○	○
CPMBRAKE	○	○	○	○	○	○
CPMPRT	○	○	○	○	○	○
CPMRESET	○	○	○	○	○	○
CPMSTART	○	○	○	○	○	○
CPMVDI	○	○	○	○	○	○
CPOF	○	○	○	○	○	○
CPON	○	○	○	○	○	○
CPRECOF	●	●	●	●	●	●
CPRECON	●	●	●	●	●	●
CPRES	○	○	○	○	○	○
CPROT	●	●	●	●	●	●
CPROTDEF	●	●	●	●	●	●
CPSETTYPE	○	○	○	○	○	○
CPSYNCOP	○	○	○	○	○	○
CPSYNCOP2	○	○	○	○	○	○
CPSYNCOV	○	○	○	○	○	○
CPSYNFIP	○	○	○	○	○	○
CPSYNFIP2	○	○	○	○	○	○
CPSYNFIV	○	○	○	○	○	○
CR	●	●	●	●	●	●
CROT	●	●	●	●	●	●
CROTS	●	●	●	●	●	●
CRPL	●	●	●	●	●	●
CSCALE	●	●	●	●	●	●
CSPLINE	○	○	○	○	○	○
CT	●	●	●	●	●	●
CTAB	-	-	-	-	-	-
CTABDEF	-	-	-	-	-	-
CTABDEL	-	-	-	-	-	-
CTABEND	-	-	-	-	-	-
CTABEXISTS	-	-	-	-	-	-
CTABFNO	-	-	-	-	-	-
CTABFPOL	-	-	-	-	-	-
CTABFSEG	-	-	-	-	-	-
CTABID	-	-	-	-	-	-

Utasítás	SINUMERIK 828D					
• alapkivitel • opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
CTABINV	-	-	-	-	-	-
CTABISLOCK	-	-	-	-	-	-
CTABLOCK	-	-	-	-	-	-
CTABMEMTYP	-	-	-	-	-	-
CTABMPOL	-	-	-	-	-	-
CTABMSEG	-	-	-	-	-	-
CTABNO	-	-	-	-	-	-
CTABNOMEM	-	-	-	-	-	-
CTABPERIOD	-	-	-	-	-	-
CTABPOL	-	-	-	-	-	-
CTABPOLID	-	-	-	-	-	-
CTABSEG	-	-	-	-	-	-
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	•	•	•	•	•	•

Utasítás	SINUMERIK 828D					
● alapkivitel ○ opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
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	●	●	●	●	●	●
CYCLE752	●	●	●	●	●	●
CYCLE753	●	●	●	●	●	●
CYCLE754	●	●	●	●	●	●
CYCLE755	●	●	●	●	●	●
CYCLE756	●	●	●	●	●	●
CYCLE757	●	●	●	●	●	●
CYCLE758	●	●	●	●	●	●
CYCLE759	●	●	●	●	●	●
CYCLE800	○	○	○	○	○	○
CYCLE801	-	-	-	-	-	-
CYCLE802	-	-	-	-	-	-
CYCLE830	-	-	-	-	-	-

Utasítás	SINUMERIK 828D					
● alapkivitel ● opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
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	●	●	●	●	●	●
CYCLE4078	●	●	●	●	●	●
CYCLE4079	●	●	●	●	●	●

Utasítások D ... F

Utasítás	SINUMERIK 828D					
• alapkivitel • opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
D	•	•	•	•	•	•
D0	•	•	•	•	•	•
DAC	•	•	•	•	•	•
DC	•	•	•	•	•	•
DCI	•	•	•	•	•	•
DCM	•	•	•	•	•	•
DCU	•	•	•	•	•	•
DEF	•	•	•	•	•	•
DEFINE	•	•	•	•	•	•
DEFAULT	•	•	•	•	•	•
DELAYFSTON	•	•	•	•	•	•
DELAYFSTOF	•	•	•	•	•	•
DELDL	•	•	•	•	•	•
DELDTG	•	•	•	•	•	•
DELETE	•	•	•	•	•	•
DELMLOWNER	•	•	•	•	•	•
DELMLRES	•	•	•	•	•	•
DELMT	-	-	-	-	-	-
DELOBJ	-	-	-	-	-	-
DELT	•	•	•	•	•	•
DELTC	•	•	•	•	•	•
DELTOOLENV	•	•	•	•	•	•
DIACYCOFA	•	•	•	•	•	•
DIAM90	•	•	•	•	•	•
DIAM90A	•	•	•	•	•	•
DIAMCHAN	•	•	•	•	•	•
DIAMCHANA	•	•	•	•	•	•
DIAMCYCOF	•	•	•	•	•	•
DIAMOF	•	•	•	•	•	•
DIAMOFA	•	•	•	•	•	•
DIAMON	•	•	•	•	•	•
DIAMONA	•	•	•	•	•	•
DIC	•	•	•	•	•	•
DILF	•	•	•	•	•	•
DISABLE	•	•	•	•	•	•
DISC	•	•	•	•	•	•
DISCL	•	•	•	•	•	•

Utasítás	SINUMERIK 828D					
● alapkivitel ● opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
DISPLOF	●	●	●	●	●	●
DISPLON	●	●	●	●	●	●
DISPR	●	●	●	●	●	●
DISR	●	●	●	●	●	●
DISRP	●	●	●	●	●	●
DITE	●	●	●	●	●	●
DITS	●	●	●	●	●	●
DIV	●	●	●	●	●	●
DL	-	-	-	-	-	-
DO	●	●	●	●	●	●
DRFOF	●	●	●	●	●	●
DRIVE	●	●	●	●	●	●
DRIVEA	●	●	●	●	●	●
DYNFINISH	●	●	●	●	●	●
DYNNORM	●	●	●	●	●	●
DYNPOS	●	●	●	●	●	●
DYNROUGH	●	●	●	●	●	●
DYNSEMIFIN	●	●	●	●	●	●
DZERO	●	●	●	●	●	●
EAUTO	○	○	○	○	○	○
EGDEF	○	○	○	○	○	○
EGDEL	○	○	○	○	○	○
EGOFC	○	○	○	○	○	○
EGOFS	○	○	○	○	○	○
EGON	○	○	○	○	○	○
EGONSYN	○	○	○	○	○	○
EGONSYNE	○	○	○	○	○	○
ELSE	●	●	●	●	●	●
ENABLE	●	●	●	●	●	●
ENAT	○	○	○	○	○	○
ENDFOR	●	●	●	●	●	●
ENDIF	●	●	●	●	●	●
ENDLABEL	●	●	●	●	●	●
ENDLOOP	●	●	●	●	●	●
ENDPROC	●	●	●	●	●	●
ENDWHILE	●	●	●	●	●	●
ESRR	○	○	○	○	○	○
ESRS	○	○	○	○	○	○
ETAN	○	○	○	○	○	○

Utasítás	SINUMERIK 828D					
● alapkivitel ● opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
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	●	●	●	●	●	●
FILETIME	●	●	●	●	●	●
FINEA	●	●	●	●	●	●
FL	●	●	●	●	●	●
FLIN	●	●	●	●	●	●
FMA	●	●	●	●	●	●
FNORM	●	●	●	●	●	●
FOCOF	○	○	○	○	○	○
FOCON	○	○	○	○	○	○
FOR	●	●	●	●	●	●

Utasítás	SINUMERIK 828D					
• alapkivitel • opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
FP	•	•	•	•	•	•
FPO	-	-	-	-	-	-
FPR	•	•	•	•	•	•
FPRAOF	•	•	•	•	•	•
FPRAON	•	•	•	•	•	•
FRAME	•	•	•	•	•	•
FRC	•	•	•	•	•	•
FRCM	•	•	•	•	•	•
FROM	•	•	•	•	•	•
FTOC	•	•	•	•	•	•
FTOCOF	•	•	•	•	•	•
FTOCON	•	•	•	•	•	•
FXS	•	•	•	•	•	•
FXST	•	•	•	•	•	•
FXSW	•	•	•	•	•	•
FZ	•	•	•	•	•	•

Utasítások G ... L

Utasítás	SINUMERIK 828D					
• alapkivitel • opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
G0	•	•	•	•	•	•
G1	•	•	•	•	•	•
G2	•	•	•	•	•	•
G3	•	•	•	•	•	•
G4	•	•	•	•	•	•
G5	•	•	•	•	•	•
G7	•	•	•	•	•	•
G9	•	•	•	•	•	•
G17	•	•	•	•	•	•
G18	•	•	•	•	•	•
G19	•	•	•	•	•	•
G25	•	•	•	•	•	•
G26	•	•	•	•	•	•
G33	•	•	•	•	•	•

Utasítás	SINUMERIK 828D					
• alapkivitel • opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
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	•	•	•	•	•	•
G140	•	•	•	•	•	•
G141	•	•	•	•	•	•
G142	•	•	•	•	•	•
G143	•	•	•	•	•	•
G147	•	•	•	•	•	•
G148	•	•	•	•	•	•
G153	•	•	•	•	•	•
G247	•	•	•	•	•	•
G248	•	•	•	•	•	•

Utasítás	SINUMERIK 828D					
• alapkivitel • opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
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	-	-	-	-	-	-
G931	•	•	•	•	•	•
G942	•	•	•	•	•	•
G952	•	•	•	•	•	•
G961	•	•	•	•	•	•
G962	•	•	•	•	•	•
G971	•	•	•	•	•	•
G972	•	•	•	•	•	•
G973	•	•	•	•	•	•
GEOAX	•	•	•	•	•	•

Utasítás	SINUMERIK 828D					
• alapkivitel • opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
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	•	•	•	•	•	•
GOTOF	•	•	•	•	•	•
GOTOS	•	•	•	•	•	•
GP	•	•	•	•	•	•
GWPSOF	•	•	•	•	•	•
GROUP_ADDEND	•	•	•	•	•	•
GROUP_BEGIN	•	•	•	•	•	•
GROUP_END	•	•	•	•	•	•
GWPSON	•	•	•	•	•	•
H...	•	•	•	•	•	•
HOLES1	-	-	-	-	-	-
HOLES2	-	-	-	-	-	-
I	•	•	•	•	•	•
I1	•	•	•	•	•	•
IC	•	•	•	•	•	•
ICYCOF	•	•	•	•	•	•
ICYCON	•	•	•	•	•	•
ID	•	•	•	•	•	•
IDS	•	•	•	•	•	•

Utasítás	SINUMERIK 828D					
• alapkivitel • opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
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	•	•	•	•	•	•
K	•	•	•	•	•	•
K1	•	•	•	•	•	•
KONT	•	•	•	•	•	•
KONTC	•	•	•	•	•	•
KONTT	•	•	•	•	•	•
KR	•	•	•	•	•	•
L	•	•	•	•	•	•

Utasítás	SINUMERIK 828D					
● alapkivitel ● opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
LEAD						
Szerszámtájolás	-	-	-	-	-	-
tájolási polinom	-	-	-	-	-	-
LEADOF	-	-	-	-	-	-
LEADON	-	-	-	-	-	-
LENTOAX	●	●	●	●	●	●
LFOF	●	●	●	●	●	●
LFON	●	●	●	●	●	●
LFPOS	●	●	●	●	●	●
LFTXT	●	●	●	●	●	●
LFWP	●	●	●	●	●	●
LIFTFAST	●	●	●	●	●	●
LIMS	●	●	●	●	●	●
LLI	●	●	●	●	●	●
LN	●	●	●	●	●	●
LOCK	●	●	●	●	●	●
LONGHOLE	-	-	-	-	-	-
LOOP	●	●	●	●	●	●

Utasítások M ... R

Utasítás	SINUMERIK 828D					
● alapkivitel ○ opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
M0	●	●	●	●	●	●
M1	●	●	●	●	●	●
M2	●	●	●	●	●	●
M3	●	●	●	●	●	●
M4	●	●	●	●	●	●
M5	●	●	●	●	●	●
M6	●	●	●	●	●	●
M17	●	●	●	●	●	●
M19	●	●	●	●	●	●
M30	●	●	●	●	●	●
M40	●	●	●	●	●	●
M41 ... M45	●	●	●	●	●	●

Utasítás	SINUMERIK 828D					
● alapkivitel ○ opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
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	●	●	●	●	●	●
NEWT	-	-	-	-	-	-
NORM	●	●	●	●	●	●
NOT	●	●	●	●	●	●
NPROT	●	●	●	●	●	●
NPROTDEF	●	●	●	●	●	●
NUMBER	●	●	●	●	●	●
OEMIPO1	-	-	-	-	-	-

Utasítás	SINUMERIK 828D					
● alapkivitel ○ opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
OEMIPO2	-	-	-	-	-	-
OF	●	●	●	●	●	●
OFFN	●	●	●	●	●	●
OMA1	-	-	-	-	-	-
OMA2	-	-	-	-	-	-
OMA3	-	-	-	-	-	-
OMA4	-	-	-	-	-	-
OMA5	-	-	-	-	-	-
OR	●	●	●	●	●	●
ORIAXES	-	-	-	-	-	-
ORIAXPOS	-	-	-	-	-	-
ORIC	-	-	-	-	-	-
ORICONCCW	-	-	-	-	-	-
ORICONCW	-	-	-	-	-	-
ORICONIO	-	-	-	-	-	-
ORICONTO	-	-	-	-	-	-
ORICURVE	-	-	-	-	-	-
ORID	-	-	-	-	-	-
ORIEULER	-	-	-	-	-	-
ORIMKS	-	-	-	-	-	-
ORIPATH	-	-	-	-	-	-
ORIPATHS	-	-	-	-	-	-
ORIPLANE	-	-	-	-	-	-
ORIRESET	-	-	-	-	-	-
ORIROTA	-	-	-	-	-	-
ORIROTC	-	-	-	-	-	-
ORIROTR	-	-	-	-	-	-
ORIROTT	-	-	-	-	-	-
ORIRPY	-	-	-	-	-	-
ORIRPY2	-	-	-	-	-	-
ORIS	-	-	-	-	-	-
ORISOF	-	-	-	-	-	-
ORISOLH	-	-	-	-	-	-
ORISON	-	-	-	-	-	-
ORIVECT	-	-	-	-	-	-
ORIVIRT1	-	-	-	-	-	-
ORIVIRT2	-	-	-	-	-	-
ORIWKS	-	-	-	-	-	-
OS	○	○	○	○	○	○

Utasítás	SINUMERIK 828D					
● alapkivitel ○ opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
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	-	-	-	-	-	-
POLF	●	●	●	●	●	●
POLFA	●	●	●	●	●	●
POLFMASK	●	●	●	●	●	●
POLFMLIN	●	●	●	●	●	●
POLY	-	-	-	-	-	-
POLYPATH	-	-	-	-	-	-
PON	-	-	-	-	-	-

Utasítás	SINUMERIK 828D					
● alapkivitel ○ opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
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	●	●	●	●	●	●
RELEASE	●	●	●	●	●	●
REP	●	●	●	●	●	●
REPEAT	●	●	●	●	●	●
REPEATB	●	●	●	●	●	●
REPOSA	●	●	●	●	●	●
REPOSH	●	●	●	●	●	●
REPOSHA	●	●	●	●	●	●

Utasítás	SINUMERIK 828D					
● alapkivitel ○ opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
REPOSL	●	●	●	●	●	●
REPOSQ	●	●	●	●	●	●
REPOSQA	●	●	●	●	●	●
RESETMON	●	●	●	●	●	●
RET	●	●	●	●	●	●
RETB	●	●	●	●	●	●
RIC	●	●	●	●	●	●
RINDEX	●	●	●	●	●	●
RMB	●	●	●	●	●	●
RME	●	●	●	●	●	●
RMI	●	●	●	●	●	●
RMN	●	●	●	●	●	●
RND	●	●	●	●	●	●
RNDM	●	●	●	●	●	●
ROT	●	●	●	●	●	●
ROTS	●	●	●	●	●	●
ROUND	●	●	●	●	●	●
ROUNDUP	●	●	●	●	●	●
RP	●	●	●	●	●	●
RPL	●	●	●	●	●	●
RT	●	●	●	●	●	●
RTLIOF	●	●	●	●	●	●
RTLION	●	●	●	●	●	●

Utasítások S ... Z

Utasítás	SINUMERIK 828D					
● alapkivitel ○ opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
S	●	●	●	●	●	●
SAVE	●	●	●	●	●	●
SBLOF	●	●	●	●	●	●
SBLON	●	●	●	●	●	●
SC	●	●	●	●	●	●
SCALE	●	●	●	●	●	●
SCC	●	●	●	●	●	●

Utasítás	SINUMERIK 828D					
● alapkivitel ○ opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
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	●	●	●	●	●	●
SPIF1	-	-	-	-	-	-
SPIF2	-	-	-	-	-	-
SPLINEPATH	○	○	○	○	○	○
SPN	-	-	-	-	-	-
SPOF	-	-	-	-	-	-
SPOS	●	●	●	●	●	●
SPOSA	●	●	●	●	●	●
SPP	-	-	-	-	-	-
SPRINT	●	●	●	●	●	●

Utasítás	SINUMERIK 828D					
● alapkivitel ○ opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
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	○	○	○	○	○	○
TCA (828D: _TCA)	●	●	●	●	●	●
TCARR	●	●	●	●	●	●
TCI	●	●	●	●	●	●
TCOABS	●	●	●	●	●	●
TCOFR	●	●	●	●	●	●
TCOFRX	●	●	●	●	●	●
TCOFRY	●	●	●	●	●	●
TCOFRZ	●	●	●	●	●	●
THETA	-	-	-	-	-	-

Utasítás	SINUMERIK 828D					
● alapkivitel ○ opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
TILT	-	-	-	-	-	-
TLIFT	○	○	○	○	○	○
TML	●	●	●	●	●	●
TMOF	●	●	●	●	●	●
TMON	●	●	●	●	●	●
TO	●	●	●	●	●	●
TOFF	●	●	●	●	●	●
TOFFL	●	●	●	●	●	●
TOFFOF	●	●	●	●	●	●
TOFFON	●	●	●	●	●	●
TOFFR	●	●	●	●	●	●
TOFRAME	●	●	●	●	●	●
TOFRAMEX	●	●	●	●	●	●
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	●	●	●	●	●	●

Utasítás	SINUMERIK 828D					
● alapkivitel ○ opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
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	●	●	●	●	●	●
WALCS8	●	●	●	●	●	●
WALCS9	●	●	●	●	●	●
WALCS10	●	●	●	●	●	●
WALIMOF	●	●	●	●	●	●
WALIMON	●	●	●	●	●	●
WHEN	●	●	●	●	●	●
WHENEVER	●	●	●	●	●	●
WHILE	●	●	●	●	●	●
WORKPIECE	●	●	●	●	●	●
WRITE	●	●	●	●	●	●

Utasítás	SINUMERIK 828D					
● alapkivitel ○ opció - nem áll rendelkezésre	SW24x(5) CNC-SW G-tech export (gce42)	SW26x(3) CNC-SW G-tech export (gce62)	SW28x(1) CNC-SW G-tech Adv. export (gce82)	SW24x(5) CNC-SW G-tech export (gse42)	SW26x(3) CNC-SW G-tech export (gse62)	SW28x(1) CNC-SW G-tech Adv. export (gse82)
WRTPR	●	●	●	●	●	●
X	●	●	●	●	●	●
XOR	●	●	●	●	●	●
Y	●	●	●	●	●	●
Z	●	●	●	●	●	●

21.3 Aktuális nyelv a HMI-ben

A következő táblázat tartalmazza a kezelőfelületen rendelkezésre álló összes nyelvet.

Az aktuálisan beállított nyelv a munkadarabprogramban és a szinkronakciókban a következő rendszerváltozókkal kérdezhető le:

$AN_LANGUAGE_ON_HMI = <érték>

<Érték>	Nyelv	Nyelv rövidítés
1	német (Németország)	DEU
2	francia	FRA
3	angol (Egyesült Királyság)	ENG
4	spanyol	ESP
6	olasz	ITA
7	holland	NLD
8	kínai (egyszerűsített)	CHS
9	svéd	SVE
18	magyar	HUN
19	finn	FIN
28	cseh	CSY
50	portugál (Brazília)	PTB
53	lengyel	PLK
55	dán	DAN
57	orosz	RUS
68	szlovák	SKY
72	román	ROM
80	kínai (hagyományos)	CHT
85	koreai	KOR
87	japán	JPN
89	török	TRK

Megjegyzés

A $AN_LANGUAGE_ON_HMI aktualizálása történik:

- rendszer felfutás után
- NC- és/vagy PLC-Reset után.
- az M2N keretében egy másik NC-re átkapcsolás keretében.
- nyelv átkapcsolás után a HMI-n.

Függelék

A

A.1 Rövidítések listája

A	
A	kimenet
ADI4	Analog Drive Interface 4 tengelyre
AC	Adaptive Control
ALM	Active Line Module
ARM	aszinkron forgómotor
AS	automatizálási rendszer
ASCII	American Standard Code for Information Interchange: amerikai kód-szabvány az információ-cserére
ASIC	Application Specific Integrated Circuit: felhasználói áramkör
ASUP	aszinkron alprogram
AUXFU	Auxiliary Function: segédfunkció
AWL	utasításlista
AWP	felhasználói program

B	
BA	üzemmód
BAG	üzemmód-csoport
BCD	Binary Coded Decimals: binérkódban kódolt tizedes számok
BERO	érintés nélküli közelítés-kapcsoló
BI	Binector Input
BICO	Binector Connector
BIN	bináris fájlok: bináris fájlok
BIOS	Basic Input Output System
AKR	alap-koordinátarendszer
BO	Binector Output
BTSS	kezelőtábla interfész

C	
CAD	Computer-Aided Design
CAM	Computer-Aided Manufacturing
CC	Compile Cycle: Compile ciklusok
CEC	Cross Error Compensation
CI	Connector Input
CF kártya	Compact Flash-Card

C	
CNC	Computerized Numerical Control: számítógépes numerikus vezérlés
CO	Connector Output
CoL	Certificate of License
COM	kommunikáció
CPA	Compiler Projecting Data: Compiler beállítási adatok
CRT	Cathode Ray Tube: képcső
CSB	Central Service Board: PLC-modul
CU	Control Unit
CP	Communication Processor
CPU	Central Processing Unit: központi számítóegység
CR	Carriage Return
CTS	Clear To Send: soros adatinterfészeknél az adás-készenlét jelentése
CUTCOM	Cutter Radius Compensation: szerszámsugár-korrekció

D	
DAU	digitál-analóg átalakító
DB	adat-modul (PLC)
DBB	adat-modul bájt (PLC)
DBD	adat-modul dupla-szó (PLC)
DBW	adat-modul szó (PLC)
DBX	adat-modul bit (PLC)
DDE	Dynamic Data Exchange
DDS	Drive Data Set: hajtás adatkészlet
DIN	Deutsche Industrie Norm
DIO	Data Input/Output: adatátvitel kijelzés
DIR	Directory: könyvtár
DLL	Dynamic Link Library
DO	Drive Object
DPM	Dual Port Memory
DPR	Dual Port RAM
DRAM	dinamikus tároló (nem pufferelt)
DRF	Differential Resolver Function: differenciális-forgásjelző-funkció (kézikerék)
DRIVE-CLiQ	Drive Component Link with IQ
DRY	Dry Run: próbafutás előtolás
DSB	Decoding Single Block: egyes mondat dekódolás
DSC	Dynamic Servo Control / Dynamic Stiffness Control
DW	adatszó
DWORD	dupla-szó (aktuális 32 bit)

E	
E	bemenet
EES	Execution from External Storage
E/A	be-/kimenet
ENC	Encoder: valósérték jeladó
EFP	egyszerű periféria modul (PLC ki/be modul)
EGB	elektronikusan veszélyeztetett modul/elem
EMV	elektromágneses összeférhetőség
EN	európai szabvány
ENC	Encoder: valósérték jeladó
EnDat	jeladó interfész
EPROM	Erasable Programmable Read Only Memory: törölhető, elektromosan programozható, csak olvasható tároló
ePS Network Services	Internet támogatású gép táv-felügyelt szolgáltatások
EQN	abszolút jeladó típus jelölése 2048 szinusz jel / fordulattal
ES	Engineering System
ESR	kibővített leállítás és visszahúzás
stb.	ETC billentyű ">"; softkey sáv bővítése azonos menüben

F	
FB	funkció-modul (PLC)
FC	Function Call: funkció-modul (PLC)
FEPROM	Flash–EPROM: olvasható és átírható tároló
FIFO	First In First Out: tároló, amely címmegadás nélkül dolgozik, s amely adatai a betárolással azonos sorrendben kerülnek kiolvassra.
FIPO	finom-interpolátor
FPU	Floating Point Unit: lebegőpontos egység
FRK	marósugár-korrekció
FST	Feed Stop: előtolás állj
FUP	funkció terv (programozási módszer a PLC-re)
FW	Firmware

G	
GC	Global Control (PROFIBUS: Broadcast-Telegramm)
GDIR	globális munkadarabprogram-tároló
GEO	geometria, pl geometriai tengely
GIA	Gear Interpolation Data: hajtómű interpolációs adatok
GND	Signal Ground
GP	alap-program (PLC)
GS	hajtómű fokozat
GSD	készülék alap-adatok egy PROFIBUS Slave leírásához

G	
GSDML	Generic Station Description Markup Language: XML alapú leíró nyelv egy GSD fájl létrehozásához
GUD	Global User Data: globális felhasználói adatok

H	
HEX	hexadecimális számok rövid megnevezése
HiFu	segédfunkció
HLA	hidraulikus lineáris hajtás
HMI	Human Machine Interface: SINUMERIK kezelőfelület
HSA	főorsó-hajtás
HW	hardver

I	
IBN	üzembehelyezés
IKA	interpolatorikus kompenzáció
IM	interfész modul: csatoló-modul
IMR	Interface-Modul Receive: vevőüzemi csatoló-modul
IMS	Interface-Modul Send: adóüzemi csatoló-modul
INC	Increment: lépésmérték
INI	Initializing Data: inicializálási adatok
IPO	interpolátor
ISA	International Standard Architecture
ISO	International Standard Organization

J	
JOG	Jogging: beállító-üzem

K	
K_V	szabályzókör erősítési tényező
K_P	arányos erősítés
$K_Ü$	áttételi viszony
KOP	kontaktus-terv (programozási módszer a PLC-re)

L	
LAI	Logic Machine Axis Image: logikai géptengely-leképezés
LAN	Local Area Network
LCD	Liquid-Crystal Display: folyadékkristály kijelző
LED	Light-Emitting Diode: fény-dióda
LF	Line Feed

L	
LMS	helyzetmérő rendszer
LR	helyzetszabályozó
LSB	Least Significant Bit: legkisebb értékű bit
LUD	Local User Data alkalmazói adatok (helyi)

M	
MAC	Media Access Control
MAIN	Main program: főprogram (OB1, PLC)
MB	megabájt
MCI	Motion Control Interface
MCIS	Motion–Control–Information–System
MCP	Machine Control Panel: gépkezelőhely
MD	gépadat ill. gépadatok
MDA	Manual Data Automatic: kézi beadás
MDS	Motor Data Set: motor adatkészlet
MELDW	jelentés szó
GKR	Gép Koordináta Rendszer
MM	Motor Module
MPF	Main Program File: főprogram (NC)
MSTT	gépkezelőhely

N	
NC	Numerical Control: számjegyes vezérlés mondat-előkészítéssel, elmozdulási tartománnyal stb.
NCU	Numerical Control Unit: NC hardver egység
NRK	NC operációs rendszerének jelölése
NST	interfész jel
NURBS	Non-Uniform Rational B-Spline
NV	nullaponteltolás
NX	Numerical Extension: tengelybővítés modul

O	
OB	szervezési modul a PLC-ben
OEM	Original Equipment Manufacturer
OP	Operation Panel: kezelőberendezés
OPI	Operation Panel Interface: kezelőhely csatlakozás
OPT	Options: opciók
OLP	Optical Link Plug: fénykábel csatlakozó
OSI	Open Systems Interconnection: szabvány számítógépes kommunikációhoz

P	
PAA	kimenetek folyamat-leképezése
PAE	bemenetek folyamat-leképezése
PC	Personal Computer
PCIN	vezérlési adatcsere szoftver neve
PCMCIA	Personal Computer Memory Card International Association: tárolókártya szabvány
PCU	PC Unit: PC-Box (számítógép egység)
PG	programozó-készülék
PKE	paraméter jelölő: egy PKW része
PKW	paraméter jelölő: érték (egy PPO paraméter része)
PLC	Programmable Logic Control: illesztő-vezérlés
PN	PROFINET
PNO	PROFIBUS felhasználók szervezete
PO	POWER ON
POE	program szervezési egység
POS	pozíció/pozícionálás
POSMO A	Positioning Motor Actuator: pozícionáló motor
POSMO CA	Positioning Motor Compact AC: komplett hajtásegység integrált szabályzó- és teljesít-mény-modulokkal, pozícionáló egységgel és programtárolóval, váltóáramú betáplálás
POSMO CD	Positioning Motor Compact DC: mint CA, de egyenáramú betáplálással
POSMO SI	Positioning Motor Servo Integrated: pozícionáló motor; egyenáramú betáplálás
PPO	Parameter Prozessdaten Objekt ;ciklikus adat-távirat PROFIBUS–DP átvitelnél és "Változtatható fordulatszámú hajtások" profilnál
PPU	Panel Processing Unit (egy panel alapú CNC-vezérlés pl. SINUMERIK 828D központi hadvere)
PROFIBUS	Process Field Bus: soros adat-busz
PRT	programteszt
PSW	program vezérlőszó
PTP	Point to Point: pontról pontra
PUD	Program Global User Data: program-globális alkalmazói változók
PZD	folyamat adatok: egy PPO folyam adatok része

Q	
QFK	térnegyed hiba kompenzáció

R	
RAM	Random Access Memory: írható/olvasható tároló
REF	referenciapontra menet funkció
REPOS	újra pozícionálási funkció
RISC	Reduced Instruction Set Computer: processzor-típus kis utasításkészlettel és gyors utasítás-végrehajtással
ROV	Rapid Override: gyorsmenet-korrekció

R	
RP	R-paraméterek, számítási paraméterek, előre definiált felhasználói változók
RPA	R-Parameter Active: tároló-tartomány az NC-ben R paraméter számoknak
RPY	Roll Pitch Yaw: egy koordinátarendszer forgatásjellege
RTLI	Rapid Traverse Linear Interpolation: lineáris interpoláció gyorsmeneti mozgásnál
RTS	Request To Send: adórész bekapcsolása, vezérlőjel a soros adat-interfészről
RTCP	Real Time Control Protocol

S	
SA	szinkronakció
SBC	Safe Break Control: biztos fékvezérlés
SBL	Single Block: egyes-mondat
SBR	Subroutine: alprogram (PLC)
SD	beállítási adat ill. beállítási adatok
SDB	rendszer adatmodul
SEA	Setting Data Active: beállítási adatok adatjelölése (adattípus)
SERUPRO	Search-Run by Program Test: mondatkeresés programteszttel
SFB	rendszer funkció-modul
SFC	System Function Call
SGE	biztonsági bemenet
SGA	biztonsági kimenet
SH	biztos állj
SIM	Single in Line Module
SK	softkey
SKP	Skip: funkció egy munkadarabprogram mondat kihagyáshoz
SLM	szinkron lineár-motor
SM	léptetőmotor
SMC	Sensor Module Cabinet Mounted
SME	Sensor Module Externally Mounted
SMI	Sensor Module Integrated
SPF	Sub Program File: alprogram (NC)
SPS	tároló-programozható vezérlés = PLC
SRAM	statikus tároló (pufferelt)
SRK	vágóélsugár-korrekció
SRM	szinkron forgómotor
SSFK	orsóemelkedési hibakompenzáció
SSI	Serial Synchron Interface: soros szinkron interfész
SSL	mondatkeresés
STW	vezérlőszó
SUG	tárcsa-kerületi sebesség
SW	szoftver
SYF	System Files: rendszerfájlok
SYNACT	Synchronized Action: szinkronakció

T	
TB	Terminal Board (SINAMICS)
TCP	Tool Center Point: szerszámcsúcs
TCP/IP	Transport Control Protocol / Internet Protocol
TCU	Thin Client Unit
TEA	Testing Data Active: jelölés a gépadatokra
TIA	Totally Integrated Automation
TM	Terminal Module (SINAMICS)
TO	Tool Offset: szerszámkorrekció
TOA	Tool Offset Active: jelölés (fájltípus) a szerszámkorrekciókra
TRANSMIT	Transform Milling Into Turning: koordináta-transzformáció maró-megmunkáláshoz egy esztergagépen
TTL	Transistor–Transistor–Logik (interfész típus)
TZ	technológiai ciklus

U	
UFR	User Frame: nullaponteltolás
UP	alprogram
USB	Universal Serial Bus
USV	megszakításmentes áramellátás

V	
VDI	belső kommunikációs interfész az NC és PLC között
VDI	Verein Deutscher Ingenieure
VDE	Verband Deutscher Elektrotechniker
VI	Voltage Input
VO	Voltage Output
VSA	előtoló-hajtás

W	
WAB	puha rá- és lemenet funkció
MKR	Munkadarab Koordináta Rendszer
WKZ	szerszám
WLK	szerszámhossz-korrekció
WOP	műhely-orientált programozás
WPD	Work Piece Directory: munkadarab-könyvtár
WRK	szerszámsugár-korrekció
WZ	szerszám
WZK	szerszámkorrekció
WZV	szerszámkezelés
WZW	szerszámcsere

X	
XML	Extensible Markup Language

Z	
ZOA	Zero Offset Active: jelölés a nullaponteltolásokra
ZSW	állapotszó (hajtásokra)

A.2 Dokumentáció áttekintés

Általános dokumentáció

Reklámanyag
– SINUMERIK 840D sl
– SINUMERIK 828D
– SINUMERIK 828D BASIC

NC 62 katalógus SINUMERIK 840D sl

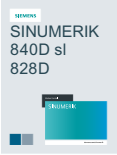
NC 82 katalógus SINUMERIK 828D

PM 21 katalógus SIMOTION, SINAMICS S120

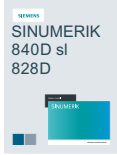
Rendszer kézikönyv Okos szerszámok gépkezelés

Rendszer kézikönyv Ctrl Energy

Felhasználói dokumentáció



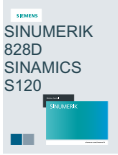
Kezelési kézikönyv
– Univerzális
– Esztergálás
– Marás
– Kösörülés



Programozási kézikönyv
– Alapok
– Munka-előkészítés
– Mérőciklusok

Programozási kézikönyv
– ISO esztergálás
– ISO marás

Diagnózis kézikönyv Vészjelzések



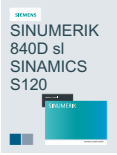
Diagnózis kézikönyv Vészjelzések

Gépgyártó- / szerviz-dokumentáció

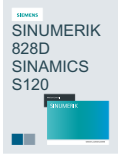
Készülék kézikönyv
– NCU
- Kezelő komponensek és hálózat
– ADI4

Készülék kézikönyv Üzembehelyezési kézikönyv Szerviz kézikönyv

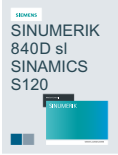
Üzembehelyezési kézikönyv
– CNC: NCK, PLC, hajtás
– alap szoftver és kezelő szoftver



Listák kézikönyv
– Gépadatok
– Interfész jelek
– Változók



Listák kézikönyv
– Gépadatok
– Interfész jelek
– Paraméterek
– Változók



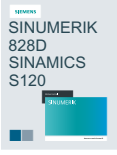
Rendszer kézikönyv Vezérfonal a gép beállításához

Gépgyártó- / szerviz-dokumentáció

Működési kézikönyv
– Alap funkciók
– Bővítő funkciók
– Különleges funkciók
– Szinkronakciók
– ISO dialektusok

Működési kézikönyv Szerszámkezelés

Működési kézikönyv Safety Integrated



Működési kézikönyv Safety Integrated

Beállítási kézikönyv
– Elektromos zavar érzéketlenség felépítési irányelvek
– Industrial Security

Információ / képzés

Képzési anyagok
- Marás egyszerűen ShopMill-lel
- Esztergálás egyszerűen ShopTurn-nel

Kézikönyv Szerszám és forma-készítés

Elektronikus dokumentáció

DOConCD

Industry Online Support (SIOS)

Industry Mall

Fogalmak

Abszolút méret

Egy tengely mozgás céljának megadása egy mérettel, ami a pillanatnyilag érvényes koordinátarendszer nullapontjára vonatkozik. Lásd → láncméret.

Adatátviteli program PCIN

A PCIN egy segédprogram CNC felhasználói adatok, mint pl. munkadarabprogramok, szerszámkorrekciók stb. küldésére és fogadására a soros interfészen. A PCIN program az MS-DOS alatt szabványos ipari PC-ken tud futni.

Adatmodul

1. A → PLC adategysége, amelyhez → HIGHSTEP programok hozzá tudnak férni.
2. Az → NC adategysége: Adatmodulok adat-definíciókat tartalmaznak a globális felhasználói adatokra. Az adatok a definíciónál közvetlenül inicializálhatók.

Adatszó

Két bájt nagyságú adategység egy→ adatmodulban.

Alap-koordinátarendszer

"Cartesius"-féle koordinátarendszer, a gép-koordinátarendszerre transzformációval lesz leképezve.

A → munkadarabprogramban a programozó az alap-koordinátarendszer tengelyneveit használja. Ha nincs aktív → transzformáció, akkor párhuzamosan áll a → gép-koordinátarendszerrel. A különbség a → tengelynevekben van.

Alaptengely

Olyan tengely, amely parancs- vagy valósértéke egy kompenzációs érték kiszámítására bevonásra kerül.

Alprogram

Az alprogram megnevezés még abból az időből származik, amikor a munkadarabprogramok fel voltak osztva → fő- és alprogramokra. Ez a fix felosztás a mai SINUMERIK NC-nyelvben már nincs meg. Elvileg minden munkadarabprogram vagy minden → ciklus egy másik munkadarabprogramon belül alprogramként felhívható. Ekkor az a következő → programszinten (x+1) ((alprogram szint (x+1)) fut le.

Archiválás

Fájlok és/vagy könyvtárak kiolvasása egy **külső** tároló készülékre.

Aszinkron alprogram

Munkadarabprogram, amely az aktív munkadarabprogramhoz aszinkronban (függetlenül) egy interruptjellel (pl. a "gyors NC-bemenet" jel) indítható.

Automatika

A vezérlés üzemmódja (mondatkövető menet a DIN szerint): Üzemmód NC-rendszereknél, amelyben egy → munkadarabprogram kiválasztásra és folyamatos feldolgozásra kerül.

Baudrate

Adatátvitel sebessége (bit/s).

Beállítási-adatok

Olyan adatok, amelyek a szerszámgép tulajdonságait a rendszer-szoftver által meghatározott módon közlik az NC-vezérléssel.

Biztonsági funkciók

A vezérlés állandóan aktív ellenőrzéseket tartalmaz, amelyek a -> CNC-ben, az illesztő-vezérlésben (-> PLC) és a gépben a zavarokat olyan korán felismerik, hogy a munkadarab, szerszám vagy a gép megsérülése messzemenően kizárható. Zavar esetén a megmunkálási folyamat megszakad és a hajtások leállnak, a zavar tárolásra és az vészjelzésként kiadásra kerül. Egyidejűleg közölve lesz a PLC-vel az, hogy egy CNC-vészjelzés áll fenn.

Boot-olás

A rendszerprogram betöltése Power On után.

C tengely

Tengely, amelynél egy vezérelt forgómozgás és pozicionálás történik a munkadarab-orsóval.

Ciklusok

Védett alprogram egy a → munkadarabon ismételten fellépő megmunkálási folyamat végrehajtására.

Cím

Egy cím egy bizonyos operandus vagy operandus-tartomány jelölője, pl. bemenet, kimenet stb.

CNC

Lásd → NC

Computericed Numerical Control: tartalmazza az → NC, → PLC, HMI, → COM komponenseket.

CNC

Lásd → NC

Computericed Numerical Control: tartalmazza az → NC, → PLC, HMI, → COM komponenseket.

COM

Az NC-vezérlés komponensei a kommunikáció végrehajtására és koordinálására.

CPU

Central Processor Unit, lásd → Tároló-programozható vezérlés

C-Spline

A C-Spline a legismertebb és legtöbbet alkalmazott Spline. Az átmenetek a támpontokon érintő- és görbület-állandók. 3. fokú polinomok kerülnek alkalmazásra.

Csatorna

Egy csatornát az jellemez, hogy az a többi csatornától függetlenül egy → munkadarabprogramot fel tud dolgozni. Egy csatorna exkluzív vezérli a hozzárendelt tengelyeket és orsókat. Különböző csatornák munkadarabprogram-lefutásai → szinkronizációval koordinálhatók.

Csavarvonal interpoláció

A csavarvonal interpoláció különösen alkalmas külső- és belsőmenetek egyszerű előállítására forma-marással vagy kenőhorony marására.

Ennél a csavarvonal két mozgásból tevődik össze:

- körmozgás a síkban
- egyenes-mozgás merőlegesen erre a síkra

Diagnózis

1. A vezérlés kezelési tartománya
2. A vezérlésnek van öndiagnózis programja és a szervizhez tesztelési segítség: állapot-vészjelzés-, és szervizkijelzések

DRF

Differential Resolver Function: NC-funkció, amely egy elektronikus kézikerékkel kapcsolatban egy növekményes nullaponteltolást állít elő automatika-üzemben.

Egyenes-interpoláció

A szerszám egy egyenesen a célponthoz megy és eközben megmunkálja a munkadarabot.

Együttfutó tengely

Az együttfutó tengely a → Gantry tengely, amelynek a parancspozíciója a → vezető tengely mozgásából van levezetve és azzal szinkron mozog. A kezelő és a programozó szempontjából az együttfutó tengely "nem létezik".

Elő-koincidencia

Mondatváltás már akkor, amikor a pályaút egy megadott különbséggel megközelített a végpozíciót.

Előtolás-override

A programozott sebesség a → gépi vezérlőtáblán keresztül vagy a → PLC-től az aktuális sebességgel átlapolódik (0-200 %). Az előtolás-sebességet kiegészítőleg a megmunkálási programban egy programozható százalék-tényezővel (1-200 %) helyesbíteni lehet.

Elővezérlés, dinamikus

A → kontúrnak a követési távolságból fakadó pontatlanságait a dinamikus, gyorsulásfüggő elővezérléssel közel meg lehet szüntetni. Ezáltal nagy → pályasebességeknél is kiváló megmunkálási pontosság adódik. Az elővezérlést a → munkadarabprogramban csak az összes tengelyre együtt lehet be- ill. kikapcsolni

Felhasználó által definiált változók

A felhasználók a → munkadarabprogramban vagy adatmodulokban (globális felhasználói adatok) való tetszőleges használatra létrehozhatnak a felhasználó által definiált változókat. Egy definíció az adattípus megadását és a változónevet tartalmazza. Lásd → rendszerváltozók.

Felhasználói program

A felhasználói programok az S7-300-as automatizálási rendszerekre a STEP 7 programnyelvben vannak elkészítve. A felhasználói program modulárisan van felépítve és egyes modulokból áll.

Az alapvető modultípusok a következők:

- Kód-modul
  Ezek a modulok tartalmazzák a STEP-7 utasításokat.
- Adatmodulok
  Ezek a modulok tartalmazzák a STEP-7 programok változóit és állandóit.

Felhasználói tároló

Minden program és adat, mint munkadarabprogramok, alprogramok, kommentárok, szerszámkorrekciók, nullaponteltolások/ framek továbbá csatorna és program felhasználói adatok a közös CNC felhasználói tárolóban eltárolhatók.

Ferde megmunkálás

Fúró- és marómegmunkálásokat olyan munkadarabokon, amelyek nem a gép koordinátasíkjában fekszenek, a "ferde megmunkálás" segítségével lehet kényelmesen végrehajtani.

Fixpontra menet

Szerszámgépek rá tudnak menni megadott fixpontokra, mint pl. szerszámcsere-pont, betöltőpont, paletta-cserepont stb. Ezen pontok koordinátái a vezérlésben vannak letéve. A vezérlés mozgatja az illető tengelyeket, ha lehetséges → gyorsmenetben.

Forgatás

Egy → frame olyan komponensei, amelyek a koordinátarendszernek egy bizonyos szöggel történő elforgatását definiálják.

Föld

Föld a berendezés egymással összekötött nem aktív részeinek összessége, ami még hiba esetén sem lehet veszélyes érintési feszültségű.

Főmondat

":"-vel bevezetett mondat, amely az összes adatot tartalmazza arra, hogy a munkafolyamatot egy → munkadarabprogramban indítani lehessen.

Főprogram

A "főprogram" megnevezés még abból az időből származik, amikor a munkadarabprogramok fel voltak osztva fő- és alprogramokra. Ez a fix felosztás a mai SINUMERIK NC-nyelvben már nincs meg. Elvileg a csatornában minden munkadarabprogramot ki lehet választani és elindítani. Ekkor az a → programszint 0-ban (főprogram szint) fut le. A főprogramban további munkadarabprogramokat vagy → ciklusokat lehet alprogramként felhívni.

Frame

Egy frame egy olyan számítási előírás, amely egy "Cartesius"-féle koordinátarendszert egy másik "Cartesius"-féle koordinátarendszerbe átvezet. Egy Frame a → nullaponteltolás, → forgatás, → skálázás, → tükrözés komponenseit tartalmazza.

Geometria

Egy → munkadarab leírása a → munkadarab-koordinátarendszerben.

Geometria-tengely

A geometria-tengelyek 2 ill. 3 dimenziós → munkadarab koordinátarendszereket képeznek amelyekben a → munkadarabprogramokban van programozva a munkadarab geometriája.

Gép-fixpont

A szerszámgép által egyértelműen meghatározott pont, pl. gép-referenciapont.

Gépkezelőhely

A szerszámgép kezelőtáblája a billentyűk, forgókapcsolók stb. kezelő-elemekkel és egyszerű kijelző-elemekkel, mint pl. LED-ek. Ez a szerszámgép PLC-n keresztüli közvetlen befolyásolására szolgál.

Gép-koordinátarendszer

Koordinátarendszer a szerszámgép tengelyeire vonatkoztatva.

Gép-nullapont

A szerszámgép azon fix pontja, amelyre az összes (levezetett) mérőrendszer visszavezethető.

Géptengelyek

A szerszámgépen fizikailag létező tengelyek.

Görbület

Egy kontúr k görbülete a kontúrpontra illeszkedő kör r sugarának a reciproka (k = 1/r).

Gyors digitális be-/kimenetek

A digitális bemeneteken keresztül pl. gyors CNC program-rutinok (interrupt-rutinok) indíthatók. A digitális CNC-kimeneteken keresztül gyors, programvezérelt kapcsolási funkciók válthatók ki.

Gyors leemelés a kontúrról

Egy interrupt megérkezésénél a CNC-megmunkálási programon keresztül egy mozgás indítható el, amely lehetővé teszi a szerszám gyors leemelését az éppen megmunkált munkadarab-kontúrról. Kiegészítőleg a visszameneteli szög és az út értéke paraméterezhető. Gyors leemelés után kiegészítőleg egy interrupt-rutin hajtható végre.

Gyorsítás rántás-határolással

A gépen egy optimális gyorsítás-viselkedés elérésére, s egyidejűleg a mechanika kímélésére a megmunkálási programban ugrásszerű gyorsítás és folyamatos (rántásnélküli) gyorsítás között át lehet kapcsolni.

Gyorsmenet

Egy tengely leggyorsabb menet sebessége. Ez pl. akkor kerül alkalmazásra, ha a szerszámnak egy nyugalmi állásból rá kell a→ munkadarab-kontúrra mennie vagy vissza kell húzódnia a munkadarab-kontúrról. A gyorsmeneti sebesség gép-specifikusan gépadattal van beállítva.

Hajtás

A hajtás a CNC-nek az a része, amelyik a fordulatszám- és nyomaték-szabályozást az NC utasításai alapján végrehajtja.

Hálózat

A hálózat több S7-300 és további végberendezések, pl.: PG összeköttetése → csatolókábellel. A hálózaton történik az összekötött készülékek között az adatcsere.

Hardver konfiguráció

SIMATIC S7-Tool a hardver komponensek konfigurálására és paraméterezésére egy S7-projektben.

Határfordulatszám

Maximális/minimális (orsó-)fordulatszám: A → PLC vagy a → beállítási adatok gépadat előre megadásával egy orsó maximális fordulatszáma behatárolható.

HIGHSTEP

A programozási lehetőségek összefoglalása az AS300/AS400 rendszer→ PLC-jére.

Hüvelyk mértékrendszer

Olyan mértékrendszer, amely a távolságokat "hüvelykben" és annak tört részeiben definiálja.

Időreciprok előtolás

A tengelymozgásokhoz az előtolási sebesség helyett az időt is lehet programozni, amire a mondatban a pályaút megtételéhez szükség van (G93).

Interpolátor

Az → NC azon logikai egysége, amely a célpozíciók adatai szerint a munkadarabprogramban az egyes tengelyekben megteendő mozgások közbenső értékeit határozza meg.

Interpolatorikus kompenzáció

Az interpolációs kompenzációkkal mint → orsóemelkedés hiba, belógás-, derékszög- és hőmérséklet-kompenzációk a gép mechanikai hibái lesznek kompenzálva.

Interrupt rutinok

Az Interrupt rutinok speciális → alprogramok, amelyeket a megmunkálási folyamat eseményei (külső jelek) indíthatnak. A megmunkálásban levő munkadarabprogram megszakításra kerül, a tengelyek megszakítási pozíciói automatikusan tárolódnak.

Jelentések

Az összes a munkadarabprogramban programozott jelentés és a rendszer által felismert → vészjelzés a kezelőtáblán szöveggel, a dátum és idő és a törlési ismérv megfelelő szimbólumának megadásával kijelzésre kerül. A vészjelzések és a jelentések kijelzése külön történik.

Jelölő

A DIN 66025 szerinti szavak a változókra (számítási változók, rendszerváltozók, felhasználói változók), az alprogramokra, a kulcsszavakra és szavakra jelölők által (nevek) több címbetűvel kiegészülnek. Ezek a kiegészítések jelentésükben azonosak a szavakéval a mondatfelépítésben. Jelölőknek egyértelműeknek kell lenniük. Ugyanazt a jelölőt nem szabad különböző objektumokra alkalmazni.

JOG

A vezérlés üzemmódja (beállító üzem): A JOG üzemmódban be lehet a gépet állítani. Az egyes tengelyeket és orsókat az iránybillentyűk segítségével pillanatindítós üzemben el lehet mozgatni. A JOG üzemmód további funkciói a → referenciapontra menet, → Repos valamint → Preset (valósérték beállítás).

Készdarab-kontúr

A készre megmunkált munkadarab kontúrja. Lásd → nyersdarab.

Kezelőfelület

A kezelőfelület (BOF) egy CNC-vezérlés kijelző eszköze egy képernyő formájában. Ez függőleges és vízszintes softkey-kkel van kialakítva.

Kompenzációs érték

A mérőadó által mért tengelypozíció és a kívánt, programozott tengely-pozíció különbsége.

Kompenzációs táblázat

Támpontok táblázata. A bázistengely kiválasztott pozícióira a kompenzációs tengely kompenzációs értékeit adja.

Kompenzációs tengely

Olyan tengely, amely parancs- vagy valósértéke egy kompenzációs értékkel módosul.

Kontúr

A → munkadarab körvonala

Kontúrfelügyelet

A kontúrhűségre méretként egy meghatározott tűréssávon belül a követési hiba felügyelve lesz.. Egy nem megengedetten nagy követési hiba pl. a hajtás túlterhelése következtében adódhat. Ebben az esetben egy vészjelzés keletkezik és a tengelyek le lesznek állítva.

Koordinátarendszer

Lásd → gép-koordinátarendszer, → munkadarab-koordinátarendszer

Korrekció-tároló

A vezérlés adattartománya, amelyben szerszámkorrekció-adatok kerülnek tárolásra.

Körinterpoláció

A → szerszámnak a kontúr meghatározott pontjai között egy megadott előtolással egy körön kell mozognia és eközben a munkadarabot meg kell munkálnia.

Körtengely

Körtengelyek egy munkadarab- vagy szerszám-elforgatást valósítanak meg egy előre megadott szöghelyzetbe.

Közbenső mondatok

Elmozdulásokat kiválasztott → szerszámkorrekcióva (G41/G42) csak egy korlátozott számú közbenső mondattal (mondatok tengelymozgások nélkül a korrekciós síkban) szabad megszakítani, hogy a szerszámkorrekció még megfelelően legyen kiszámítva. A közbenső mondatok megengedett száma, amelyeket a vezérlés előre beolvas, rendszerparaméterrel állítható be.

Kulcsoskapcsoló

A kulcsoskapcsoló a → gépi kezelőhelyen 4 állással rendelkezik, amelyeket a vezérlés operációs rendszere funkciókkal látott el. A kulcsoskapcsolóhoz három különböző színű kulcs tartozik, amelyek a megadott állásokban húzhatók ki.

Kulcsszavak

Meghatározott írásmódos szavak, amelyeknek a programozói nyelven a → munkadarabprogram számára egy meghatározott jelentésük van.

KÜ

Áttételi viszony

Külső nullaponteltolás

A → PLC által megadott nullaponteltolás.

KV

Körerősítési tényező, egy szabályzókör szabályozástechnikai értéke

Láncméret

Növekmény-méret is Egy tengely mozgáscéljának megadása egy megteendő útszakasszal és iránnyal egy már elért pontra vonatkoztatva. Lásd → abszolút-méret

Lazaság kompenzáció

A mechanikus lazaság kiegyenlítésére szolgál, pl. a golyósorsók irányváltási hibájára. Minden tengelyre meg lehet külön adni a lazaság kompenzációt

Lépésmérték

Elmozdulási úthossz-megadás a növekmény-számmal (lépésmérték) A növekmény-szám → beállítási adatként eltárolható ill. megfelelően feliratozott billentyűkkel (10, 100, 1000, 10000) kiválasztható.

Lineáris tengely

A lineáris tengely egy olyan tengely, amely a körtengellyel szemben egy egyenest ír le.

Look Ahead

A **Look Ahead** funkcióval az elmozdulási mondatoknak egy paraméterezhető számán keresztüli "előtekintése" által optimális megmunkálási sebesség érhető el.

Magasszintű CNC nyelv

A magasszintű nyelv NC programok, → szinkronakciók és → ciklusok írására szolgál. Következőket nyújtja: vezérlő-struktúrák, → felhasználó által definiált változókat, → rendszerváltozókat, → makrótechnikát.

Makrotechnika

Az utasítások sorának összefoglalása egy jelölő alatt. A jelölő a programban az összefoglalt utasítások összességét képviseli.

MDA

A vezérlés üzemmódja: Manual Data Automatic. Az MDA-üzemmódban egyes programmondatok vagy mondat-sorozatok egy fő- vagy alprogram hivatkozás nélkül beadhatók és utána az NC-Start billentyűvel azonnal végrehajthatók.

Megmunkálási csatorna

Egy csatorna-felépítésen keresztül párhuzamos mozgáslefolyásokkal a mellékidők lerövidíthetők, pl. egy töltőportál elmozgatása a megmunkáláshoz szimultán. Egy CNC-csatorna egy önálló CNC-vezérlésként tekintendő dekódolással, mondatelőkészítéssel és interpolációval.

Mellékmondat

"N" által bevezetett mondat információkkal egy munkaszakaszra, pl. egy pozíció megadás.

Menetfúrás kiegyenlítő tokmány nélkül

Ezzel a funkcióval menetet tudunk kiegyenlítő-tokmány nélkül fúrni. Az orsó körtengelykénti és a fúrótengely interpoláló elmozdulása által a menetek pontosan a végfúrás-mélységre vágódnak, pl. zsáklyuk-furatos menet (előfeltétel: az orsó tengelyüzeme).

Méretmegadás metrikus és hüvelyk

A megmunkálási programban a pozíció és emelkedési értékek hüvelykben programozhatók. A programozható méretmegadástól függetlenül (`G70`/`G71`) a vezérlés egy alap méretrendszerre állítható be.

Metrikus mértékrendszer

Az egységek szabványosított rendszere: hosszakra pl. mm (milliméter), m (méter).

Modul

Modulnak nevezzük az összes fájlt, amelyekre a program előállításához és feldolgozásához szükség van.

Mondatkeresés

A munkadarabprogramok teszteléséhez vagy a megmunkálás megszakítása után a "Mondatkeresés" funkción keresztül a munkadarabprogram tetszőleges része felhívható, ahol a megmunkálást folytatni akarjuk.

Mozgási tartomány

A maximálisan megengedett mozgási tartomány lineáris tengelyeknél ± 9 dekád. Az abszolút érték függ a választott beadási és helyzetszabályozási felbontástól és a mértékrendszertől (hüvelyk vagy metrikus).

Munkadarab

A szerszámgép által előállítandó /megmunkálandó darab.

Munkadarab-kontúr

Az előállítandó /megmunkálandó → munkadarab parancs kontúrja.

Munkadarab-koordinátarendszer

A munkadarab-koordinátarendszer kiindulópontja a → munkadarab-nullapontban van. A munkadarab-koordinátarendszerben történő programozásnál a méretek és irányok erre a rendszerre vonatkoznak.

Munkadarab-nullapont

A munkadarab-nullapont a → munkadarab-koordinátarendszer kiindulópontját képezi. Ezt a → gépi nullaponthoz megadott távolságokkal kell meghatározni.

Munkadarabprogram

Olyan utasítások sora az NC-vezérlésre, amelyek összesen egy meghatározott → munkadarab előállítását eredményezik. Ugyancsak egy adott → nyersdarabon egy meghatározott megmunkálás végrehajtása.

Munkadarabprogram kezelés

A munkadarabprogram kezelést a → munkadarabok szerint lehet szervezni. A felhasználói tároló nagysága meghatározza a kezelendő programok és adatok számát. Minden fájlt (programok és adatok) egy max. 24 alfanumerikus jelből álló névvel lehet ellátni.

Munkadarabprogram mondat

Egy → munkadarabprogram része, Line Feed-del határolva. Különbséget teszünk → főmondatok és → mellékmondatok között.

Munkatároló

A munkatároló egy RAM-tároló a → CPU-ban, amelyben a processzor a programfeldolgozás alatt az felhasználói programhoz hozzáfér.

Munkatér

Háromdimenziós tér, amelybe a szerszámcsúcs a szerszámgép szerkezete alapján be tud menni. Lásd → Védőtartomány.

Munkatér-határolás

A munkatér-határolással a tengelyek elmozdulási tartományát a végkapcsolókhoz kiegészítőleg be lehet határolni. Tengelyenként a védett munkatér leírására egy érték-pár lehetséges.

NC

A → CNC számjegyes vezérlésű komponense, amely→ munkadarabprogramokat dolgoz fel és koordinálja a szerszámgép mozgási folyamatait.

Négyzetes hibakompenzáció

Az a kontúrhiba a térnegyed-átmeneteken, amely a vezetőpályákon a váltakozó súrlódási viszonyok által keletkezik, a négyzetes hibakompenzáció messzemenően megszüntethető. A négyzetes hibakompenzáció paraméterezése egy körforma-teszteléssel történik.

NRK

Numeric Robotic Kernel (→ NC operációs rendszere)

Nullaponteltolás

Egy új vonatkoztatási pont megadása egy koordinátarendszerhez egy meglevő nullapontra és egy → frame-re történő vonatkoztatással.

1. Beállítható
   Minden CNC-tengelyre egy megadható számú beállítható nullaponteltolás áll rendelkezésre. A G utasításokkal választható eltolások alternatívan hatnak.
2. Külső
   Az összes, a munkadarab-nullapont helyzetét meghatározó eltoláshoz kiegészítőleg egy külső nullaponteltolás kézikerékkel (DRF-eltolás) vagy a PLC-től átlapoltan valósítható meg.
3. Programozható
   A `TRANS` utasítással az összes pálya- és pozicionáló tengelyre nullaponteltolások programozhatók.

NURBS

A vezérlésen belüli mozgásvezetés és pályainterpoláció NURBS (**N**on **U**niform **R**ational **B**-**S**plines) bázisán kerül végrehajtásra Ezzel a vezérlésen belül az összes interpolációra egy egységes módszer áll rendelkezésre.

Nyersdarab

Az a darab, amellyel egy munkadarab megmunkálása elkezdődik.

OEM

Azon gépgyártók részére, akik a saját kezelői felületüket állítják elő vagy technológia-specifikus funkciókat akarnak a vezérlésbe bevinni, az egyéni megoldásokra (OEM-applikációk) szabad lehetőség van biztosítva.

Orsóemelkedés-hibakompenzáció

Egy az előtolásban résztvevő golyósorsó mechanikus pontatlanságainak a vezérlés általi kiegyenlítése a tárolt eltérési mérésértékek alapján.

Osztótengely

A osztótengelyek egy munkadarab- vagy szerszámelforgatást hoznak egy osztóraszternek megfelelő szöghelyzetbe. Egy raszter eléréséné az osztótengely "pozícióban van“.

Override

Kézi ill. programozható beavatkozási lehetőség, amely a kezelő részére lehetővé teszi a programozott előtolások vagy fordulatszámok befolyásolását azért, hogy egy meghatározott munkadarabhoz vagy anyagra illeszteni tudja.

Pályaelőtolás

A pályaelőtolás a -> pályatengelyekre hat. Ez a résztvevő → geometria-tengelyek előtolásainak geometriai összege.

Pályasebesség

A maximálisan programozható pályasebesség a beadás-felbontástól függ. Például 0,1 mm-es felbontásnál a maximálisan programozható pályasebesség 1000 m/perc.

Pályatengely

Pályatengelyek a → csatorna összes azon megmunkálási tengelyei, amelyeket az → interpolátor úgy vezet, hogy egyidőben indulnak, gyorsítanak, megállnak és érik el a végpontjukat .

Pályavezérlő-üzem

A pályavezérlő-üzem célja az, hogy megakadályozza a → pályatengelyek munkadarabprogram-mondatvégeken történő nagyobb lefékezését, s annak biztosítását, hogy a következő mondatra történő váltás lehetőleg azonos pályasebességgel történjék.

Periféria-modul

Periféria-modulok a CPU és a folyamat közötti kapcsolatot hozzák létre.

Periféria-modulok:

- → digitális be-/kiadási egységek
- → analóg be-/kiadási egységek
- → szimulátor-egységek

PLC

Programmable **L**ogic **C**ontrol: → Tároló-programozható vezérlés Az→NC komponense: Illesztő-vezérlés a szerszámgép vezérlő logikájának feldolgozásához

PLC programozás

A PLC programozása a **STEP 7** szoftverrel történik A STEP 7 programozó szoftver a **WINDOWS** operációs rendszerre épül és a STEP 5 programozás funkcióinak innovatív továbbfejlesztése.

PLC programtároló

SINUMERIK 840D sl: A PLC felhasználói tárolóban a PLC felhasználói programok és a felhasználói adatok a PLC-alapprogrammal együtt vannak tárolva.

Polár-koordináták

Koordinátarendszer, amely egy pont helyzetét egy síkban annak a nullaponttól lévő távolságával és azzal a szöggel határozza meg, amelyet a sugárvektor egy meghatározott tengellyel képez.

Polinom interpoláció

A polinom interpolációval különféle görbéket lehet létrehozni, mint **egyenes-, parabola-, hatványfüggvények** (SINUMERIK 840D sl).

Pontos állj

Programozott pontos-állj utasításnál az egy mondatban megadott pozícióra pontosan és szükség esetén nagyon lassan történik rámenetel. A megközelítési idő lecsökkentésére a gyorsmenetre és az előtolásra → pontos-állj határok definiálhatók.

Pontos-állj határ

Ha az összes pályatengely elérte a pontos-állj határát, akkor a vezérlés úgy viselkedik, mintha egy célpontot pontosan elért volna. Egy mondat-továbbkapcsolás történik a→ munkadarabprogramban.

Pozicionáló tengely

Tengely, amely egy segédmozgást hajt végre egy szerszámgépen. (pl. szerszám-tár, paletta-szállítás). Pozicionáló tengelyek olyan tengelyek, amelyek a → pályatengelyekkel nem interpolálnak.

Programmodul

Programmodulok tartalmazzák a → munkadarabprogramok fő- és alprogramjait.

Programozási kulcs

Jelek és jelsorozatok, amelyeknek a programozási nyelven a → munkadarabprogramra egy meghatározott jelentőséggel bírnak.

Programozható frame-k

Programozható → frame-kkel dinamikusan a munkadarabprogram-feldolgozása alatt új koordinátarendszer-kiindulási pontok definiálhatók. Egy az új frame alapján történő abszolút meghatározást és az egy fennálló kiindulási pontra vonatkozó additív meghatározást különböztetünk meg.

Programozható munkatér-határolás

A szerszám mozgásterének behatárolása egy programozható határolásokkal definiált térben.

Programszint

A csatornában elindított munkadarabprogram → főprogramként fut le a programszint 0-ban (főprogram szint). A főprogramban felhívott minden munkadarabprogram → alprogramként egy 1 ... n saját programszinten fut.

Pufferelem

A pufferelem biztosítja, hogy a → felhasználói program a → CPU-ban hálózat kimaradás biztosan van tárolva és a meghatározott adattartományok tárolók, időzítések és számlálók megmaradnak.

Referenciapont

A szerszámgép azon pontja, amelyre a→ géptengelyek mérőrendszere vonatkozik.

Rendszer-tároló

A rendszer-tároló egy tároló a CPU-ban, amelyikben a következő adatok vannak elhelyezve:

- adatok, a melykre az operációs rendszernek van szüksége
- Időzítés, számláló, jelölő operandusok

Rendszerváltozók

A programozó cselekedete nélkül egy → munkadarabprogram létező változója. Ezt egy adattípus és a$ jellel bevezetett változónév határozza meg. Lásd → Felhasználó által definiált változók.

R-paraméter

Számítási paraméter, amelyet a→ munkadarabprogram programozója tetszőleges célokra a programban be tud állítani vagy le tud kérdezni.

Sebesség-vezetés

Annak érdekében, hogy a mondatonkénti nagyon kis értékek elmozdulására egy elfogadható elmozdulási sebességet érhessünk el, be lehet állítani a több mondaton keresztüli előrelátó kiértékelést (→ Look Ahead).

Segédfunkciók

Segédfunkciókkal a → munkadarabprogramokban → paramétereket lehet a → PLC-nek átadni, amelyek ott a gépgyártó által meghatározott reakciókat váltják ki.

Skálázás

Egy → frame komponense, amely a tengelyspecifikus mérték-változásokat okoz.

Softkey

Billentyű, amely feliratozása a képernyőn egy olyan mező, ami dinamikusan illesztődik az aktuális kezelői helyzetre. A szabadon felhasználható funkciós billentyűk (softkey-k) szoftver által definiált funkciókhoz lesznek hozzárendelve

Spline interpoláció

A Spline-interpolációval a vezérlés egy parancs kontúr csak néhány, előre megadott támpontjaiból egy sima görbelefolyást tud előállítani.

Szabvány ciklusok

A gyakran ismétlődő megmunkálási feladatokra szabványciklusok állnak rendelkezésre:

- Fúrás/marás technológiára
- Esztergálás technológiára

A "Program" kezelői tartományban a "Ciklustámogatás" menü alatt a rendelkezésre álló ciklusok listája látható. A kívánt megmunkálási ciklus kiválasztása után az érték-hozzárendeléshez szükséges paraméterek szöveggel kijelzésre kerülnek és értékekkel láthatók el.

Szerkesztő

A szerkesztő lehetővé teszi programok /szövegek /programmondatok előállítását, megváltoztatását, kiegészítését, összefűzését és betoldását.

Szerszám

A munkadarab megmunkálásához szükséges szerszám (pl. esztergakés, maró, fúró, köszörűkő, lézersugár ...).

Szerszámkorrekció

A szerszám méreteinek figyelembe vétele a pálya kiszámításánál.

Szerszámsugár-korrekció

Egy kívánt → munkadarab-kontúr közvetlen programozásához a vezérlésnek az alkalmazott szerszám sugarának figyelembevételével egy ekvidisztáns pályán kell a programozott kontúrhoz elmennie (`G41`/`G42`).

Szinkron-akciók

1. Segédfunkció-kiadás
   A munkadarab megmunkálása alatt a CNC-programból technológiai funkciók (→ segédfunkciók) adhatók ki a PLC-nek. Ezekkel a segédfunkciókkal pl. a szerszámgép kiegészítő berendezései vezérelhetők, mint pl. a csúcstámasz, markoló, szorítótokmány, stb.
2. Gyors segédfunkció-kiadás
   Időkritikus kapcsolási funkciókra a → segédfunkciók nyugtázási időit minimálni lehet és ezzel ki lehet kerülni a megmunkálási folyamatban szükségtelen megállás-pontokat.

Szinkronizáció

Utasítások meghatározott megmunkálási helyeknek a → munkadarab-programokban a különböző→ csatorna folyamatok koordinálására.

Szinkrontengelyek

A szinkrontengelyek útjára ugyanazt az idő szükséges, mint a geometria-tengelyek a pályaútjára.

Szoftver végálláskapcsoló

A szoftver végálláskapcsolók egy tengely elmozdulási tartományát határolják be, és megakadályozzák a szánnak a hardver-végálláskapcsolóra történő rámenetelét. Tengelyenként 2 értékpár adható előre meg, amelyek elválasztva a → PLC-n keresztül aktiválhatók.

Szöveg szerkesztő

Lásd → Szerkesztő

Tájolt orsó-állj

A munkadarab-orsó megállása az előre megadott szöghelyzetben, pl. azért, hogy egy meghatározott helyen egy kiegészítő megmunkálást hajthassunk végre.

Tároló-programozható vezérlés

A tároló programozható vezérlések (SPS) elektronikus vezérlések, amelyek funkciói programként vannak a vezérlésben tárolva. A készülék felépítése és huzalozása tehát nem függ a vezérlés funkciójától. A tároló programozható vezérlésnek a számítógépnek megfelelő struktúrája van és a következőkből tevődik össze: CPU (központi egység) tárolóval, be- / kimeneti modulok és belső buszrendszer A perifériák és a programozási nyelv a vezérléstechnika követelményei szerint vannak kialakítva.

Teljes törlés

A teljes törlésnél a → CPU következő tárolói törlődnek:

- → munkatároló
- → töltőtároló írás /olvasás tartománya
- → rendszertároló
- → mentéstároló

Tengelycím

Lásd → Tengelynév

Tengelyek

A CNC tengelyek a funkció-terjedelmüknek megfelelően vannak osztályozva:

- Tengelyek: interpoláló pályatengelyek
- Segédtengelyek: nem interpoláló fogásvételi és pozícionáló tengelyek tengely-specifikus előtolással. A segédtengelyek a tulajdonképpeni megmunkálásban nem vesznek részt, pl. szerszám-mozgató, szerszámtár.

Tengelynév

Az egyértelmű azonosításhoz a vezérlés összes csatorna- és → géptengelyét a vezérlésben egyértelmű névvel kell megjelölni. A → geometria-tengelyek megnevezése X, Y, Z. A geometria-tengelyek körül forgó → kör-tengelyek neve A, B, C.

TOA–egység

Minden → TOA-tartomány több TOA-egységet tartalmazhat. A lehetséges TOA-egységek számát az aktív → csatornák maximális száma határolja. Egy TOA-egység pontosan egy szerszámdat-modult és egy táradat-modult tartalmaz. Ezen kívül tartalmazhat még egy szerszámtartó-adatmodult (opció).

TOA–tartomány

A TOA–tartomány átfogja az összes szerszám- és táradatot. Általában ez a tartomány az adatok hatásköre szempontjából egybeesik a → csatorna tartománnyal. Azonban gépadatokkal meg lehet adni, hogy több csatorna osztozzon egy → TOA-egységen, így ezeknek a csatornáknak közös szerszámadatok állnak rendelkezésre.

Töltőtároló

A töltőtároló a → PLC CPU 314-nél azonos a → munkatárolóval.

Transzformáció

Egy tengely additív vagy abszolút nullaponteltolása.

Tükrözés

Tükrözésnél egy kontúr koordináta-értékeinek előjelei egy tengelyhez viszonyítva megcserélődnek. Egyidejűleg több tengelyre vonatkoztatottan lehet tükrözni.

Üzemmód

Egy SINUMERIK vezérlés üzemének kezelési koncepciója. A → Jog, → MDA, → Automatika üzemmódok vannak meghatározva.

Üzemmód-csoport

A technológiailag összetartozó tengelyeket és orsókat össze kehet foglalni egy üzemmód-csoportba (BAG). Egy BAD tengelyeit/orsóit egy vagy több → csatorna vezérelheti. A BAG csatornáihoz mind ugyanaz az → üzemmód van hozzárendelve.

V.24

Soros interfész adat be-/kivitelre. Ezen az interfészen keresztül lehet a megmunkálóprogramokat, továbbá a gyártói és az alkalmazói adatokat betölteni ill. menteni.

Vágóélsugár-korrekció

A kontúr programozásánál egy hegyes szerszámból indulnak ki. Mivel ez a gyakorlatban nem valósítható meg, az alkalmazott szerszám görbületi sugarát meg kell adni a vezérlésnek és az azt figyelembe veszi. Ennél a görbületi középpont a görbületi sugárral eltolva a kontúrtól azonos távolságra lesz vezetve.

Változó definíció

Egy változó definíció egy adattípus és egy változónév meghatározást foglal magába. A változó-névvel a változó értékéhez hozzá lehet férni.

Védőtér

Háromdimenziós tér a munkatéren belül, ahova nem érhet el a szerszám csúcsa.

Vészjelzések

Összes → A jelentések és vészjelzések a kezelőhelyen szövegesen dátummal és idővel és törlési kritérium megfelelő szimbólumával vannak kijelezve. A vészjelzések és a jelentések kijelzése külön történik.

1. Vészjelzések és jelentések a munkadarabprogramban.
   A vészjelzéseket és a jelentéseket a munkadarabprogramból közvetlenül ki lehet jeleztetni.
2. Vészjelzések és jelentések a PLC-től
   A gép vészjelzéseit és jelentéseit a PLC programból közvetlenül ki lehet jeleztetni. Ehhez nem szükséges külön funkciómodul csomag.

Vezető tengely

A vezető tengely a → Ganrty tengely, amelyik a kezelő és a programozó szempontjából létezik és ennek megfelelően normál NC tengelyként befolyásolható.

WinSCP

WinSCP egy szabadon felhasználható nyitott forráskódú program Windows-hoz a fájlok átviteléhez.

Index

-

-tároló
    előrefutás-, 519
    munka-, 222
    program-, 211

$

$AA_ATOL, 542
$AA_COUP_ACT
    tengely vezetőérték csatolásnál, 572
    vontatásnál:, 548
$AA_ESR_ENABLE, 664
$AA_LEAD_SP, 572
$AA_LEAD_SV, 572
$AC_ACT_PROG_NET_TIME, 653
$AC_ACTUAL_PARTS, 656
$AC_AXCTSWA, 638
$AC_AXCTSWE, 638
$AC_CTOL, 542
$AC_CUT_INV, 467
$AC_CUTMOD, 467
$AC_CUTMOD_ANG, 467
$AC_CUTMODK, 467
$AC_CUTTING_TIME, 652
$AC_CYCLE_TIME, 652
$AC_DELAYFST, 524
$AC_ESR_TRIGGER, 664
$AC_OLD_PROG_NET_TIME, 653
$AC_OLD_PROG_NET_TIME_COUNT, 653
$AC_OPERATING_TIME, 652
$AC_OTOL, 542
$AC_PROG_NET_TIME_TRIGGER, 653
$AC_REPOS_PATH_MODE, 533
$AC_REQUIRED_PARTS, 655
$AC_SMAXVELO, 538
$AC_SMAXVELO_INFO, 538
$AC_SPECIAL_PARTS, 656
$AC_TOTAL_PARTS, 655
$AC_TRAFO_CORR_ELEM_P, 402
$AC_TRAFO_CORR_ELEM_T, 402
$AC_TRAFO_ORIAX_LOC, 402
$AN_AXCTAS, 638
$AN_AXCTSWA, 638
$AN_ESR_TRIGGER, 664
$AN_LANGUAGE_ON_HMI, 914
$AN_POWERON_TIME, 652
$AN_SETUP_TIME, 652
$NT_CLOSE_CHAIN_T, 402
$NT_CNTRL, 402
$NT_CORR_ELEM_P, 401
$NT_CORR_ELEM_T, 401
$NT_NAME, 393
$NT_ROT_AX_NAME, 464
$NT_TRAFO_INDEX, 393
$P_ACTBFRAME, 311
$P_AD, 466
$P_BFRAME, 311
$P_CHBFRAME, 311
$P_CHBFRMASK, 312
$P_CTOL, 543
$P_CUT_INV, 467
$P_CUTMOD, 467
$P_CUTMOD_ANG, 467
$P_CUTMOD_ERR, 468
$P_CUTMODK, 467
$P_DELAYFST, 524
$P_IFRAME, 312
$P_IS_EES_PATH, 221
$P_NCBFRAME, 311
$P_NCBFRMASK, 312
$P_ORI_DIFF, 460
$P_ORI_POS, 460
$P_ORI_SOL, 461
$P_ORI_STAT, 463
$P_OTOL, 543
$P_PATH, 220
$P_PFRAME, 313
$P_PROG, 220
$P_PROGPATH, 221
$P_SIM, 279
$P_STACK, 220
$P_SUBPAR, 161
$P_TOOLENV, 476
$P_TOOLENVN, 476
$PA_ATOL, 543
$SA_LEAD_TYPE, 572
$SC_CONTPREC, 517
$SC_MINFEED, 517
$SC_PA_ACTIV_IMMED, 232
$SN_PA_ACTIV_IMMED, 232
$TC_CARR1...14, 445
$TC_CARR18...23, 445
$TC_CARR18[m], 449
$TC_DP1 ... 25, 403

$TC_ECPxy, 407
$TC_SCPxy, 407

*

* (számítási művelet), 74

/

/ (számítási művelet), 74

+

+ (számítási művelet), 74

<

< (összehasonlító operátor), 76
<< (láncolás operátor), 86
<= (összehasonlító operátor), 76
<> (összehasonlító operátor), 76

=

== (összehasonlító operátor), 76

>

> (összehasonlító operátor), 76
>= (összehasonlító operátor), 76

0

0-karakter, 84

A

ABS, 74
ACCLIMA, 512
ACOS, 74
ACTBLOCNO, 173
ACTFRAME, 287
Adat-osztály, 50
ADISPOSA, 281
ág-megadás, 216
ALF
    gyors leemelésre a kontúrról, 132
alprogram
    alkalmazás, 156
    -felhívás paraméter-átadás nélkül,
    -felhívás paraméter-átadással,
    -hívás közvetett, 195
    -hívás modális, 193
    -ismétlés,
    -név,
    programozható keresőág,
    -visszaugrás paraméterezhető (RET ...),
    -visszaugrás paraméterezhető (RETB...), 184
alprogram-hívás program ág-megadással és paraméterekkel, 198
AND, 76
APR, 45
APRB, 45
APRP, 45
APW, 45
APWB, 45
APWP, 45
Array, 52
AS, 208
ASIN, 74
ASPLINE, 244
A-Spline, 250
aszinkron ingázás, 603
    Fogásvétel a fordulás-tartományban, 612
    Fogásvételek megadása, 611
    Inga- és fogásvétel-tengely hozzárendelés, 611
    kiértékelés, IPO-ütem, 613
    Következő rész-fogásvétel, 613
    rámeneteli mozgás, 612
    Szinkronakciók, 611
ATAN2, 74
Átkapcsolható geometria-tengelyek, 632
ATOL, 540
Automatikus megszakítás-mutató, 526
Automatikus út-felosztás, 620
AV, 580
AX, 629
AXCTSWE, 637
AXCTSWEC, 637
AXCTSWED, 637
AXIS, 29
AXNAME, 85
AXSTRING, 629
AXTOCHAN, 142
AXTOSPI, 629

B

B_AND, 76

B_NOT, 76
B_OR, 76
B_XOR, 76
BAUTO, 244
Beállítási érték, 407
bejáratás kiegészítés vége - GROUP_ADDEND
    kívül programozni, 818
bemerülési mélység, 423
Beszúrás - CYCLE930
    kívül programozni, 784
BFRAME, 287
Billentés - CYCLE800
    kívül programozni, 762
BLOCK, 196
BLSYNC, 127
BNAT, 244
BOOL, 29
BOUND, 80
BRISK, 510
BRISKA, 510
BSPLINE, 244
B-Spline, 251
BTAN, 244

C

CAC, 243
CACN, 243
CACP, 243
CALL, 195
CALLPATH, 199
CASE, 105
case-insensitive, 214
CDC, 243
CFINE, 297
CHAN, 29
CHANDATA, 222
CHAR, 29
CHKDNO, 442
CIC, 243
ciklus vészjelzések, 662
címzés, 215
CLEARM, 120
CLRINT, 129
COARSE, 580
COARSEA, 281
COLLPAIR, 388
COMPCAD, 257
COMPCURV, 257
COMPLETE, 222
COMPOF, 257
COMPON, 257
COMPSURF, 257
CONTDCON, 684
CONTPRON, 678
CORRTRAFO, 395
COS, 74
COUPDEF, 580
COUPDEL, 580
COUPOF, 580
COUPOFS, 580
COUPON, 580
COUPONC, 580
COUPRES, 580
CP, 369
CPBC, 592
CPDEF, 591
CPDEL, 591
CPFMOF, 594
CPFMON, 594
CPFMSON, 593
CPFPOS + CPOF, 594
CPFPOS + CPON, 592
CPFRS, 592
CPLA, 591
CPLCTID, 592
CPLDEF, 591
CPLDEL, 591
CPLDEN, 592
CPLINSC, 596
CPLINTR, 596
CPLNUM, 592
CPLOF, 592
CPLON, 591
CPLOUTSC, 596
CPLOUTTR, 596
CPLPOS, 592
CPLSETVAL, 592
CPMALARM, 597
CPMBRAKE, 597
CPMPRT, 596
CPMRESET, 595
CPMSTART, 596
CPMVDI, 597
CPOF, 591
CPON, 591
CPRECOF, 517
CPRECON, 517
CPROT, 229
CPROTDEF, 225
CPSETTYPE, 597
CPSYNCOP, 596
CPSYNCOP2, 596
CPSYNCOV, 596

CPSYNFIP, 596
CPSYNFIP2, 596
CPSYNFIV, 596
C-Spline, 252
CTAB, 561
CTABDEF, 550
CTABDEL, 557
CTABEND, 550
CTABEXISTS, 557
CTABFNO, 566
CTABFPOL, 566
CTABFSEG, 566
CTABID, 559
CTABINV, 561
CTABISLOCK, 559
CTABLOCK, 558
CTABMEMTYP, 559
CTABMPOL, 566
CTABMSEG, 566
CTABNO, 566
CTABNOMEM, 566
CTABPERIOD, 559
CTABPOL, 566
CTABPOLID, 566
CTABSEG, 566
CTABSEGID, 566
CTABSEV, 561
CTABSSV, 561
CTABTEP, 561
CTABTEV, 561
CTABTMAX, 561
CTABTMIN, 561
CTABTSP, 561
CTABTSV, 561
CTABUNLOCK, 558
CTOL, 540
CTRANS, 297
CUT3DC, 421
CUT3DCC, 431
CUT3DCCD, 431
CUT3DCD, 421
CUT3DF, 425
CUT3DFD, 425
CUT3DFF, 425
CUT3DFS, 425
CUTMOD, 464
CUTMODK, 464
CYCLE4071
    kívül programozni, 798
CYCLE4072
    kívül programozni, 799
CYCLE4073
    kívül programozni, 803
CYCLE4074
    kívül programozni, 804
CYCLE4075
    kívül programozni, 807
CYCLE4077
    kívül programozni, 810
CYCLE4078
    kívül programozni, 813
CYCLE4079
    kívül programozni, 815
CYCLE435 - Lehúzó koordináták beállítása
    kívül programozni, 760
CYCLE495 - Profilozás
    kívül programozni, 760
CYCLE60 - Gravírozás
    kívül programozni, 712
CYCLE61 - Síkmarás
    kívül programozni, 715
CYCLE62 - kontúr felhívása
    kívül programozni, 717
CYCLE63 - Kontúrzsebet marni
    kívül programozni, 718
CYCLE64 - Kontúrzsebet előfúrni
    kívül programozni, 720
CYCLE70 - Menetmarás
    kívül programozni, 721
CYCLE72 - pályamarás
    kívül programozni, 723
CYCLE76 - négyszögcsap
    kívül programozni, 726
CYCLE77 - körcsap
    kívül programozni, 729
CYCLE78 - Furatmenet marás
    kívül programozni, 731
CYCLE79 - Sokszög
    kívül programozni, 733
CYCLE800 - Billentés
    kívül programozni, 762
CYCLE801 - rács/keret pozícióminta
    kívül programozni, 765
CYCLE802 - Tetszőleges pozíciók
    kívül programozni, 767
CYCLE81 - központozás
    kívül programozni, 735
CYCLE82 - Fúrás
    kívül programozni, 736
CYCLE83 - mélyfúrás
    kívül programozni, 739
CYCLE830 - Mély-lyuk fúrás 2
    kívül programozni, 769

CYCLE832 - High Speed Settings
kívül programozni, 775
CYCLE84 - menetfúrás kiegyenlítő tokmány nélkül
kívül programozni, 742
CYCLE840 - menetfúrás kiegyenlítő tokmánnyal
kívül programozni, 779
CYCLE85 - Dörzsölés
kívül programozni, 745
CYCLE86 - kiesztergálás
kívül programozni, 746
CYCLE899 - Nyitott vájat marása
kívül programozni, 782
CYCLE92 - leszúrás
kívül programozni, 747
CYCLE930 - beszúrás
kívül programozni, 784
CYCLE940 - szabadra szúrás
kívül programozni, 787
CYCLE95 - kontúr leforgácsolás
kívül programozni, 749
CYCLE951 - leforgácsolás
kívül programozni, 789
CYCLE952 - Kontúr-szúrás
kívül programozni, 792
CYCLE98 - Menet láncok
kívül programozni, 751
CYCLE99 - menetesztergálás
kívül programozni, 755
csatolás
generátoros, 591
Csatolás állapota
tengely vezetőérték csatolásnál, 572
vontatásnál:, 548
csatolási mód, 583
csatolási tényező:, 545
CSPLINE, 244

D

DCI, 50
DCM, 50
DCU, 50
DEF, 29
DEFAULT, 105
DEFINE ... AS, 208
DELAYFSTOF, 522
DELAYFSTON, 522
DELDL, 408
DELETE, 149
DELOBJ, 383
DELTOOLENV, 474
Derékszögű PTP-mozgás, 369
DIN alprogram-nevek, 219
DISABLE, 129
DISPLOF, 173
DISPLON, 173
DISPR, 527
DIV, 74
DL, 406
DO, 601
Dörzsölés - CYCLE85
kívül programozni, 745
DRIVE, 510
DRIVEA, 510
D-szám
átnevezni, 443
szabad kiadás, 442
vizsgálni, 442
DV, 580
DYNFINISH, 514
DYNNORM, 514
DYNPOS, 514
DYNROUGH, 514
DYNSEMIFIN, 514

E

Easy XML, 647
EAUTO, 244
EES, 213
EES címzés, 215
EG
Elektronikus hajtómű, 574
EGDEF, 574
EGDEL, 579
EGOFC, 578
EGOFS, 578
EGON, 575
EGONSYN, 575
EGONSYNE, 575
egyes-mondat
-elnyomás, 168
Elektronikus hajtómű, 574
előrefutás
-tároló, 519
ELSE, 114
ENABLE, 129
ENAT, 244
ENDFOR, 116
ENDIF, 114
ENDLABEL, 107
ENDLOOP, 116
ENDWHILE, 118
ESR, 663

ESRR, 669
ESRS, 668
ETAN, 244
Euler-szög, 331
EVERY, 601
EXECTAB, 690
EXECUTE, 693
EXECSTRING, 72
EXP, 74
EXTCALL
    SINUMERIK 828D-hez, 203
    SINUMERIK 840D sl-hez, 200
EXTCLOSE, 657
EXTERN, 190
EXTOPEN, 657

F

Fájl
    -Információk, 153
Fájlnév, 218
FALSE, 29
FCTDEF, 417
FCUB, 505
felhasználói XML, 647
Felkerekítés, 155
FENDNORM, 280
ferde beszúrás köszörülés, 365
ferdeszög-transzformáció (TRAANG)
    programozható szöggel, 364
FFWOF, 516
FFWON, 516
FIFOCTRL, 519
FILEDATE, 153
FILEINFO, 153
FILESIZE, 153
FILESTAT, 153
FILETIME, 153
FINE, 580
FINEA, 281
FLIN, 505
FNORM, 505
FOR, 116
fordulatszám szinkronitás, 580
forgástengely
    elforgatás szög, 445
    irányvektorok, 445
    távolság vektorok, 445
forgásvektor interpolációja, 347
forgatás
    tájolás vektor, 347
FPO, 505
FRAME, 29
    felhívni,
    -láncolás,
Frame komponensek
    FI, 293
    MI, 293
    RT, 293
    SC, 293
    TR, 293
frame változók
    Előre definiált frame változó, 287
    Értékek hozzárendelése, 291
    Felhívás - koordináta-transzformációk, 285
frame-ek
    csatorna-specifikus, 310
    Frame-láncok,
    hozzárendelés,
    NCU-globális, 309
    rendszer-, 310
FROM, 601
FTOCOF, 420
FTOCON, 420
Fúrás - CYCLE82
    kívül programozni, 736
Furatmenet marás - CYCLE78
    kívül programozni, 731
futási idő
    -viselkedés vezérlő-struktúráknál, 113

G

G290, 675
G291, 675
G5, 365
G62, 280
G621, 280
G7, 365
G810 ... G819, 279
G820 ... G829, 279
G-csoport
    technológia, 514
GEOAX, 632
geometria-tengely
    átkapcsolni, 632
GET, 137
GETACTTD, 444
GETD, 137
GETDNO, 443
GETTCOR, 476
GETTENV, 475
GETVARAP, 61
GETVARDFT, 63

GETVARDIM, 62
GETVARLIM, 61
GETVARPHU, 60
GETVARTYP, 64
G-kód
    közvetetten programozni, 68
Globális munkadarabprogram-tároló (GDIR), 213
GOTO, 102
GOTOB, 102
GOTOC, 102
GOTOF, 102
GOTOS, 101
GP, 69
Gravírozás - CYCLE60
    kívül programozni, 712
GROUP_ADDEND - bejáratás kiegészítés vége
    kívül programozni, 818
GROUP_BEGIN - programblokk kezdete
    kívül programozni, 817
GROUP_END - programblokk vége
    kívül programozni, 818
GUD, 30
Gyors leemelés a kontúrról, 130
gyorsítási módus, 510

H

helyzet szinkronitás szögeltolással, 580
helyzet-szinkronitás, 580
Hengerpalást-transzformáció, 319
High Speed Settings - CYCLE832
    kívül programozni, 775
HOLES1 - vonal pozícióminta
    kívül programozni, 698
HOLES2 - kör pozícióminta
    kívül programozni, 698
Homlokmarás, 335
Hosszlyuk - LONGHOLE
    kívül programozni, 710
Hosszvájat - SLOT1
    kívül programozni, 705

I

ID, 601
IDS, 601
IF, 114
IFRAME, 287
INDEX, 89
Ingamozgás
    Fogásvétel a fordulóponton, 612
    fordulás-tartomány, 610
    fordulópont, 610
Ingázás
    aszinkron, 603
    aszinkron ingázás, 603, 608
    rész-fogásvétel, 610
    szinkronakcióval vezérelni, 608
INICF, 29
inicializálás
    mezőknek, 52
Inicializáló program, 222
INIPO, 29
INIRE, 29
INIT, 120
INITIAL, 222
INITIAL_INI, 222
INT, 29
Interrupt rutinok
    Gyors leemelés a kontúrról, 130
interrupt-rutinok
    ki-/ bekapcsolni, 129
    programozható lemeneti irány, 132
    törölni, 129
    újra hozzárendelni, 128
    Visszahúzás mozgás, 132
INTERSEC, 688
IPOBRKA, 281
IPOENDA, 281
IPOSTOP, 580
IPTRLOCK, 525
IPTRUNLOCK, 525
irányvektor, 332
ISAXIS, 629
ISFILE, 152
ISNUMBER, 85
ISOCALL, 197
ISVAR, 58

J

JERKLIM, 536
JERKLIMA, 512

K

Keresési ág
    alprogram-hívásnál, 219
    programozható keresőág, 199
Keresésre alkalmatlan tartományok felismerése és keresése, 526

Kiadás
    külső készülékre/fájlba, 657
Kiesztergálás - CYCLE86
    kívül programozni, 746
Kinematika
    felbontott, 449
kinematika-típus, 449
kívül programozni, 817
kontúr
    -feldolgozás, 678
    ismét rámenni, 527
    -kódolás, 684
    -táblázat,
Kontúr felhívása - CYCLE62
    kívül programozni, 717
Kontúr leforgácsolás - CYCLE95
    kívül programozni, 749
Kontúrelem
    megtenni, 690
kontúr-feldolgozás
    hiba visszajelzés, 693
Kontúr-pontosság
    programozható, 517
Kontúr-szúrás - CYCLE952
    kívül programozni, 792
Kontúrzsebet előfúrni - CYCLE64
    kívül programozni, 720
Kontúrzsebet marni - CYCLE63
    kívül programozni, 718
kopásérték, 407
Korrekció-tároló, 403
Könyvtár-ág, 217
kör pozícióminta - HOLES2
    kívül programozni, 698
kör-adatok
    kiszámítani, 691
Körcsap - CYCLE77
    kívül programozni, 729
Körvájat - SLOT2
    kívül programozni, 708
Körzseb - POCKET4
    kívül programozni, 703
követő tengely
    tengely vezetőérték csatolásnál, 568
Központozás - CYCLE81
    kívül programozni, 735
Közvetett programozás
    címekből, 66
    G-kódok, 68
    munkadarabprogram sorok, 72
    pozíció attribútumok, 69
Külső nullaponteltolás, 299

L

L..., 188
Label, 107
láncolása
    stringek, 86
LEAD, 329
LEADOF, 568
LEADON, 568
leállítás
    hajtás alapú, 668
    NC vezetésű, 668
leforgácsolás
    támogatott funkciók, 677
Leforgácsolás - CYCLE951
    kívül programozni, 789
Lehúzó koordináták beállítása - CYCLE435
    kívül programozni, 760
LENTOAX, 496
Leszúrás - CYCLE92
    kívül programozni, 747
LIFTFAST, 130
Link
    -változók, 27
LLI, 41
LN, 74
Logikai operátorok, 76
LONGHOLE - hosszlyuk
    kívül programozni, 710
LOOP, 116
LUD, 30
Lyukasztás, 615

M

M17, 176
M30, 176
Makró, 208
Maradékidő
    egy munkadarabra, 654
maradékút törlés, 275
maró megmunkálási pont, 424
marócsúcs, 424
maró-segédpont, 424
MASLDEF, 598
MASLDEL, 598
MASLOF, 598
MASLOFS, 598
MASLON, 598
MATCH, 89

MAXVAL, 80
MCALL, 193
MD10010, 120
MD10280, 120
MD15800, 26
MD18104, 473
MD18116, 474
MD18156, 26
MD20360, 481
MD24558, 482
MD24658, 482
MEAC, 269
MEAFRAME, 305
MEAS, 266
MEASA, 269
MEAW, 266
MEAWA, 269
megállási mondat, 526
Meghajtó-név, 216
megmunkálási idő, 652
Mély-lyuk fúrás (CYCLE83)
    kívül programozni, 739
Mély-lyuk fúrás 2 - CYCLE830
    kívül programozni, 769
Menet láncok - CYCLE98
    kívül programozni, 751
Menetesztergálás - CYCLE99
    kívül programozni, 755
Menetfúrás kiegyenlítő tokmány nélkül - CYCLE84
    kívül programozni, 742
Menetfúrás kiegyenlítő tokmánnyal - CYCLE840
    kívül programozni, 779
Menetmarás - CYCLE70
    kívül programozni, 721
Mérési feladat állapota, 277
mező
    -definíció, 52
    -elem, 52
Mező-index, 54
MINDEX, 89
MINVAL, 80
MMC, 647
MOD, 74
MODAXVAL, 629
Mondat-kijelzés
    elnyomni, 173
mozgás vége kritérium
    programozható, 281
MPF, 212
munkadarab
    -főkönyvtár, 212
    -könyvtárak, 212
    -számláló, 655
munkadarab-lánc, 400
Munkatároló, 222

N

NAMETOINT, 386
NCK, 29
NCK címzés, 215
Négyszögcsap - CYCLE76
    kívül programozni, 726
Négyszögzseb - POCKET3
    kívül programozni, 700
nevező-polinom, 262
NEWCONF, 144
NOC, 580
NOT, 76
NPROT, 229
NPROTDEF, 225
Nullaponteltolás
    külső, 299
NUMBER, 85
NUT, 340
nyelv módus, 675
Nyersdarab definíció, 671
Nyitott vájat marása - CYCLE899
    kívül programozni, 782

O

OEM funkciók, 279
OEM-címek, 279
OEMIPO1/2, 279
OMA1 ... OMA5, 279
online szerszámhossz-korrekció, 453
OR, 76
ORIAXES, 338
ORIC, 436
ORICONCCW, 340
ORICONCW, 340
ORICONIO, 340
ORICONTO, 340
ORICURVE, 343
ORID, 436
ORIEULER, 338
ORIMKS, 336
ORIPATH, 351
ORIPATHS, 351
ORIPLANE, 340
ORIRESET(A, B, C), 327

ORIROTA, 347
ORIROTC
szerszámtájolás forgatásnál, 347
szerszámtájolás interpolációnál, 352
ORIROTR, 347
ORIROTT, 347
ORIRPY, 338
ORIRPY2, 338
ORIS, 436
ORISOF, 359
ORISOLH, 456
ORISON, 359
ORIVECT, 338
ORIVIRT1, 338
ORIVIRT2, 338
ORIWKS, 336
Orsó
-csere, 137
OS, 603
OSB, 603
OSC, 436
OSCILL, 608
OSCTRL, 603
OSD, 436
OSE, 603
OSNSC, 603
OSOF, 436
OSP1, 603
OSP2, 603
OSS, 436
OSSE, 436
OST, 436
OST1, 603
OST2, 603
OTOL, 540
Összehasonlító operátorok, 76

P

P..., 192
P_ACTFRAME, 313
Pályamarás - CYCLE72
kívül programozni, 723
Pályára vonatkoztatott szerszámtájolás, 350
Paraméter
aktuális-, 160
-átadás alprogram-hívásnál, 160, 190
formális-, 159
Szerszám-, 403
parancsérték csatolás, 583
PCALL, 198
PDELAYOF, 615
PDELAYON, 615
Peremfeltételek transzformációknál, 380
PFRAME, 287
PHI
tájolási polinom, 345
tájolásnál a kúppalást felületen, 340
PHU, 43
PL
Polinom interpolációnál, 258
Spline interpolációnál, 244
PO, 258
PO[PHI]
szerszámtájolás forgatásnál, 351
tájolási polinom, 345
tájolásnál a kúppalást felületen, 340
PO[PSI]
szerszámtájolás forgatásnál, 351
tájolási polinom, 345
tájolásnál a kúppalást felületen, 340
PO[THT]
szerszámtájolás forgatásnál, 351
tájolási polinom, 345
PO[XH]
két érintési pontos tájolás megadásnál, 343
tájolási polinom, 346
PO[YH]
két érintési pontos tájolás megadásnál, 343
tájolási polinom, 346
PO[ZH]
két érintési pontos tájolás megadásnál, 343
tájolási polinom, 346
POCKET3 - négyszögzseb
kívül programozni, 700
POCKET4 - körzseb
kívül programozni, 703
polár transzformáció, 319
POLF
NC vezetésű visszahúzáshoz, 664
POLFA, 664
POLFMASK
NC vezetésű visszahúzáshoz, 664
POLFMLIN
NC vezetésű visszahúzáshoz, 664
Polinom-együttható, 259
polinom-interpoláció, 258
POLY, 258
POLYPATH, 258
PON, 623
PONS, 615
POSFS, 580
POT, 74

Pozíció tulajdonságok
    közvetetten programozni, 69
PREPRO, 176
PRESETON, 301
PRESETONS, 303
PRIO, 127
PRLOC, 29
Process DataShare, 657
Profilozás - CYCLE495
    kívül programozni, 760
Program
    -címzés,
    -elágazás, 105
    -futásidő, 652
    Inicializáló-, 222
    -ismétlés, 192
    -tároló, 212
    -ugrások, 102
Programblokk kezdete - GROUP_BEGIN, 817
programblokk vége - GROUP_END
    kívül programozni, 818
programhurok
    IF hurok, 114
    REPEAT hurok, 118
    számláló hurok, 116
    véges hurok, 116
    WHILE hurok, 118
programrész
    -ismétlés, 107
programrész ismétlés
    küzvetett CALL programozással, 196
Programtároló
    Fájltípusok, 212
    Szabványos-könyvtárak, 212
PROTA, 389
PROTD, 391
PROTS, 390
PSI
    tájolási polinom, 345
    tájolásnál a kúppalást felületen, 340
PTP, 369
PTP pontról pontra menet, 369
PTPG0, 369
PTPWOC, 369
PUD, 30
PUNCHACC, 615
PUTFTOC, 419
PUTFTOCF, 418
PW, 244

R

Rács/keret pozícióminta - CYCLE801
    kívül programozni, 765
rándítás
    -határolás, 510
    -korrekció, 536
READ, 150
REAL, 29
REDEF, 35
RELEASE, 137
Rendelkezésre állás
    rendszer-független, 5
rendszer
    -függő rendelkezésre állás, 5
rendszer frame-k, 310
Rendszerváltozók
    Mérő-tapintó állapot, 276
    mérőtapintó határolás, 277
REP, 52
REPEAT, 107
REPEATB, 107
REPOSA, 527
REPOSH, 527
REPOSHA, 527
REPOSL, 527
REPOSQ, 527
REPOSQA, 527
RET, 177
RET(paraméterezhető), 178
RETB (paraméterezhető), 184
RG, 25
RINDEX, 89
RMBBL, 527
RMEBL, 527
RMIBL, 527
RMNBL, 527
ROUND, 74
ROUNDUP, 155
RPY-szög, 332
Run MyScreens, 647

S

sapkázás, 615
sarokkésleltetés a belső sarkokon, 280
sarokkésleltetés minden sarkokon, 280
SAVE, 166
SBLOF, 168
SBLON, 168

SCPARA, 641
SD, 244
SD41610, 402
SD41611, 402
SD42475, 357
SD42476, 357
SD42477, 357
SD42900, 411
SD42910, 411
SD42920, 412
SD42930, 413
SD42935, 414
SD42940, 415
SD42984, 466
sebesség-csatolás, 583
SET, 52
SETAL, 662
SETDNO, 443
SETINT, 127
SETM, 120
SETTCOR, 483
Síkmarás - CYCLE61
    kívül programozni, 715
simítás
    tájolás lefutása, 359
SIN, 74
Skatulyázási mélység
    vezérlő-struktúráknál, 113
SLOT1 - hosszvájat
    kívül programozni, 705
SLOT2 - körvájat
    kívül programozni, 708
SOFT, 510
SOFTA, 510
Sokszög - CYCLE79
    kívül programozni, 733
SON, 615
SONS, 615
SPF, 212
SPI, 629
SPIF1, 615
SPIF2, 615
Spline
    -interpoláció, 244
    -típusok, 250
Spline-kötelék, 255
SPLINEPATH, 255
SPN, 620
SPOF, 615
SPP, 620
SPRINT, 92
SQRT, 74
START, 120
STARTFIFO, 519
STAT, 370
STOPFIFO, 519
STOPRE, 519
string,
    -hossz, 88
    -láncolása, 86
    -műveletek, 84
string-ből
    formattálás, 92
STRINGIS, 643
STRLEN, 88
SUBSTR, 90
SYNR, 29
SYNRW, 29
SYNW, 29
Szabadra szúrás - CYCLE940
    kívül programozni, 787
Számítási paraméterek
    csatorna-specifikus, 24
    globális, 25
számláló hurok, 116
Szerszám
    -paraméter, 403
    -tájolás,
    -tájolás frame-váltásnál, 452
Szerszámkorrekció
    koordinátarendszer kopásértékekre,
    Korrekció-tároló, 403
szerszámkorrekciók
    hozzáadódó, 406
szerszám-lánc, 400
Szerszámsugár-korrekció
    Sarokkésleltetés, 280
szerszámtájolás
    pálya-viszonyítású, 350
Szerszámtájolás alaphelyzet ORIRESET, 328
Szerszámtartó
    -kinematika, 445
    -tájolható, 450
Szinguláris helyek, 337
szinkronfutás
    durva, 583
    finom, 583
Szinkronorsó
    csatolás, 580
    -pár megadása, 585

T

Tájolás programozása, 338

tájolási transzformáció TRAORI
Általános 5/6-tengelyes transzformáció, 318
Gépkinematika, 318
tájolás programozás, 327
Tájolás programozás változatok, 327
tájolási mozgások, 317
tájolható szerszámtartó, 445
tájolótengelyek, 338
TAN, 74
TANG, 499
TANGDEL, 503
TANGOF, 503
TANGON, 501
TCARR, 450
TCOABS, 450
TCOFR, 450
TCOFRX, 450
TCOFRY, 450
TCOFRZ, 450
tengely
-csere, 137
Tengely vezető érték csatolás, 568
Tengelyek
vontatás-, 547
Tetszőleges pozíciók - CYCLE802
kívül programozni, 767
THETA
szerszámtájolás forgatásnál, 347
szerszámtájolás interpolációnál, 352
THETA tájolási vektor, 347
TILT, 329
TLIFT, 500
TMOF, 627
TMON, 627
TOFFOF, 453
TOFFON, 453
TOLOWER, 87
TOOLENV, 471
TOUPPER, 87
TOWBCS, 413
TOWKCS, 413
TOWMCS, 413
TOWSTD, 413
TOWTCS, 413
TOWWCS, 413
TRAANG
programozható szöggel, 364
TRACON, 367
TRACYL, 361
TRAFOOF, 381
TRAFOON, 393
TRAILOF, 545
TRAILON, 545
TRANSMIT, 361
Transzformáció billenthető lineáris tengellyel, 325
transzformáció fajták
Általános funkció, 315
Transzformációk
A szerszámtájolás kinematika-független alaphelyzete, 316
Három-, négy- és öt-tengelyes transzformáció, 326
Kinematikai transzformáció, 316
láncolt, 367
Láncolt transzformációk, 317
Tájolási transzformáció, 315
TRAORI, 326
Trigger-esemény
mérésnél, 273
TRUE, 29
TRUNC, 74
TU, 374

U

ugrás
programkezdetre, 101
ugrás-jelölőkre, 102
ugrás-jelölő
programrész ismétléseknél, 107
programugrásoknál, 102
ULI, 41
UNTIL, 118
Üzemmód
mérésnél, 274

V

vágóél-szám, 442
valósérték csatolás, 583
Változók
felhasználó által definiált, 29
Típus-konverzió, 84
Variable
Típus-konverzió, 83
védelmi tartományok, 225
végtelen hurok, 116
VELOLIM, 537
VELOLIMA, 512
Vészjelzések
NC-programban beállítani, 662
vezérlő
-struktúrák, 113

vezető érték csatolás
Valós- és parancsérték csatolás, 571
vezető és követő tengely szinkronizálása, 570
Vezető érték szimuláció, 572
vezető tengely
tengely vezetőérték csatolásnál, 568
Visszahúzás
hajtás alapú, 669
NC vezetésű, 664
Vonal pozícióminta - HOLES1
kívül programozni, 698
vontatás, 545
vontatás egyesülés, 545
Vontatott tengelyek, 547

W

WAITC, 580
WAITE, 120
WAITENC, 639
WAITM, 120
WAITMC, 120
WHEN, 601
WHEN-DO, 611
WHENEVER, 601
WHENEVER-DO, 611
WHILE, 118
WORKPIECE, 671
WRITE, 145

X

XOR, 76