

NCT[®]

101T, 104T, 115T

Eszterga vezérlő
Programozási leírás

Az x.066 SW változattól

Gyártó és fejlesztő: **NCT Ipari Elektronikai kft.**

H1148 Budapest Fogarasi út 7

✉ Levélcím: 1631 Bp. pf.26

☎ Telefon: (+36 1) 467 63 00

☎ Telefax:(+36 1) 467 63 09

Villanyposta: nct@nct.hu

Honlap: www.nct.hu

Tartalomjegyzék

1 Bevezetés	9
1.1 Az alkatrészprogram	9
Szó	9
Címlánc	9
Mondat	9
Programszám és Programnév	10
Programkezdet, Programvég	10
Programformátum a tárban	10
Programformátum külső eszközzel történő kommunikációnál	10
Főprogram és alprogram	10
DNC csatorna	11
1.2 Alapfogalmak	12
2 Vezérelt tengelyek	16
2.1 A tengelyek elnevezése	16
2.2 A tengelyek mérték-, és ikremensrendszere	16
3 Előkészítő funkciók (G kódok)	18
4 Az interpoláció	21
4.1 A pozicionálás (G00)	21
4.2 Az egyenes interpoláció (G01)	21
4.3 A kör-, és a síkbeli spirális interpoláció (G02, G03)	22
4.4 Egyenletes emelkedésű menet vágása (G33)	28
4.5 Változó emelkedésű menet vágása (G34)	29
4.6 A polárkoordináta interpoláció (G12.1, G13.1)	30
4.7 A hengerinterpoláció (G7.1)	34
5 A koordinátaadatok	36
5.1 Abszolút és inkrementális programozás (G90, G91), az I operátor	36
5.2 Inch/Metrikus átalakítás (G20, G21)	37
5.3 Koordinátaadatok megadása és értékhatára	37
5.4 Átmérőben, vagy sugárban történő programozás	38
5.5 Forgó tengelyek átfordulás kezelése	39
6 Az előtolás	42
6.1 A gyorsmeneti előtolás	42
6.2 A munkaelőtolás	42
6.2.1 Percenkénti (G94) és fordulatonkénti (G95) előtolás	43
6.2.2 A munkaelőtolás értékének behatárolása	44
6.3 Gyorsítás/lassítás. Az F előtolás figyelembe vétele	45
6.4 Az előtolásvezérlő funkciók	47
6.4.1 G09: pontos megállás	47
6.4.2 G61: pontos megállás üzemmód	47
6.4.3 G64: folyamatos forgácsolás üzemmód	47
6.4.4 G63: a százalék kapcsoló és stop tiltás üzemmód	47

6.4.5	Automatikus előtoláscsökkentés belső sarkoknál. (G62).	<u>48</u>
6.4.6	Automatikus előtoláscsökkentés belső köríveknél...	<u>49</u>
6.5	Automatikus lassítás sarkoknál.	<u>49</u>
6.6	A pálya mentén normális irányban fellépő gyorsulások korlátozása köríveknél...	<u>52</u>
7	A várakozás.	<u>53</u>
8	A referenciapont.	<u>54</u>
8.1	Automatikus referenciapont felvétel (G28).	<u>54</u>
8.2	Az 1., 2., 3., 4. referenciapontra állás (G30).	<u>55</u>
8.3	Automatikus visszatérés a referenciaponttól (G29).	<u>55</u>
9	Koordinátarendszerek, síkválasztás.	<u>57</u>
9.1	A gépi koordinátarendszer.	<u>57</u>
9.1.1	A gépi koordinátarendszer beállítása...	<u>57</u>
9.1.2	A gépi koordinátarendszer kiválasztása (G53).	<u>58</u>
9.2	A munkadarab koordinátarendszerek.	<u>58</u>
9.2.1	A munkadarab koordinátarendszerek beállítása.	<u>58</u>
9.2.2	A munkadarab koordinátarendszer kiválasztása.	<u>59</u>
9.2.3	A munkadarab koordinátarendszerek eltolásának állítása programból.	<u>60</u>
9.2.4	Új munkadarab koordinátarendszer létrehozása (G92).	<u>60</u>
9.3	A lokális koordinátarendszer.	<u>61</u>
9.4	Síkválasztás (G17, G18, G19).	<u>63</u>
10	A főorsó funkció.	<u>65</u>
10.1	A főorsó fordulatszám parancs (S kód).	<u>65</u>
10.2	A konstans vágósebesség számítás programozása..	<u>65</u>
10.2.1	A konstans vágósebesség számítás megadása (G96, G97).	<u>65</u>
10.2.2	A konstans vágósebesség értékének behatárolása (G92).	<u>66</u>
10.2.3	Tengely kijelölése a konstans vágósebesség számításához.	<u>67</u>
10.3	A főorsó pozícióvisszacsatolás.	<u>67</u>
10.4	Orientált főorsó megállás..	<u>67</u>
10.5	A főorsó pozícionálás (indexálás).	<u>68</u>
10.6	A főorsó fordulatszám ingadozás figyelése (G25, G26).	<u>68</u>
11	A szerszámkezelés.	<u>71</u>
12	Vegyes és segédfunkciók.	<u>72</u>
12.1	Vegyes funkciók (M kódok).	<u>72</u>
12.2	Segédfunkciók (A, B, C kódok).	<u>73</u>
12.3	A különböző funkciók végrehajtási sorrendje.	<u>73</u>
13	Az alkatrészprogram szervezése.	<u>74</u>
13.1	A mondatszám (N cím).	<u>74</u>
13.2	Feltételes mondatkihagyás (/ cím).	<u>74</u>
13.3	Főprogram és alprogram.	<u>74</u>
13.3.1	Az alprogram hívása.	<u>75</u>
13.3.2	Visszatérés alprogramból.	<u>75</u>
13.3.3	Ugrás főprogramon belül.	<u>76</u>

14 A szerszámkorrekció	78
14.1 Hivatkozás szerszámkorrekcióra.	<u>78</u>
14.2 Szerszámkorrekciós értékek módosítása programból (G10).	<u>82</u>
14.3 A szerszámhossz–korrekció figyelembe vétele.	<u>82</u>
14.4 A szerszámsugár korrekció (G38, G39, G40, G41, G42)..	<u>84</u>
14.4.1 A sugárkorrekció számítás bekapcsolása. Ráállítás a kontúrra... ..	<u>87</u>
14.4.2 A sugárkorrekció számítás bekapcsolt állapota. Haladás a kontúron... ..	<u>91</u>
14.4.3 A szerszámsugár korrekciószámítás kikapcsolása. Leállítás a kontúrról..	<u>94</u>
14.4.4 Irányváltás a sugárkorrekció számításban... ..	<u>97</u>
14.4.5 A vektor megtartás programozása (G38)..	<u>99</u>
14.4.6 Sarokív programozása (G39)..	<u>100</u>
14.4.7 Általános tudnivalók a síkbeli sugárkorrekció alkalmazása esetére.	<u>101</u>
14.4.8 A kontúrkövetés zavarproblémái. Interferenciavizsgálat..	<u>106</u>
15 Különleges transzformációk	110
15.1 Tükrözés kettős szerszámtartó esetén (G68).	<u>110</u>
15.2 Léptékezés (G50, G51)..	<u>111</u>
15.3 Tükrözés (G50.1, G51.1).	<u>112</u>
16 Automatikus geometriai számítások	113
16.1 Letörés és lekerekítés programozása.	<u>113</u>
16.2 Egyenes megadása irányszögével..	<u>114</u>
16.3 Síkbeli metszéspontszámítások..	<u>116</u>
16.3.1 Két egyenes metszéspontja.	<u>116</u>
16.3.2 Egyenes és kör metszéspontja.	<u>118</u>
16.3.3 Kör és egyenes metszéspontja.	<u>120</u>
16.3.4 Két kör metszéspontja.	<u>122</u>
16.3.5 A metszéspontszámítások láncolása.	<u>124</u>
17 Esztergáló ciklusok	125
17.1 Egyszerű ciklusok.....	<u>125</u>
17.1.1 A hosszesztergáló ciklus (G77).	<u>125</u>
17.1.2 Az egyszerű menetvágó ciklus (G78).	<u>127</u>
17.1.3 Az oldalazó ciklus (G79).	<u>129</u>
17.1.4 Az egyszerű ciklusok használata.	<u>131</u>
17.2 Összetett ciklusok.....	<u>132</u>
17.2.1 Nagyoló ciklus (G71)..	<u>132</u>
17.2.2 Homlok nagyoló ciklus (G72).	<u>137</u>
17.2.3 Minta ismétlő ciklus (G73).	<u>139</u>
17.2.4 Símitó ciklus (G70).	<u>141</u>
17.2.5 Homlok beszűrő ciklus (G74).	<u>142</u>
17.2.6 Beszűrő ciklus (G75).	<u>144</u>
A menetvágó ciklus (G76)..	<u>146</u>
18 Fúróciklusok	152
18.1 A fúróciklusok részletes leírása... ..	<u>158</u>
18.1.1 Nagysebességű mélyfúróciklus (G83.1).	<u>158</u>
18.1.2 Balmenetfúró ciklus (G84.1).	<u>159</u>
18.1.3 Kiesztergálás automatikus szerszámelhúzással (G86.1).	<u>160</u>
18.1.4 A ciklusállapot kikapcsolása (G80).	<u>161</u>

18.1.5 Fúróciklus, kiemelés gyorsmenettel (G81).	161
18.1.6 Fúróciklus várakozással, kiemelés gyorsmenettel (G82).	162
18.1.7 Mélyfúróciklus (G83).	163
18.1.8 Menetfúró ciklus (G84).	164
18.1.9 Menetfúró ciklus kiegyenlítő betét nélkül (G84.2, G84.3).	165
18.1.10 Fúróciklus, kiemelés előtolással (G85).	168
18.1.11 Fúróciklus, gyorsmeneti kiemelés álló főorsóval (G86).	169
18.1.12 Fúróciklus, kézi működtetés a talpponton/ Kiesztergálás visszafelé, automatikus szerszámelhúzással (G87).	170
18.1.13 Fúróciklus, várakozás után kézi működtetés a talpponton (G88).	172
18.1.14 Fúróciklus, talpponton várakozás, kiemelés előtolással (G89).	173
18.2 Megjegyzések a fúróciklusok használatához.	173
19 A sokszögesztergálás.	175
19.1 A sokszögesztergálás működési elve.	175
19.2 A sokszögesztergálás programozása (G51.2, G50.2).	176
20 Mérőfunkciók.	178
20.1 Mérés a maradék út törlésével (G31).	178
20.2 Automatikus szerszámhossz mérés (G36, G37).	179
20.3 Összefüggés a mérés pontossága és az alkalmazott előtolás között.	180
21 Biztonsági funkciók.	181
21.1 Programozható munkatér behatárolás (G22, G23).	181
21.2 Paraméteres végállás.	182
21.3 Tiltott tartomány figyelés mozgásindítás előtt.	183
22 A felhasználói makró.	184
22.1 Az egyszerű makróhívás (G65).	184
22.2 Az öröklődő makróhívás.	185
22.2.1 Makróhívás minden mozgásparancs után: (G66).	185
22.2.2 Makróhívás minden mondatból: (G66.1).	186
22.3 Felhasználói makróhívás G kódra.	187
22.4 Felhasználói makróhívás M kódra.	188
22.5 Felhasználói alprogramhívás M kódra.	188
22.6 Felhasználói alprogramhívás T kódra.	189
22.7 Felhasználói alprogramhívás S kódra.	189
22.8 Felhasználói alprogramhívás A, B, C kódra.	189
22.9 Az alprogramhívás és makróhívás közti különbség.	190
22.9.1 Többszörös hívás.	190
22.10 A felhasználói makró formátuma.	191
22.11 A programnyelv változói.	191
22.11.1 Változó azonosítása.	192
22.11.2 Hivatkozás változóra.	192
22.11.3 Üres változók.	192
22.11.4 A változók számábrázolása.	192
22.12 A változók típusai.	193
22.12.1 Lokális változók.	193
22.12.2 Globális változók.	194
22.12.3 Rendszerváltozók.	194

22.13 A programnyelv utasításai.	203
22.13.1 Az értékadó utasítás.	203
22.13.2 Aritmetikai műveletek és függvények.	203
22.13.3 Feltételes kifejezések.	206
22.13.4 Feltétel nélküli elágazás.	206
22.13.5 Feltételes elágazás.	207
22.13.6 Feltételes utasítás.	207
22.13.7 Ciklusszervezés.	207
22.13.8 Adatkiadási parancsok.	210
22.14 NC és makró utasítások.	213
22.15 A makromondatok végrehajtása.	214
22.16 Makrók és alprogramok kijelzése automata üzemmódban.	215
22.17 STOP gomb használata makróutasítás végrehajtása közben.	215
Jegyzetek.	216
Betűrendes index.	217

09.11.11

© Copyright NCT November 11, 2009

E leírás tartalmára minden kiadói jog fenntartva. Utánnymáshoz – kivonatosan is – engedélyünk megszerzése szükséges.

A leírást a legnagyobb körültekintéssel állítottuk össze és gondosan ellenőriztük, azonban az esetleges hibákért vagy téves adatokért *és az ebből eredő károkért felelősséget nem vállalunk.*

1 Bevezetés

1.1 Az alkatrészprogram

Az alkatrészprogram olyan utasítások halmaza, amelyeket a vezérlés értelmezni képes és amelyek alapján a gép működését irányítja.

Az alkatrészprogram mondatokból áll. A mondatokat szavak alkotják.

Szó: Cím és Adat

A szó két részből tevődik össze: címből és adatból. A cím egy vagy több karakter, az adat pedig numerikus érték, amelynek lehet egész és tizedes értéke is. Bizonyos címek kaphatnak előjelet, illetve I operátort.

Címlánc:

Címek	Jelentés	Értékhatár
O	programszám	0001 - 9999
/	opcionális mondat	1 - 9
N	mondatszám	1 - 99999
G	előkészítő funkció	*
X, Y, Z, U, V, W	hosszkoordináták	I, -, *
A, B, C, H	szögkoordináta	I, -, *
R	körsugár, segédadat	I, -, *
I, J, K	kör középpont koordináta, segédkoordináta	-, *
E	segédkoordináta	-, *
F	előtolás	*
S	főorsó fordulatszám	*
M	vegyes funkció	1 - 999
T	szerszámszám/korrekció száma	1 - 9999
L	ismétlési szám	1 - 9999
P	segédadat, várakozási idő	-, *
Q	segédadat	-, *
,C	letörés szárhossza	-, *
,R	lekerekítés sugara	-, *
,A	egyenes irányszöge	-, *
(megjegyzés	*

Azoknál a címeknél, amelyeknél a * jel látható az értékhatár oszlopban, az adat tizedes értéket is felvehet.

Azoknál a címeknél, ahol az I jel és a – jel látható, a címre adható inkrementális operátor illetve előjel.

Nem jelezzük ki, és nem tároljuk a + jelet.

Mondat

A mondat szavakból tevődik össze.

A mondatokat a tárban LF (Line Feed) karakter választja el egymástól. A mondatokban nem kö-

telező a mondatszám használata. Hogy a mondatvéget és a mondatkezdetet el lehessen különíteni egymástól a képernyőn a mondat elejét új sorban kezdjük, és > karaktert teszünk eléje és az egy sornál hosszabb mondatoknál az új sorba kerülő szavakat két karakterrel beljebb kezdjük. A > karaktert az N és / címek szóköz nélkül követik, ezzel is kiemelve az így megkülönböztetett mondatokat.

A tárban a szavak között nincs szóköz (space), a kijelzés során azonban a szavak közé szóköz kerül. Ha egy szó nem fér ki teljes egészében egy sorban, akkor az adott szó új sorba kerül.

Programszám és Programnév

A programszám és programnév a program azonosítására szolgál. A programszám használata kötelező, a programnévé nem.

A **programszám** címe: **O**, melyet pontosan **négy számjegynek** kell követni.

A **programnév** nyitó "(" és záró ")" zárójelek közötti tetszőleges karaktersorozat. Legfeljebb 16 karaktert tartalmazhat.

A programszámot és programnevet a többi programmondattól LF (**Line Feed**) karakter választja el a tárban.

A képernyőn a szerkesztés során az első sorban a programszám és programnév mindig kijelzésre kerül.

A háttértárban két azonos programszámú állomány nem lehet.

Programkezdet, Programvég

A program % karakterrel kezdődik, és % karakterrel végződik. A programszerkesztés során a programzáró karakter mindig az utolsó mondat után áll, így biztosítható, hogy a már lezárt mondatok akkor is megőrződnek, ha programszerkesztés közben áramkimaradás történik.

Programformátum a tárban

A tárban elhelyezkedő program ASCII karakterek halmaza.

A program formátuma:

```
%O1234 (PROGRAMNEV) LF/1N12345G1X0Z...LFG2Z5...LF....LF
...LF
...LF
N1G40...M2LF
%
```

A fenti karaktersorban

LF az LF karaktert (**Line Feed**),

% a programkezdetet, -véget,

szimbolizálja.

Programformátum külső eszközzel történő kommunikációnál

A fenti programformátum érvényes külső eszközzel történő kommunikáció során is.

Főprogram és alprogram

Az alkatrészprogramokat két fő részre lehet osztani:

főprogramokra, és

alprogramokra.

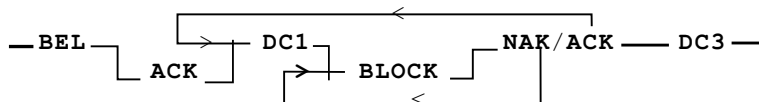
Az alkatrész megmunkálását a főprogram írja le. Ha a megmunkálás során ismétlődő mintákat kell különböző helyeken megmunkálni, akkor ezeket a programszakaszokat nem kell ismét leírni a főprogramban, hanem alprogramot kell rá szervezni, amely tetszőleges helyről hívható, akár egy másik alprogramból is. Az alprogramból vissza lehet térni a hívó programba.

DNC csatorna

Lehetőség van egy külső egységen (például számítógépen) lévő program végrehajtására is, anélkül, hogy azt a vezérlés memóriájában tárolnánk. Ekkor a vezérlés nem a memóriából olvassa ki a programot, hanem az RS232 felületen keresztül a külső adathordozóból. Ezt a kapcsolatot nevezzük DNC csatornának. Ez a módszer különösen hasznos olyan programok végrehajtásánál, melyek méretük folytán bele sem férnének a vezérlés memóriájába.

A DNC csatorna egy protokollal vezérelt adatátviteli csatorna az alábbiak szerint:

Vezérlés:
Küldő:



Ahol a mnemonikok jelentése (és ASCII kódjuk):

BEL (7): a vezérlés felszólítja a küldő oldalt a kapcsolat felvételére. Ha egy bizonyos időn belül nem érkezik ACK a vezérlés újra BEL-t ad ki.

ACK (6): nyugtázás.

NAK (21): hibás adatátvitel (például hardver hiba a vonalon vagy BCC hiba), a BLOCK átvitelét meg kell ismételni.

DC1 (17): induljon a következő BLOCK átvitele.

DC3 (19): kapcsolat megszakítása.

BLOCK

- alapvetően egy NC mondat (a mondatot lezáró LF is), és ezek összege (BCC) 7 biten tárolva a mondat utolsó byte-jaként (a BCC 7., legfelső bitje minden esetben 0). A mondatban nem lehet SPACE (32), vagy annál kisebb ASCII kódú karakter.
- **EOF** (26) (**End Of File**) a küldő file vége jelet küld és ezzel megszakítja a kapcsolatot.

A DNC üzemmódhoz a második fizikai csatornát (csak ez használható DNC csatornának) állítsuk 8 bit páros paritás módba.

A vezérlésen automata üzemmódban a DNC funkciógombbal kapcsolhatjuk ki-be a DNC állapotot. A DNC csatornáról végrehajtott főprogram csak lineáris szekvenciájú lehet. Ez nem vonatkozik az esetlegesen hívott alprogramokra vagy makrókra, viszont ezeknek a vezérlés memóriájában kell lenni. Ha a főprogramban eltérünk a lineáris szekvenciától (GOTO, DO WHILE) a vezérlés 3058 NEM LEHET DNC KÖZBEN hibaüzenetet ad. Ha a vezérlés BLOCK hibát érzékel és NAK-kal válaszol meg kell ismételni a BLOCK-ot.

1.2 Alapfogalmak

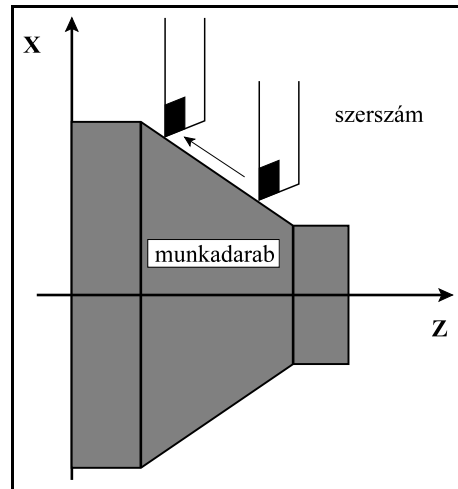
Az interpoláció

A vezérlés a megmunkálás során a szerszámot egyenes- és körpálya mentén képes mozgatni. Ezt a tevékenységet a továbbiakban interpolációnak nevezzük.

Szerszámmozgás egyenes mentén:

program:

```
G01 Z__
X__ Z__
```

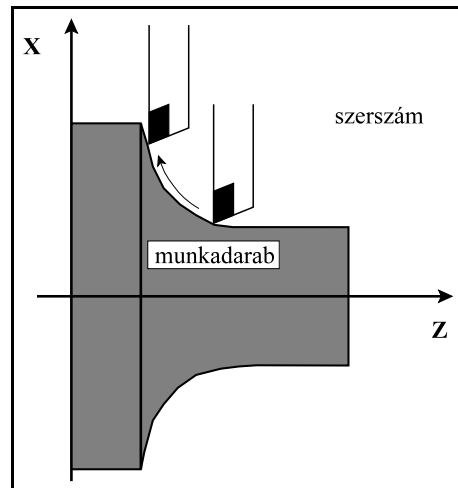


1.2-1 ábra

Szerszámmozgás körív mentén:

program:

```
G02 X__ Z__ R__
```



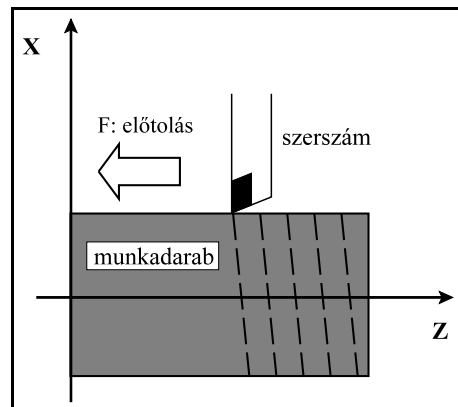
1.2-2 ábra

Előkészítő funkciók (G kódok)

Egy adott mondat által végrehajtandó tevékenység típusát az előkészítő funkciók, vagy más néven G kódok segítségével írjuk le. Például: a G01 kód egyenes interpolációt vezet be.

Előtolás

Előtolásnak nevezzük a szerszámnak a munkadarabhoz viszonyított sebességét a forgácsolás közben. Programban F címen és egy számértékkel adhatjuk meg a kívánt előtolást. Például: F2 jelentése 2 mm/fordulat.



1.2-3 ábra

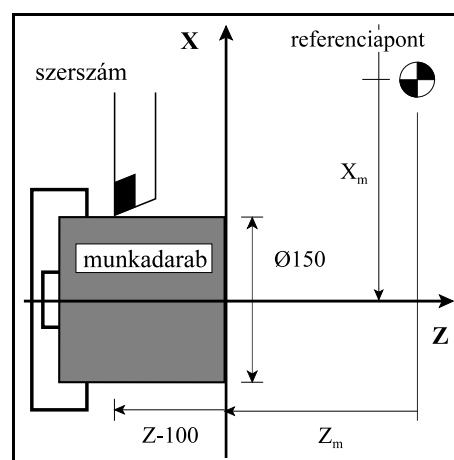
Referenciapont

A referenciapont a szerszámgepen egy fix pont. A gép bekapcsolása után a szánokat referencia-pontra kell küldeni. Ezután a vezérlés már abszolút koordinátájú adatokat is tud értelmezni.

Koordinátarendszer

Az alkatrész rajzán feltüntetett méretek az alkatrész egy adott pontjához képest értendők. Ez a pont a munkadarab-koordinátarendszer nullpontja. Az alkatrészprogramba ezeket a méretadatokat kell beírni a koordinátacímekre. Például: X150 Z-100 jelentése: a munkadarab koordináta-rendszer 150; és -100 mm koordinátájú pontja X és Z irányban.

Ahhoz, hogy a vezérlés a programozott koordinátaadatok értelmezni tudja meg kell adni a referenciapont és a munkadarab nullpont közti távolságot. Ez a munkadarab nullpont bemérésével történik.



1.2-4 ábra

Abszolút koordinátamegadás

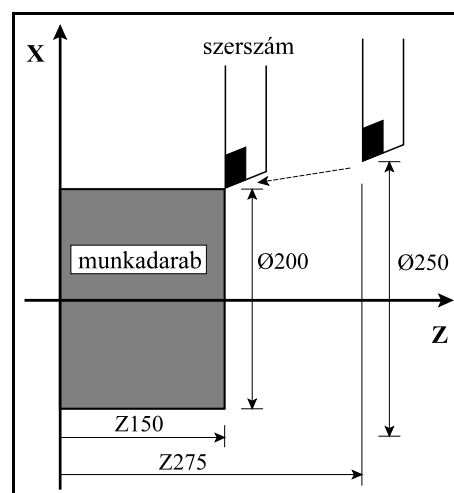
Abszolút koordinátamegadás esetén a szerszám a koordinátarendszer kezdőpontjától számított távolságra mozog, azaz a koordinátán megadott pozíciójú pontra.

Az abszolút adatmegadás kódja: G90.

A

```
G90 X200 Z150
```

utasítással a fenti pozíciójú pontra mozgatja a szerszámot, bárhol is állt a parancskiadás előtt.



1.2-5 ábra

Növekményes (inkrementális) koordinátamegadás

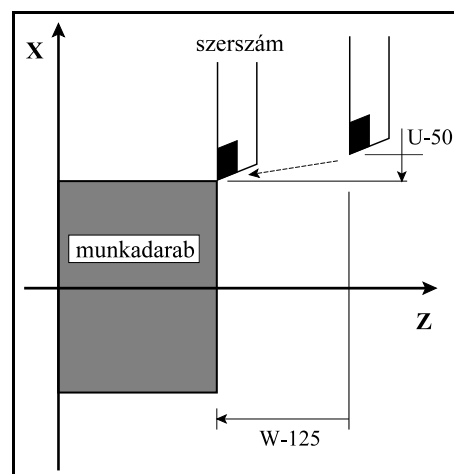
Növekményes koordinátamegadás esetén a vezérlés a koordinátaadatot úgy értelmezi, hogy a szerszám a pillanatnyi pozíciótól számított távolságra mozogjon:

```
U-50 W-125
```

Az inkrementális adatmegadás kódja: G91. A G91 kód az összes koordinátaértékre vonatkozik. A fenti utasítással a

```
G91 X-50 Z-125
```

utasítással is ki lehet fejezni.



1.2-6 ábra

Átmérőben történő programozás

Az X irányú méretet paraméter beállítás alapján lehet átmérőben is programozni.

Öröklődő funkciók

A programnyelvben bizonyos utasítások hatása, vagy értékük nagysága öröklődik mindaddig, amíg ellenkező értelmű parancsot nem adunk ki, vagy más értéket nem adunk a megfelelő funkciónak. Például: Az

```
N15 G90 G1 X20 Z30 F0.2  
N16 X30  
N17 Z100
```

programrészletben az N15 mondatban felvett G90 (abszolút adatmegadás) és a G1 (lineáris interpoláció) állapota, illetve F (előtolás) értéke öröklődik az N16-os és N17-es mondatokban. Így nincs szükség ezeket a funkciókat mondatról-mondatra megadni.

Nem öröklődő (egylövetű) funkciók

Bizonyos funkciók hatása, vagy adatok értéke csak az adott mondatban érvényes. Ezeket a funkciókat nem öröklődő, vagy egylövetű funkcióknak nevezzük.

Főorsó fordulatszám parancs

A főorsó fordulatszámot S címen lehet megadni. Ezt szokás még S funkciónak is nevezni. Az S1500 utasítás azt mondja meg, hogy a főorsó 1500 ford/perces fordulatszámmal forogjon.

Konstans vágósebesség számítás

A vezérlés automatikusan úgy változtatja a főorsó fordulatszámát az átmérő függvényében, hogy a szerszám hegyének a munkadarab felületéhez képesti sebessége állandó legyen. Ezt nevezzük konstans vágósebesség számításnak.

Szerszámszám

A megmunkálás során különböző szerszámokkal kell a különböző forgácsolási műveleteket elvégezni. A szerszámokat számokkal különböztetjük meg egymástól. A szerszámokra T kóddal hivatkozhatunk. A T kód első két számjegye a szerszám kódja (vagyis hányadik pozícióban található a revolverfejben), a T kód második két számjegye pedig a kiválasztott szerszámhoz tartozó korrekciós csoport száma. A programban a

```
T0212
```

utasítás azt jelenti, hogy a 02-es szerszámot választottuk és a 12-es korrekciós csoportot rendeltük hozzá.

Vegyes funkciók

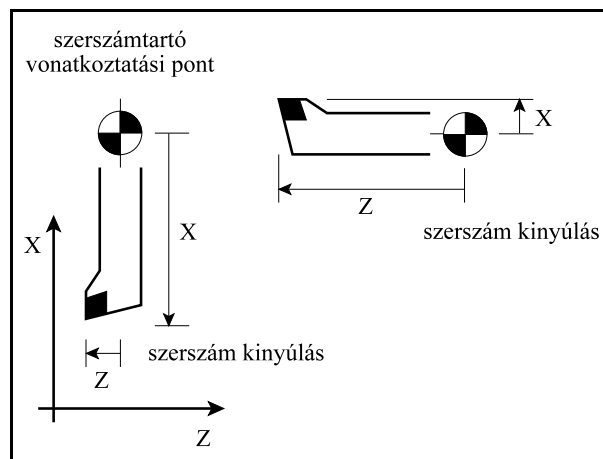
A megmunkálás során számos ki-, bekapcsolási műveletet kell elvégezni. Például: elindítani a főorsót, bekapcsolni a hűtővizet. Ezeket a műveleteket a vegyes vagy M funkciók segítségével lehet elvégezni. Például: az

```
M3 M8
```

utasításokban M3 jelentése: főorsó forgás óramutatóval megegyező irányba, M8 jelentése pedig: kapcsolod be a hűtővizet.

Hosszkorrekció

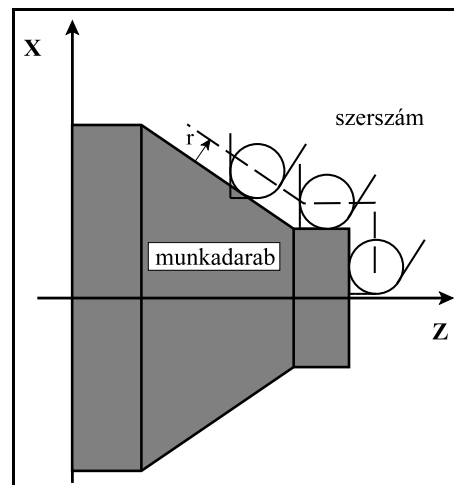
A megmunkálás során különböző hosszúságú szerszámokkal végezzük a különböző műveleteket. Ugyanazt a műveletet viszont egy nagyobb széria gyártása esetén, például a szerszám törése miatt, szintén különböző hosszúságú szerszámmal kell végezni. Annak érdekében, hogy az alkatrészprogramban leírt mozgások függetlenek legyenek a szerszám hosszától, azaz kinyúlásától, a vezérléssel közölni kell a különböző szerszámhosszakat. Ehhez a szerszámok hosszát be kell mérni. Ha a programban azt akarjuk, hogy a szerszám csúcsa mozogjon a megadott pontra, le kell hívni annak a hosszadatnak az értékét, amelyet bemérés során megadtunk. Ez a T kód második két helyiértékén történik. Ettől kezdve a szerszám csúcsát mozgatja a vezérlő a megadott pontra.



1.2-7 ábra

Sugárkorrekció

Egy kontúr esztergálásánál, ha nem a tengelyekkel párhuzamos mozgást végez a szerszám, akkor kapunk pontos méretet, ha nem a szerszám hegyét vezetjük a kontúron, hanem a szerszám sugar középpontját vezetjük a kontúrra merőlegesen, attól sugárnyi (r) távolságra. Annak érdekében, hogy a programban ne a szerszám középpont pályáját kelljen leírni, figyelembe véve a szerszámok sugarát, hanem a darab tényleges kontúradatait, be kell vezetni a sugárkorrekciót. A programban a T címen lehívott korrekciós csoportban kell megadni a szerszám sugarát.



1.2-8 ábra

2 Vezérelt tengelyek

Tengelyek száma alapképzésben	2 tengely
Bővítőtengelyek száma	6 tengely (8 tengely összesen)
Egyidejűleg mozgatható tengelyek száma	8 tengely (lineáris interpolációval)

2.1 A tengelyek elnevezése

A vezérelt tengelyek elnevezését a paramétertárban lehet definiálni. Itt ki lehet jelölni, hogy melyik fizikai tengely milyen címre mozogjon.

Alapképzésben a tengelyek nevei: X és Z.

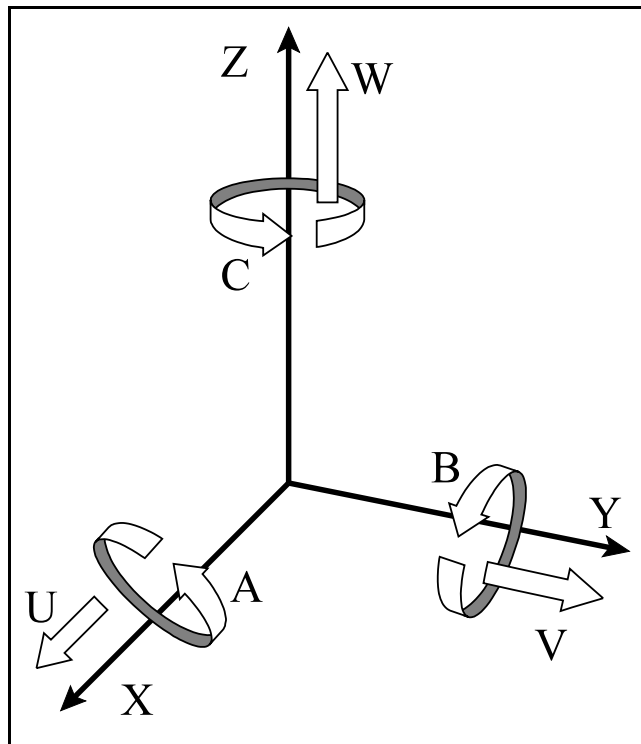
A bővítőtengelyek elnevezése a tengely típusától függ.

A lineáris mozgást végző bővítőtengelyek lehetséges elnevezése: Y, U, V és W. Ha U, V, W tengelyek valamelyik főiránnyal párhuzamosak, akkor az X tengellyel párhuzamos bővítőtengely neve U, az Y-nal párhuzamos neve V, és a Z-vel párhuzamos neve W.

A forgómozgást végző tengelyek nevei: A, B, és C. Az X iránnyal párhuzamos tengelyű forgótengely neve A, az Y-nal párhuzamos neve B, és a Z-vel párhuzamos pedig C.

A főorsó tengely neve, ha polár-, vagy hengerkoordináta interpolációt használunk: C.

Amennyiben a gépen nem található U, V, vagy W nevű tengely a felsorolt címeken sorrendben az X, Y, Z irányú inkrementális elmozdulás adható meg. C irányú inkrementális elmozdulás esetén H cím használható.



2.1-1 ábra

2.2 A tengelyek mérték-, és ikremensrendszere

Koordinátaadatokat 8 számjegyen lehet megadni. A koordinátaadatoknak lehet előjele is. A + előjelet nem tesszük ki a szám elé.

A bemenő hosszkoordináták adatait meg lehet adni mm-ben és inchben. Ez a bemeneti mértékrendszer. A bemeneti mértékrendszert a programból lehet kiválasztani.

A gépre felszerelt útmérő eszköz mérheti a pozíciót mm-ben és inchben. Az útmérő eszköz határozza meg a kimeneti mértékrendszert, amit a vezérlésnek paraméteren kell megadni. Egy gépen belül nem lehet a mértékrendszereket a tengelyek között keverni.

Amennyiben a be- és kimeneti mértékrendszer különböző az átváltást a vezérlés automatikusan

végzi.

A forgástengelyek mértékrendszere mindig fok.

A legkisebb beadható méretet tekintjük a vezérlés **bemeneti inkremensrendszerének**. A vezérlés bemeneti inkremensrendszerét paraméterrel lehet kiválasztani. Háromféle rendszer között lehet választani: IR-A , IR-B és IR-C. Egy gépen belül nem lehet az inkremensrendszereket a tengelyek között keverni.

A vezérlés a bemenő adatok feldolgozása után a tengelyek mozgásához útdatokat ad ki. Ezeknek az adatoknak a felbontása mindig a bemenő inkremensrendszer kétszerese. Ezt nevezzük a vezérlés **kimeneti inkremensrendszerének**.

A vezérlés bemeneti inkremensrendszerét tehát az útmérők felbontása határozza meg.

Inkremensrendszer	Legkisebb beadható méret	Legnagyobb beadható méret
IR-A	0.01 mm	999999.99 mm
	0.001 inch	99999.999 inch
	0.01 fok	999999.99 fok
IR-B	0.001 mm	99999.999 mm
	0.0001 inch	9999.9999 inch
	0.001 fok	99999.999 fok
IR-C	0.0001 mm	9999.9999 mm
	0.00001 inch	999.99999 inch
	0.0001 fok	9999.9999 fok

Az X tengely koordinátaadatait értelmezheti a vezérlés átmérőben is, ha az 4762 DIAM paraméter 1. A fenti táblázatban közölt értékhatárok ekkor átmérőben értendők, és nagyságuk változatlan.

3 Előkészítő funkciók (G kódok)

A G cím és az azt követő szám határozza meg az adott mondatban a parancs jellegét. A következő táblázat tartalmazza a vezérlés által értelmezett G kódokat, csoportosításukat és funkciójukat.

G kód	Csoport	Funkció	Oldal
G00*	01	pozicionálás	21
G01*		egyenes interpoláció	21
G02		kör interpoláció ójm.	22
G03		kör interpoláció óje.	22
G04	00	várakozás	53
G05.1		többszörös előfeldolgozás üzemmód beállítása	
G07.1		hengerkoordináta interpoláció	34
G09		pontos megállás az adott mondatban	47
G10		programozott adatbeadás	60, 82
G11		programozott adatbeadás tiltása	
G12.1	26	polárkoordináta interpoláció be	30
G13.1*		polárkoordináta interpoláció ki	30
G17	02	$X_p Y_p$ sík választása	63
G18*		$Z_p X_p$ sík választása	63
G19		$Y_p Z_p$ sík választása	63
G20	06	inches adatmegadás	37
G21		metrikus adatmegadás	37
G22*	04	munkatér behatárolás bekapcsolása	181
G23		munkatér behatárolás kikapcsolása	181
G25*	25	főorsó fordulatszám ingadozás figyelés kikapcsolása	68
G26		főorsó fordulatszám ingadozás figyelés bekapcsolása	68
G28	00	programozott referenciapont felvétel	54
G29		visszatérés a referenciaponttól	55
G30		első, második, harmadik és negyedik referenciapontra állás	55
G31		mérés a maradék út törlésével	178
G33	01	menetvágás	28
G34		változó emelkedésű menet vágása	29
G36	00	automatikus szerszámhosszmérés X irányban	179
G37		automatikus szerszámhosszmérés Z irányban	179
G38		sugárkorrekciós vektor megtartása	99
G39		sarokív sugárkorrekcióval	100
G40*	07	szerszám-sugár-korrekció számítás kikapcsolása	84, 94

G kód	Csoport	Funkció	Oldal
G41		szerszámsugár-korrekción számítás balról	84, 87
G42		szerszámsugár-korrekción számítás jobbról	84, 87
G50*	11	léptékezés kikapcsolása	111
G51		léptékezés	111
G50.1*	18	tükrözés kikapcsolása	112
G51.1		tükrözés bekapcsolása	112
G51.2	20	sokszögestergálás be	176
G50.2		sokszögestergálás ki	176
G52	00	koordinátaeltolás	61
G53		pozicionálás a gép koordinátarendszerében	58
G54*	14	első munkadarab koordinátarendszer választása	59
G55		második munkadarab koordinátarendszer választása	59
G56		harmadik munkadarab koordinátarendszer választása	59
G57		negyedik munkadarab koordinátarendszer választása	59
G58		ötödik munkadarab koordinátarendszer választása	59
G59		hatodik munkadarab koordinátarendszer választása	59
G61	15	pontos megállás üzemmód	47
G62		előtoláscsökkentés sarkoknál	48
G63		override tiltás	47
G64*		folyamatos forgácsolás üzemmód	47
G65		egyszerű makrohívás	184
G66		öröklődő makrohívás minden mozgásparancs után	185
G66.1		öröklődő makrohívás minden mondatból	186
G67		öröklődő makrohívás törlése	185
G68	16	tükrözés kettős szerszámtartóra bekapcsolása	110
G69*		tükrözés kettős szerszámtartóra kikapcsolása	110
G70	00	símító ciklus	141
G71		nagyoló ciklus	132
G72		homlok nagyoló ciklus	137
G73		minta ismétlő ciklus	139
G74		homlok beszúró ciklus	142
G75		beszúró ciklus	144
G76		menetvágó ciklus	146
G77	01	hosszesztergáló ciklus	125
G78		egyszerű menetvágó ciklus	127
G79		oldalazó ciklus	129
G80*	09	ciklusállapot kikapcsolása	161

G kód	Csoport	Funkció	Oldal
G81		fűrőciklus, kiemelés gyorsmenettel	161
G82		fűrőciklus várakozással, kiemelés gyorsmenettel	162
G83		mélyfűrőciklus	163
G83.1		nagysebességű mélyfűrőciklus	158
G84		menetfűrőciklus	164
G84.1		balmenetfűrő ciklus	159
G84.2		jobbmenetfűrőciklus kiegyenlítőbetét nélkül	165
G84.3		balmenetfűrőciklus kiegyenlítőbetét nélkül	165
G85		fűrőciklus, kiemelés előtollással	168
G86		fűrőciklus, gyorsmeneti kiemelés álló főorsónál	169
G86.1		kiesztérgálás automatikus szerszámelhúzással	160
G87		kiesztérgálás visszafelé, automatikus/kézi szerszámelhúzással	170
G88		fűrőciklus, kézi működtetés a talpponton	172
G89		fűrőciklus, talpponton várakozás, kiemelés előtollással	173
G90*	03	abszolút méretmegadás	36
G91*	03	növekményes méretmegadás	36
G92	00	koordinátarendszer beállítás	60
G94*	05	percenkénti előtollás	43
G95*	05	fordulatonkénti előtollás	43
G96	13	konstans vágósebesség-számítás bekapcsolása	65
G97*	13	konstans vágósebesség-számítás kikapcsolása	65
G98*	10	visszatérés fűrőciklusból a kiindulási pontra	153
G99	10	visszatérés fűrőciklusból az R (megközelítési) pontra	153

 **Megjegyzések:**

- Egy csoporton belül a *-gal jelölt G kódok azt az állapotot jelentik, amit a vezérlés bekapcsolás után felvesz.
- Ahol egy csoporton belül több kód után is * jel található ott paraméter alapján lehet kiválasztani, melyik legyen érvényes bekapcsolás után. Ezek a következők: G00, G01; G17, G18; G90, G91; G94, G95.
- G20 és G21 közül bekapcsoláskor az lesz érvényes, amelyik kikapcsoláskor be volt állítva.
- A G05.1 parancs bekapcsolás utáni alapértelmezését a *MULBUF* paraméteren lehet megadni.
- A 00 csoportba tartozó G kódok nem öröklődők, az összes többi igen.
- Egy mondatba több G kód is írható, azzal a megkötéssel, hogy azonos csoportba tartozó funkciók közül csoportonként csak egy szerepelhet.
- Illegális G kódra történő hivatkozás, vagy több, azonos csoporthoz tartozó G kód egy mondaton belüli megadása *3005 ILLEGÁLIS G KÓD* hibajelzést eredményez.

4 Az interpoláció

4.1 A pozícionálás (G00)

A

G00 v

utasításkor az aktuális koordinátarendszerben való pozícionálásra vonatkozik.

A pozícionálás a v koordinátájú pontra történik. A v jelölés itt (és a továbbiakban) az adott szerzőszámgépen használt összes vezérelt tengelyre vonatkozik. (Ezek lehetnek: X, Y, Z, U, V, W, A, B, C) Például:

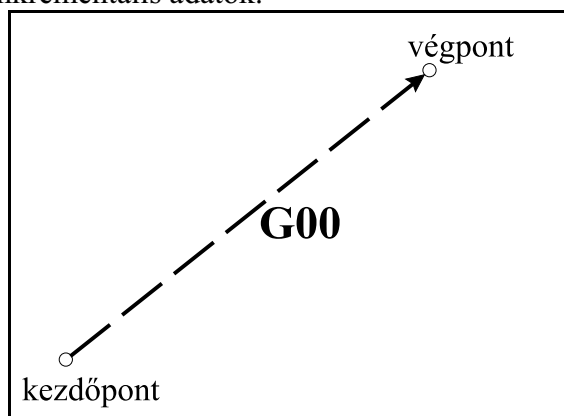
G00 X(U) ___ Z(W) ___

ahol X, Y abszolút méretmegadásra, U, W pedig inkrementális méretmegadásra vonatkozik (ha U, W nincs tengelynek kijelölve).

A pozícionálás a mondatban megadott összes tengely egyidejű mozgásával, egyenes pálya mentén történik. A koordináták lehetnek abszolút és inkrementális adatok.

A pozícionálás sebességét nem lehet programból állítani, az a szerzőszámgép építője által paraméteresen meghatározott, tengelyenként különböző értékkel történik. Több tengely egyidejű mozgása esetén a sebesség vektori eredőjét úgy számítja ki a vezérlő, hogy a pozícionálás minimális idő alatt történjék, és a sebesség egyik tengelyen se lépje túl az arra a tengelyre beállított gyorsmeneti értéket.

A G00 utasítás végrehajtása során a mozgás indításakor a vezérlés minden esetben lineáris gyorsítást, a mozgás befejezésekor lineáris lassítást hajt



4.1-1 ábra

végre. A mozgás befejezése után a vezérlés ellenőrzi a "pozícióban" jelet, ha a paramétermező *POSCHECK* paramétere 1, és nem végezi, ha a paraméter állása 0. A "pozícióban" jelre 5 másodpercig vár, ha ezután sem érkezik meg a jel *1020 POZÍCIÓ HIBA* jelzést ad a vezérlő. A pozíciótól mért legnagyobb, még elfogadható eltérést az *INPOS* paraméteren lehet megadni.

G00 öröklődő kód, addig érvényes, amíg egy másik, interpolációs parancs át nem írja. Bekapcsolás után a paramétermező *CODES* paramétercsoportjában meghatározott érték szerint G00, vagy G01 kód van érvényben.

4.2 Az egyenes interpoláció (G01)

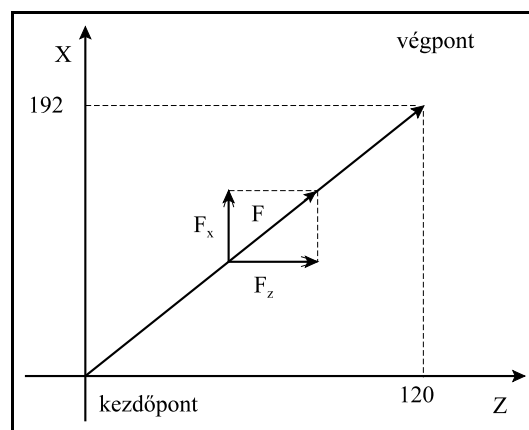
A

G01 v F

utasításkor lineáris interpolációs módot állít be. A v értékre írt adatok lehetnek abszolút illetve inkrementális értékek, és az aktuális koordinátarendszerben értelmezettek. A mozgás sebességét, az előtolást, F címen lehet programozni.

Az F címen programozott előtolás mindig a programozott pálya mentén érvényesül. Tengelymenti komponensei:

Előtolás az X tengely mentén:



4.2-1 ábra

$$F_x = \frac{X}{L} F$$

Előtolás az Z tengely mentén:

$$F_z = \frac{Z}{L} F$$

ahol x, z a megfelelő tengelyek mentén programozott elmozdulás értékek,
L a programozott elmozdulás hossza:

$$L = \sqrt{X^2 + Z^2}$$

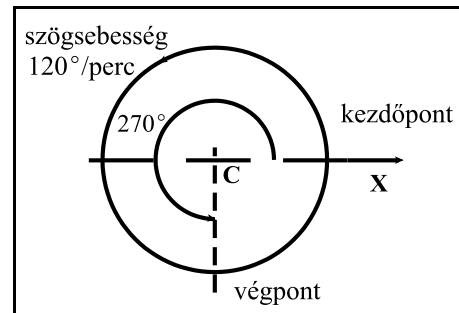
G01 X192 Z120 F0.15

Forgó tengely mentén az előtolás °/perc dimenzióban értelmezett:

G01 C270 F120

mondatban F120 jelentése: 120 °/perc.

Abban az esetben, ha egy hossz- és egy forgó tengely mozgását kapcsoljuk össze lineáris interpolációval az előtoláskomponensek szétosztása a fenti képletek alapján megy végbe.



4.2-2 ábra

Például:

G91 G01 Z100 C45 F120

mondatban a Z illetve C irányú előtoláskomponensek:

Előtolás a Z tengely mentén: $F_z = \frac{100}{\sqrt{100^2 + 45^2}} 120 = 109.4$ mm/perc

Előtolás a C tengely mentén: $F_c = \frac{45}{\sqrt{100^2 + 45^2}} 120 = 49.2$ °/perc

G01 öröklődő kód, addig érvényes, amíg egy másik, interpolációs parancs át nem írja. Bekapcsolás után a paramétermező CODES paramétercsoportjában meghatározott érték szerint G00, vagy G01 kód van érvényben.

4.3 A kör-, és a síkbeli spirális interpoláció (G02, G03)

A

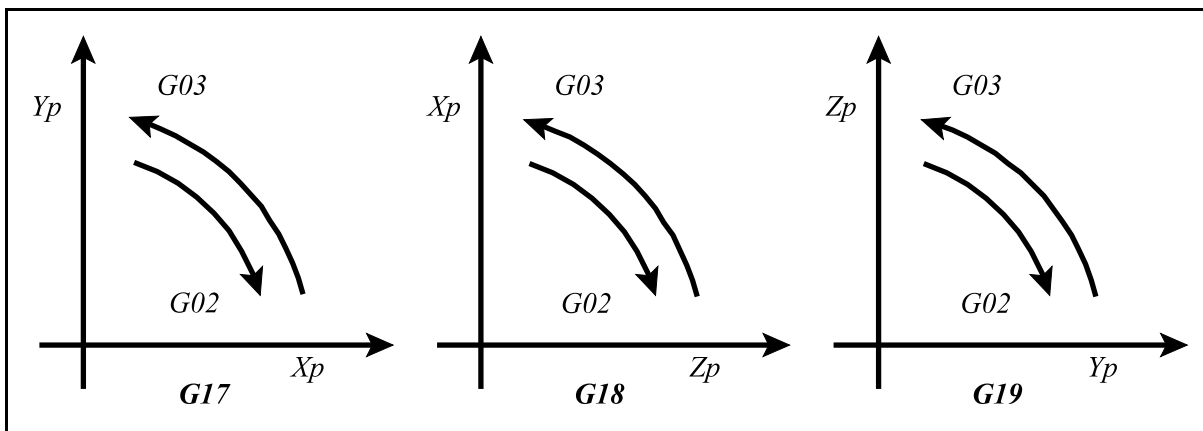
$$G17 \left\{ \begin{matrix} G02 \\ G03 \end{matrix} \right\} X_p Y_p \left\{ \begin{matrix} R \\ I J \end{matrix} \right\} F$$

$$G18 \left\{ \begin{matrix} G02 \\ G03 \end{matrix} \right\} X_p Z_p \left\{ \begin{matrix} R \\ I K \end{matrix} \right\} F$$

$$G19 \left\{ \begin{matrix} G02 \\ G03 \end{matrix} \right\} Y_p Z_p \left\{ \begin{matrix} R \\ J K \end{matrix} \right\} F$$

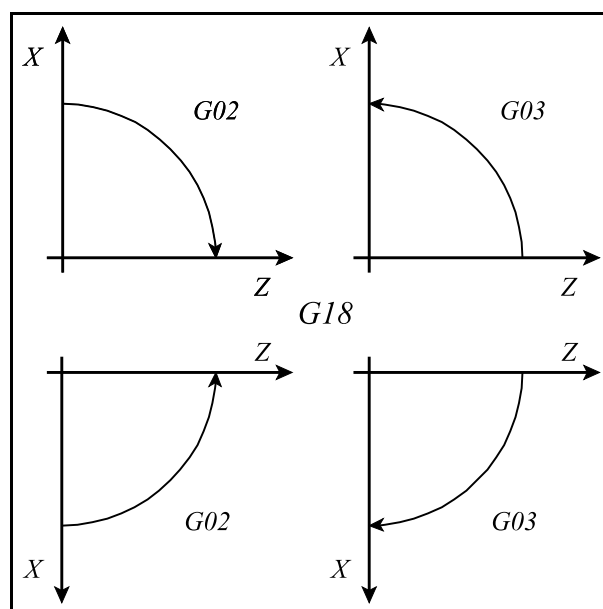
utasítássor körinterpolációt ír elő.

A körinterpoláció a G17, G18, G19 parancs által kiválasztott síkban megy végbe, G02 esetén az óramutató járásával megegyező, G03 esetén az óramutató járásával ellentétes irányban:



4.3-1 ábra

A fenti ábra G18 síkban a körirányokat jobbsodrású koordináta-rendszerben ábrázolja. Ha a szerszámtartó elrendezéséből adódóan az eszterga koordináta-rendszere balsodrású, a körirányok értelmezése megfordul.



4.3-2 ábra

X_p , Y_p , Z_p jelentése itt, és a továbbiakban:

X_p : X tengely, vagy azzal párhuzamos tengely,

Y_p : Y tengely, vagy azzal párhuzamos tengely,

Z_p : Z tengely, vagy azzal párhuzamos tengely.

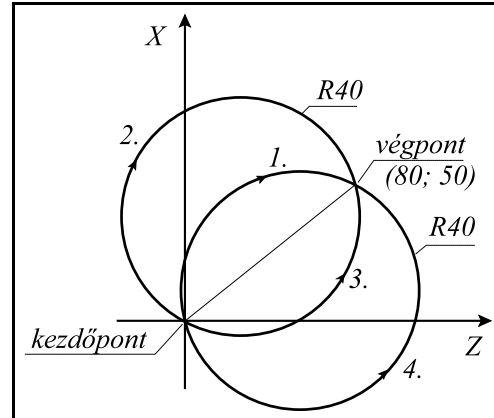
X_p , Y_p , Z_p értéke az adott koordináta-rendszerben a kör végpontjának koordinátája abszolút, vagy inkrementális adatként megadva.

Ha U, V, W cím nincs tengelynek kijelölve, akkor a kör végpontjának inkrementális koordinátáit jelölik X, Y, Z irányban.

A kör további adatainak megadása kétféleképp történhet:

1. eset:

R címen, ahol R a kör sugara. Ekkor a vezérlés a kezdő pont koordinátáiból (az a pont ahol a vezérlés a körmondattal beolvasásának pillanatában tartózkodik), a végpont koordinátáiból (X_p, Y_p, Z_p címen definiált érték), valamint a programozott R körsugárból automatikusan kiszámítja a kör középpont koordinátáit. Mivel egy adott körüljárási irány esetén (G02, vagy G03) a kezdő és végpont között két különböző, R sugarú kör húzható, ha a kör sugarát pozitív számmal adjuk meg a vezérlés a 180° -nál kisebb ív mentén halad, ha R-en negatív számot adunk meg a 180° -nál nagyobb ívet járja be. Például:



4.3-3 ábra

1. ívszakasz: G02 X80 Z50 R40
2. ívszakasz: G02 X80 Z50 R-40
3. ívszakasz: G03 X80 Z50 R40
4. ívszakasz: G03 X80 Z50 R-40

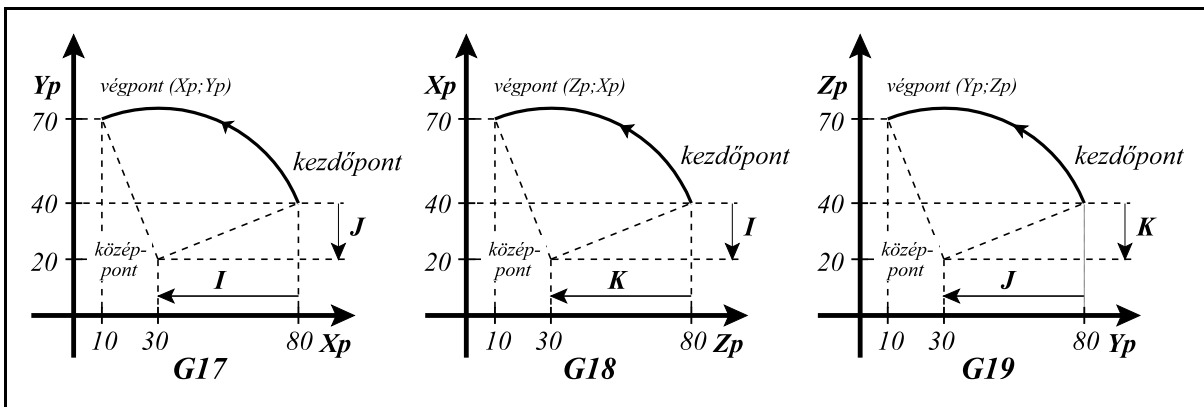
2. eset:

A kör középpontját I, J, K címen adjuk meg, az X_p, Y_p, Z_p tengelyekre. Az I, J, K címeiken megadott értékeket mindig inkrementálisan értelmezi a vezérlő, úgy, hogy az I, J, K értékek által definiált vektor a kör kezdőpontjából a kör középpontjába mutat. I értékét mindig sugárban kell megadni még, ha az X koordináta átmérőre is van állítva. Például:

G17 esetén: G03 X10 Y70 I-50 J-20 (X rádiuszban programozva)

G18 esetén: G03 X70 Z10 I-20 K-50 (X rádiuszban programozva)

G19 esetén: G03 Y10 Z70 J-50 K-20



4.3-4 ábra

F címen a pályamenti előtolást programozhatjuk, amely a körérintő irányába mutat és állandó az egész pálya mentén.

☞ *Megjegyzések:*

– I0, J0, K0 elhagyható. Például:

G03 X0 Z100 I-100

– Ha X_p , Y_p , Z_p , mind elhagyásra kerül, vagy a végpont koordináta megegyezik a kezdőpont koordinátával:

a. ha a kör középpont koordinátákat programozzuk I, J, K címen: 360° -os ívű, teljes kört interpolál a vezérlő. Például:

G03 I-100,

b. ha az R sugarat programozzuk: a vezérlő 3012 **KÖRMEGADÁS R-REL HIBÁS** jelzést ad.

– Ha a körmondatt:

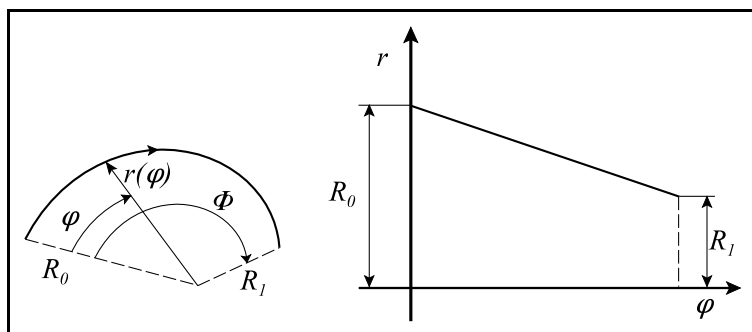
a. sem sugarat (R), sem I, J, K-t nem tartalmaz,

b. vagy a kiválasztott síkon kívüli I, J, K címre történik hivatkozás a vezérlés 3014 **KÖRMEGADÁS HIBÁS** jelzést ad. Például: G03 X0 Y100, vagy (G18) G02 X0 Z100 J-100.

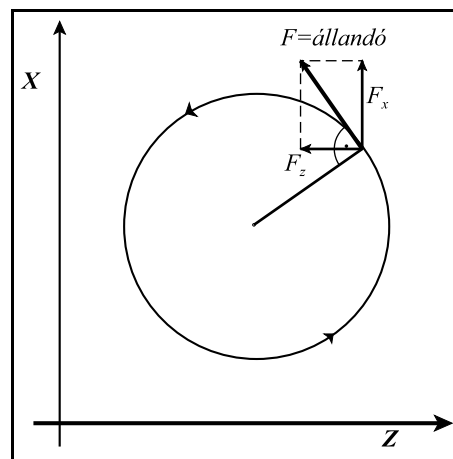
– Ha a G02, G03 mondatban meghatározott kör kezdőponti és végponti sugarának különbsége nagyobb, mint a *RADDIF* paraméteren meghatározott érték a vezérlés 3011 **SUGÁRKÜLÖNBSÉG HIBA KÖRBEN** jelzést ad.

Ha a sugarak különbsége kisebb a fenti paraméteren megadott értéknél a vezérlés a szerszámot olyan síkbeli spirális pálya mentén mozgatja, amelynél a sugár a központi szög függvényében lineárisan változik.

Változó sugarú körív interpolációjánál nem a pályamenti sebesség, hanem a szögsebesség lesz állandó.



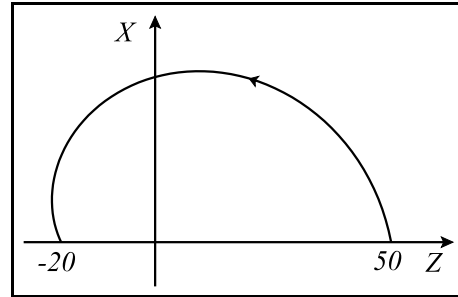
4.3-6 ábra



4.3-5 ábra

Az alábbi programrészlet arra mutat példát, hogyan lehet változó sugarú kört megadni I, K címek felhasználásával:

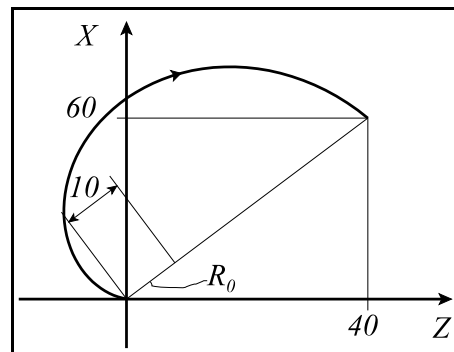
```
G90 G0 X0 Z50  
G3 Z-20 K-50
```



[4.3-7](#) ábra

Ha a megadott körsugár kisebb, mint a kezdőpontot a végponttal összekötő egyenes távolságának a fele, a vezérlő a megadott körsugarat tekinti a kör kezdőponti sugarának, és olyan változó sugarú kört interpolál, amelyik középpontja a kezdőpontot a végponttal összekötő egyenesen van, a kezdőponttól R távolságra:

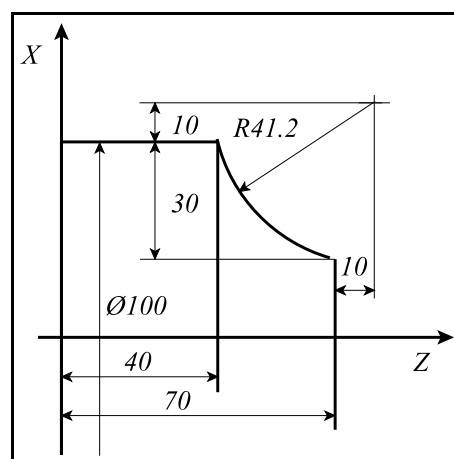
```
G0 G90 X0 Z0
G2 X60 Z40 R10
```



4.3-8 ábra

Az itt következő mintamondatokban az X koordináta átmérőben van kezelve, és feltételeztük, hogy U és W nincs tengelynek kijelölve:

```
G2 G90 X100 Z40 R41.2
vagy G2 G90 X100 Z40 I40 K10
vagy G2 G91 X60 Z30 R41.2
vagy G2 (G90) U60 W30 R41.2
vagy G2 (G90) XI60 ZI30 R41.2
vagy G2 G91 X60 Z30 I40 K10
vagy G2 (G90) U60 W30 I40 K10
vagy G2 (G90) XI60 ZI30 I40 K10
```



4.3-9 ábra

4.4 Egyenletes emelkedésű menet vágása (G33)

A

G33 v F Q

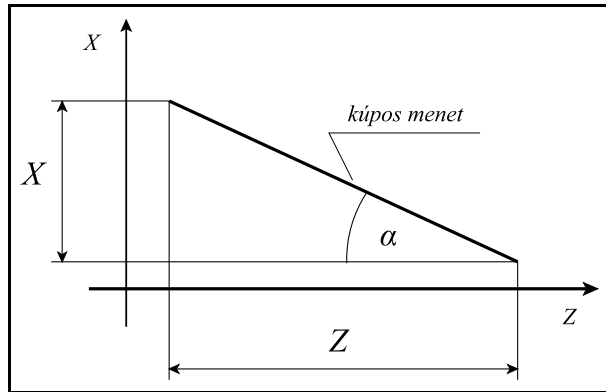
G33 v E Q

utasítás egyenletes emelkedésű hengeres, vagy kúpos menet vágását definiálja.

v vektorra maximum két tengely koordinátaadatát lehet beírni. Ha a v vektoron két koordináta adata van feltüntetve a vezérlés kúpos menetet vág. A menetemelkedést azon tengely mentén veszi figyelembe a vezérlő, amelyiken hosszabb elmozdulás adódik.

ha $\alpha < 45^\circ$, azaz $Z > X$ a programozott menetemelkedést a Z tengely mentén,

ha $\alpha > 45^\circ$, azaz $X > Z$ a programozott menetemelkedést az X tengely mentén veszi figyelembe.



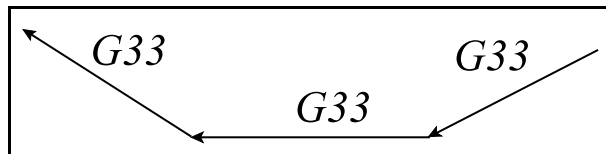
4.4-1 ábra

A menetemelkedést kétféleképp lehet definiálni:

- Ha a menetemelkedést F címen adjuk meg, akkor az adat értelmezése mm/ford, vagy inch/ford. Ha tehát egy 2.5 mm emelkedésű menetet akarunk vágni F2.5-öt kell programozni.
- Ha a menetemelkedést E címen adjuk meg a vezérlés inches menetet vág. E cím értelmezése inchenkénti menetszám. Ha például E3-at programozunk, akkor a vezérlő $\frac{1}{3}'' = 25.4/3 = 8.4667$ mm emelkedésű menetet vág.

Q címen adjuk meg azt a szögértéket, hogy a főorsó jeladó nullimpulzusától számítva hány fokot forduljon el a főorsó, mielőtt elkezd a menetet vágni. Több-bekezdésű menetet a Q érték megfelelő programozásával lehet vágni, vagyis itt lehet programozni, hogy a különböző bekezdéseket milyen főorsó szögelfordulás alatt kezdje el vágni a vezérlő. Például, ha egy kétbekezdésű menetet akarunk vágni, az első bekezdést Q0-ról indítjuk (külön programozni nem kell), a második bekezdést pedig Q180-ról.

A G33 öröklődő funkció. Ha egymás után több menetvágó mondatot programozunk, tetszőleges, egyenes szakaszokkal határolt felületre vághatunk menetet:



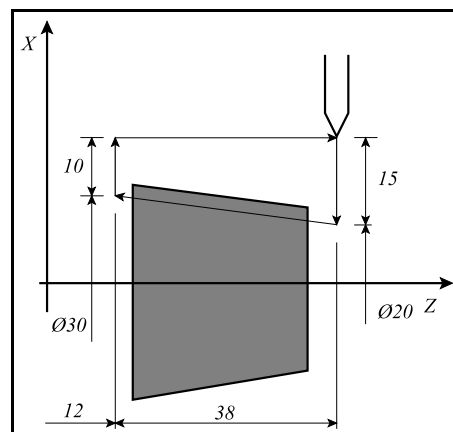
4.4-2 ábra

A vezérlés a főorsó jeladó nullimpulzusára az első mondatban szinkronozódik rá, és a további mondatoknál már nem végez szinkronizációt, következésképp a menet emelkedése folyamatos lesz az összes szakaszon. Ebből adódóan a programozott Q főorsó szögelfordulást is csak az első mondatban veszi figyelembe.

Példa menetvágás programozására:

```
G0 G90 X50 Z40
U-30
G33 U10 W38 F2
G0 U20
W-38
```

A példában X-ben átmérő programozást vettünk figyelembe.



4.4-3 ábra

☞ *Megjegyzések:*

- Ha a menetvágó mondatban két koordinátánál több van egyszerre kitöltve, vagy F és E cím is ki van töltve a vezérlő 3020 ADATMEGADÁSI HIBA hibajelzést ad.
- Ha a menetvágó mondatban E-re 0-t adunk meg, 3022 0-VAL OSZTÁS G33 hiba keletkezik.
- G33 parancs végrehajtásához jeladót kell szerelni a főorsóra.
- G33 parancs végrehajtása során az előtolás és főorsó override értékeket automatikusan 100%-nak veszi a vezérlés és az előtolás állj billentyű hatása csak a mondat végrehajtása után érvényesül.
- A szervorendszer követési hibája miatt a menet elején és végén ráfutási és kifutási távolságot kell hagyni az anyagon kívül a szerszámnak, hogy a menetemelkedés állandó legyen a teljes szakaszon.
- A menetvágás során az előtolás nagysága (mm/perc egységben) nem haladhatja meg a paramétermezőben tengelyenként beállítható FEEDMAX_n értékét.
- A menetvágás során a főorsó fordulatszám értéke nem haladhatja meg a főorsó jeladóra mechanikusan engedélyezett maximális fordulatszám, illetve a jeladó határfrekvenciájából (a jeladó által leadható maximális frekvenciából) visszszámolt fordulatszám értékek közül a kisebbiket.

4.5 Változó emelkedésű menet vágása (G34)

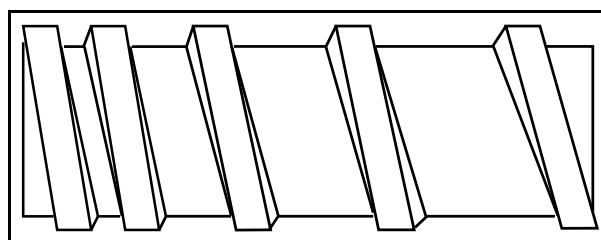
A

G34 v F Q K

utasítás változó emelkedésű hengeres, vagy kúpos menet vágását definiálja. A v, F, Q beemenő adatok értelmezése megegyezik a G33 funkcionál leírtakkal. K értelmezése:

K: a menetemelkedés főorsó fordulatonkénti növekedése, vagy csökkenése.

K értéke 0.001 mm/ford-tól (0.0001 inch/ford-tól) 500 mm/ford-ig (10 inch/ford-ig) terjedhet.



4.5-1 ábra

4.6 A polárkoordináta interpoláció (G12.1, G13.1)

A polárkoordináta interpoláció a vezérlésnek egy olyan működési módja, amelyben a derékszögű (Descartes) koordinátarendszerben leírt munkadarab kontúr pályáját egy lineáris és egy forgó tengely mozgásával járja le.

A

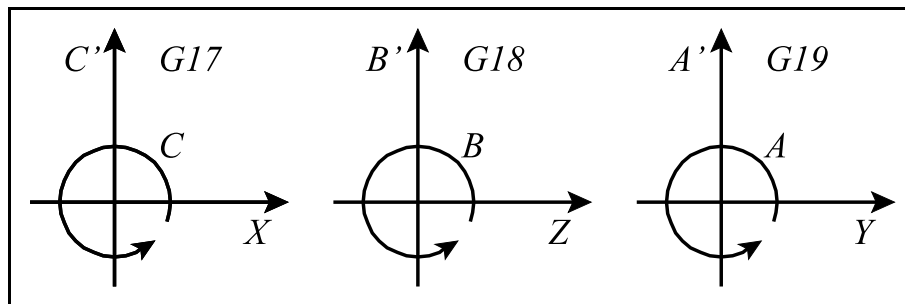
G12.1 polárkoordináta interpoláció be
 utasítás bekapcsolja a polárkoordináta üzemet. Az ezután következő programrészben a marószerszám pályáját derékszögű koordinátarendszerben, a hagyományos módon, egyenes és körinterpoláció programozásával írhatjuk le, a szerszámsugár korrekció figyelembe vételével. *Az utasítást mindig külön mondatban adjuk meg, és más utasítást nem programozhatunk mellé.*

A

G13.1 polárkoordináta interpoláció ki
 utasítás kikapcsolja a polárkoordináta üzemet. *Az utasítást mindig külön mondatban adjuk meg, és más utasítást nem programozhatunk mellé.* A vezérlés bekapcsolás, vagy a reset után mindig G13.1 állapotot vesz fel.

Síkválasztás

A polárkoordináta interpoláció bekapcsolása előtt ki kell választani egy síkot, amely megadja az alkalmazni kívánt lineáris és forgó tengely címét.



4.6-1 ábra

A

G17 X_ C_
 utasítás az X tengelyt jelöli ki lineáris, a C-t forgó tengelynek. Az ábrán C'-vel jelöltük a virtuális tengelyt, amelynek a programozása hosszmeretek megadásával történik.

A

G18 Z_ B_
 G19 Y_ A_

utasításokkal a fent leírt módon jelölhetjük össze a további lineáris és forgó tengelyeket.

A munkadarab nullpont helyzete a polárkoordináta interpoláció során

Polárkoordináta interpoláció használata esetén az alkalmazott koordinátarendszer nullpontját a lineáris tengelyen (X) kötelezően úgy kell megválasztani, hogy a forgó szerszám forgástengelye X=0 pozícióban egybeesik a körtengely (C) forgástengelyével.

Hosszkorrekció használata a polárkoordináta interpoláció során

A polárkoordináta interpolációban használt szerszám X irányú hosszkorrekciója kötelezően 0 kell legyen, feltéve, hogy az interpolációban az X és C tengelyek vesznek részt. Az X irányú kompenzálást munkadarab nullponttolással kell beállítani (lásd az előző pontot).

A tengelyek helyzete a polárkoordináta interpoláció bekapcsolása pillanatában

A polárkoordináta interpoláció bekapcsolása előtt (G12.1 utasítás) gondoskodni kell arról, hogy a **körtengely a 0 pozíciój**ú pontban legyen. A **lineáris tengely pozíciója** lehet negatív és pozitív is, de **nem lehet 0**.

A hosszadatok programozása a polárkoordináta interpoláció során

A polárkoordináta interpoláció bekapcsolt állapotában a kiválasztott síkhoz tartozó mindkét tengelyen hosszadatokat programozunk: a kiválasztott síkban szereplő forgó tengely lesz a második (virtuális) tengely. Ha pl. a G17 X_ C_ utasítással az X, C tengelyt választottuk ki, a C címet úgy programozhatjuk, mint a G17 X_ Y_ síkválasztás esetén az Y-t.

A virtuális tengely programozását nem befolyásolja, hogy az első tengely programozása átmérőben történik-e, a **virtuális tengelyen** mindig **sugárban** kell megadni a koordinátaadatokat. Ha pl. a polárkoordináta interpoláció az X C síkban történik, függetlenül attól, hogy az X címet átmérőben, vagy sugárban adjuk meg, a C címre írt értéket sugárban kell megadni.

A polárkoordináta interpolációban részt nem vevő tengelyek mozgása

A szerszám ezeken a tengelyeken, a polárkoordináta interpoláció bekapcsolt állapotától függetlenül, úgy mozog, mint normális esetben.

Körinterpoláció programozása a polárkoordináta interpoláció során

A polárkoordináta interpoláció bekapcsolt állapotában kör megadása a már ismert módon, sugárral, vagy kör középponti koordináta programozásával lehetséges. Ha ez utóbbit választjuk I, J, K címeiket a kiválasztott síknak megfelelően, az alábbiak szerint kell használni:

G17 X_ C_	G18 Z_ B_	G19 Y_ A_
G12.1	G12.1	G12.1
...
G2 (G3) X_ C_ I_ J_	G2 (G3) B_ Z_ I_ K_	G2 (G3) Y_ A_ J_ K_

Szerszámsugár korrekció használata polárkoordináta interpoláció esetén

G41, G42 utasítás a megszokott módon használható a polárkoordináta interpoláció bekapcsolt állapotában. Ügyeljünk arra, hogy a forgó szerszám korrekciós csoportjában a szerszámállás kódja Q=0 kell legyen. Használatára az alábbi megszorítások vonatkoznak:

- A polárkoordináta interpoláció bekapcsolása (G12.1 utasítás) csak G40 állapotban lehetséges,
- Ha G12.1 állapotban bekapcsoltuk G41, vagy G42-t, a polárkoordináta interpoláció kikapcsolása (G13.1 utasítás) előtt G40-et kell programozni.

Programozási megszorítások a polárkoordináta interpoláció során

A polárkoordináta interpoláció bekapcsolt állapotában az alábbi utasítások nem használhatók:

- síkváltás: G17, G18, G19,
- koordinátatranszformációk: G52, G92,
- munkadarab koordinátarendszer váltás: G54, ..., G59,
- pozícionálás a gépi koordinátarendszerben: G53.

Az előtolás a polárkoordináta interpoláció során

Az előtolás értelmezése a polárkoordináta interpoláció bekapcsolt állapotában a derékszögű interpolációnál megszokott módon, pályamenti sebességként történik: a munkadarab és a szerszám relatív sebességét adja meg.

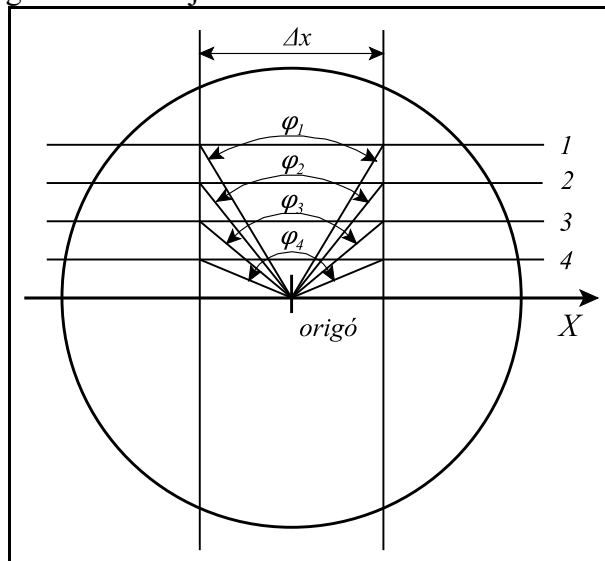
A polárkoordináta interpoláció során egy derékszögű koordinátarendszerben megadott pályát egy lineáris és egy forgó tengely mozgásával járja le. Ahogy a szerszám középpontja közeledik a körkoordináta forgástengelyéhez, úgy kellene a forgó tengelynek időegység alatt mind nagyobb

és nagyobbat lépnie ahhoz, hogy a pályamenti sebesség állandó legyen. A körtengely sebességnek viszont határt szab a forgó tengelyre megengedhető maximális sebesség, amit paraméter határoz meg. Ezért az origó közelében a vezérlés fokozatosan csökkenti a pályamenti előtolást, annak érdekében, hogy a forgó tengely sebessége ne növekedjék minden határon túl.

A mellékelt ábra azt az esetet mutatja, amikor az X tengellyel párhuzamos egyeneseket (1, 2, 3, 4) programozunk. A programozott előtoláshoz időegység alatt Δx elmozdulás tartozik. A Δx elmozduláshoz a különböző egyenesek (1, 2, 3, 4) esetén más és más szögelfordulás ($\varphi_1, \varphi_2, \varphi_3, \varphi_4$) tartozik. Látszik, hogy minél közelebb jár a megmunkálás az origóhoz, annál nagyobb szögelfordulást kell a forgó tengelynek időegység alatt megtennie, hogy a programozott előtolást tartani tudja.

Ha az időegység alatt megteendő szögelfordulás meghaladja a forgó tengelyre beállított FEEDMAX paraméter értékét a vezérlés fokozatosan csökkenti a pályamenti előtolást.

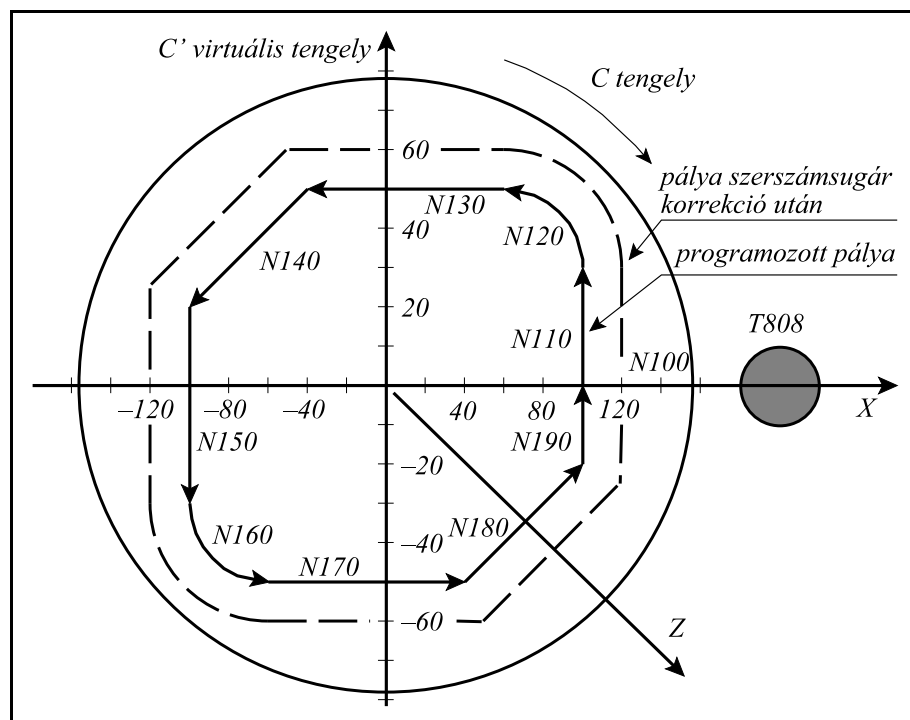
A fent elmondottak alapján kerüljük olyan programok írását, amelyeknél a szerszám középpontja az origó közelében halad.



4.6-2 ábra

Mintapélda

Az alábbiakban közlünk a polárkoordináta interpoláció használatára egy mintapéldát. Az interpolációban részt vevő tengelyek: X (lineáris tengely) és C (forgó tengely). X tengely programozása átmérőben, C tengelyé sugárban történik.



4.6-3 ábra

%O7500 (POLARKOORDINATA INTERPOLACIO)

...

N050 T808

N060 G59

(G59 koordinátarendszer kezdőpontja X irányban
C forgástengelye)

N070 G17 G0 X200 C0

(síkválasztás X, C; pozicionálás X≠0, C=0 koor-
dinátára)

N080 G94 Z-3 S1000 M3

N090 G12.1

(polárkoordináta interpoláció be)

N100 G42 G1 X100 F1000

N110 C30

N120 G3 X60 C50 I-20 J0

N130 G1 X-40

N140 X-100 C20

N150 C-30

N160 G3 X-60 C-50 R20

N170 G1 X40

N180 X100 C-20

N190 C0

N200 G40 G0 X150

N210 G13.1

(polárkoordináta interpoláció ki)

N220 G0 G18 Z100

(szerszám visszahúzása X, Z síkválasztás)

...

%

4.7 A hengerinterpoláció (G7.1)

Ha egy henger palástjára vezérpályát kell marni hengerinterpolációt alkalmazunk. Ilyenkor a henger és egy forgó tengely forgástengelyének egybe kell esnie. A programban a forgó tengely elmozdulását fokban adjuk meg, amit a vezérlő átszámít lineáris elmozdulássá a palást mentén a henger sugarának függvényében úgy, hogy lineáris és körinterpolációt lehessen programozni egy másik, lineáris tengellyel együtt. Az interpoláció után kiadódó elmozdulást visszaalakítja a forgó tengely számára szögelfordulássá.

A

G7.1 Qr hengerinterpoláció be
utasítás bekapcsolja a hengerinterpolációt, ahol

Q: a hengerinterpolációban részt vevő forgó tengely címe

r: a henger sugara.

Ha például a hengerinterpolációban részt vevő forgó tengely a C tengely, és a henger sugara 50 mm, akkor a hengerinterpolációt a G7.1 C50 utasítással kapcsolhatjuk be.

Az ezután következő programrészben egyenes és körinterpoláció megadásával írhatjuk le a henger palástjára marandó pályát. A hossz tengelyen a koordinátát mindig mm-ben, vagy inch-ben, a forgó tengelyen pedig °-ban adjuk meg.

A

G7.1 Q0 hengerinterpoláció ki
utasítás kikapcsolja a hengerinterpolációt, vagyis a G kód ugyanaz, mint a bekapcsolásé, csak a forgó tengely címére 0-t kell írni.

A fenti példával (G7.1 C50) bekapcsolt hengerinterpolációt a G7.1 C0 utasítással kapcsolhatjuk ki.

A G7.1 utasítást külön mondatban kell megadni.

Síkválasztás

A síkválasztás kódját mindig annak a lineáris tengelynek a neve határozza meg, amellyel a forgó tengely párhuzamos. Az X tengellyel párhuzamos tengelyű forgótengely az A, az Y-nal a B, a Z-vel a C:

G17 X A, vagy	G18 Z C, vagy	G19 Y B, vagy
G17 B Y	G18 A X	G19 C Z

Körinterpoláció

Hengerinterpolációs üzemmódban körinterpoláció megadása lehetséges, azonban csak a sugár R megadásával.

Körinterpoláció a kör középpontjának megadásával (I, J, K) nem lehetséges hengerinterpoláció esetén.

A kör sugara mindig mm-ben, vagy inch-ben kerül értelmezésre, soha nem fokban.

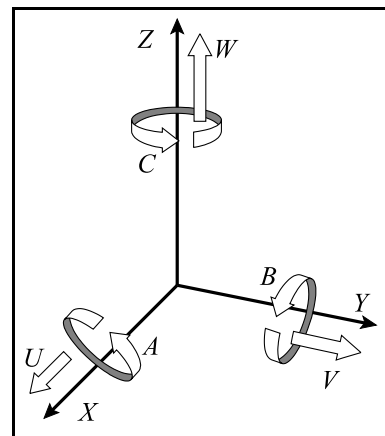
Körinterpolációt például a Z és a C tengely között kétféleképp adhatunk meg:

G18 Z_ C_	G19 C_ Z_
G2 (G3) Z_ C_ R_	G2 (G3) C_ Z_ R_

Szerszámsugár korrekció használata hengerinterpoláció esetén

G41, G42 utasítás a megszokott módon használható a hengerinterpoláció bekapcsolt állapotában. Használatára az alábbi megszorítások vonatkoznak:

– A hengerinterpoláció bekapcsolása (G7.1 Qr utasítás) csak G40 állapotban lehetséges,



4.7-1 ábra

- Ha a hengerinterpoláció állapotában bekapcsoltuk a G41, vagy G42-t, a hengerinterpoláció ki-kapcsolása (G7.1 Q0 utasítás) előtt G40-et kell programozni.

Programozási megkorlátások a hengerinterpoláció során

A hengerinterpoláció bekapcsolt állapotában az alábbi utasítások nem használhatók:

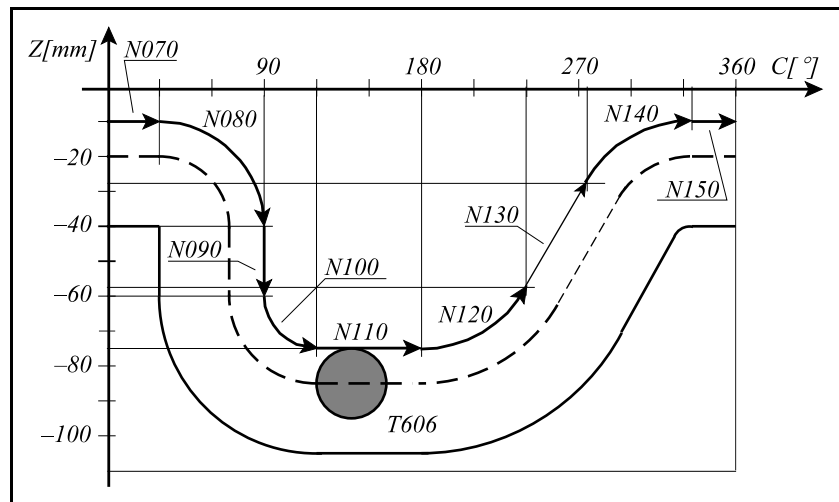
- síkváltás: G17, G18, G19,
- koordinátatranszformációk: G52, G92,
- munkadarab koordináta-rendszer váltás: G54, ..., G59,
- pozicionálás a gépi koordináta-rendszerben: G53,
- körinterpoláció a kör középpontjának (I, J, K) megadásával,
- fűróciklusok.

Mintapélda

Marjunk egy $R=28.65$ mm sugarú henger palástjára 3 mm mélyen, egy, a mellékelt ábrán látható pályát. A T606 forgó szerszám párhuzamos az X tengellyel. A henger palástján az egy fokra (1°) eső elmozdulás:

$$28.65 \text{ mm} \cdot \frac{1^\circ}{180^\circ} \cdot \pi = 0.5 \text{ mm}$$

Az ábrán látható tengelyelrendezés G19 síkváltásznak felel meg.



4.7-2 ábra

```
%O7602 (HENGERINTERPOLACIO)
```

```
...
```

```
N020 G0 X200 Z20 S500 M3 T606
```

```
N030 G19 Z-20 C0
```

(G19: C-Z sík váltása)

```
N040 G1 X51.3 F100
```

```
N050 G7.1 C28.65
```

(hengerinterpoláció bekapcsolása, a forgó tengely: C, a henger sugara 28.65mm)

```
N060 G1 G42 Z-10 F250
```

```
N070 C30
```

```
N080 G2 Z-40 C90 R30
```

```
N090 G1 Z-60
```

```
N100 G3 Z-75 C120 R15
```

```
N110 G1 C180
```

```
N120 G3 Z-57.5 C240 R35
```

```
N130 G1 Z-27.5 C275
```

```
N140 G2 Z-10 C335 R35
```

```
N150 G1 C360
```

```
N160 G40 Z-20
```

```
N170 G7.1 C0
```

(hengerinterpoláció kikapcsolása)

```
N180 G0 X100
```

```
...
```

```
%
```

5 A koordinátaadatok

5.1 Abszolút és inkrementális programozás (G90, G91), az I operátor

A bemenő koordinátaadatok megadhatók abszolút és növekményes értéként is. Abszolút adatmegadásnál a végpont koordinátáit kell a vezérlésnek megadni, míg növekményes adatnál a mondatban végrehajtandó megteendő távolságot.

G90: Abszolút adatmegadás programozása

G91: Növekményes adatmegadás programozása

A G90, G91 öröklődő funkciók. Bekapcsolásra a *CODES* paraméter alapján dönthető el melyik állapotot vegye fel a vezérlő.

Abszolút pozícióra való mozgás csak referenciapontfelvétel után lehetséges.

Példa:

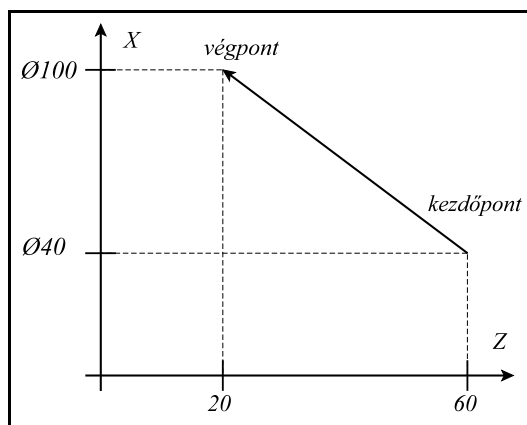
Az ábra alapján kétféleképp lehet a mozgást programozni:

```
G90 G01 X100 Z20
G91 G01 X60 Z-40
```

Az I operátor G90 abszolút adatmegadási állapotban határos. Csak arra a koordinátára vonatkozik, amelyik címe után áll. Jelentése: inkrementális adat.

A fenti példa a következőképp is megoldható:

```
(G90) G01 XI60 ZI-40
G01 XI60 Z20
G01 X100 ZI-40
```



5.1-1 ábra

Ha az U, V, W címek nincsenek tengelynek kijelölve akkor felhasználhatók az X, Y, Z irányú inkrementális mozgások jelölésére:

	abszolút parancs címe	inkrementális parancs címe
mozgás parancs X irányban	X	U
mozgás parancs Z irányban	Z	W
mozgás parancs C irányban	C	H
mozgás parancs Y irányban	Y	V

A fentiek figyelembe vételével a mintapélda:

```
(G90) G01 U60 W-40
G01 U60 Z20
G01 X100 W-40
```

5.2 Inch/Metrikus átalakítás (G20, G21)

A bemenő adatokat megadhatjuk akár metrikus akár inches mértékrendszerben, a megfelelő G kód programozásával.

G20: Inches mértékrendszer választása.

G21: Metrikus mértékrendszer választása.

A program elején a megfelelő kód megadásával ki kell választani a kívánt mértékrendszert. A kiválasztott mértékrendszer mindaddig érvényben marad amíg ellenkező értelmű parancsot nem adunk ki, tehát G20, G21 öröklődő kódok. A kódok hatása kikapcsolásra is megőrződik, tehát bekapcsolás után a kikapcsoláskori mértékrendszer lesz érvényben.

A G20/G21 kódoknak nincs hatása a szövegységben mért tengelyekre, a szög mértékegysége mindkét esetben fok.

A G20/G21 parancsot mindig önálló mondatba programozzuk egyedülálló utasításként, és más címekre ne írjunk parancsot, mert azokat a vezérlő nem fogja végrehajtani.

A következő tételekre van hatással a mértékrendszer megváltoztatása:

- Koordináta- és korrekcióadatok,
- Előtolás,
- Konstans vágósebesség,
- Pozíció-, korrekció- és előtoláskijelzés.

5.3 Koordinátaadatok megadása és értékhatára

Koordinátaadatok 8 decimális számjegyen lehet megadni.

Az alkalmazott mértékrendszer függvényében értelmezi a tizedespontot:

- X2.134 jelentése 2.134 mm, vagy 2.134 inch,
- B24.36 jelentése 24.36 fok, ha B címen szögadatot tárolunk.

A tizedespont használata nem kötelező:

- X325 jelentése például 325 mm.

A vezető nullák elhagyhatók:

- .032=0.032

Tizedespont után a követő nullák elhagyhatók:

- 0.320=.32

Az alkalmazott inkremensrendszer által meghatározott értéknél kisebb számot is értelmez a vezérlő. Például: X1.23456 parancsot, ha IR-B a beállított inkremensrendszer:

- metrikus mértékrendszer esetén 1.235 mm-ként értelmezi,
- inches mértékrendszer esetén pedig 1.2346 inch-ként.

Tehát a bemenő adatok kerekített értéként kerülnek kiadásra.

A hosszkoordináták értékhatárát az alábbi táblázat mutatja:

bemeneti mérték-rendszer	kimeneti mérték-rendszer	inkremens-rendszer	hosszkoordináták értékhatára	dimenzió
mm	mm	IR-A	$\pm 0.01-999999.99$	mm
		IR-B	$\pm 0.001-99999.999$	
		IR-C	$\pm 0.0001-9999.9999$	
inch	mm	IR-A	$\pm 0.001-39370.078$	inch
		IR-B	$\pm 0.0001-3937.0078$	
		IR-C	$\pm 0.00001-393.70078$	
inch	inch	IR-A	$\pm 0.001-99999.999$	inch
		IR-B	$\pm 0.0001-9999.9999$	
		IR-C	$\pm 0.00001-999.99999$	
mm	inch	IR-A	$\pm 0.01-999999.99$	mm
		IR-B	$\pm 0.001-99999.999$	
		IR-C	$\pm 0.0001-9999.9999$	

A szögkoordináták értékhatára:

inkremensrendszer	szögkoordináták értékhatára	dimenzió
IR-A	$\pm 0.01-999999.99$	fok
IR-B	$\pm 0.001-99999.999$	
IR-C	$\pm 0.0001-9999.9999$	

5.4 Átmérőben, vagy sugárban történő programozás

Mivel az esztergákon megmunkált munkadarabok metszete általában kör, az X tengely irányú méreteket meg lehet adni átmérőben is. Azt, hogy az X irányú méretet átmérőben, vagy rádiuszban értelmezze a vezérlés a következő paraméteren adhatjuk meg:

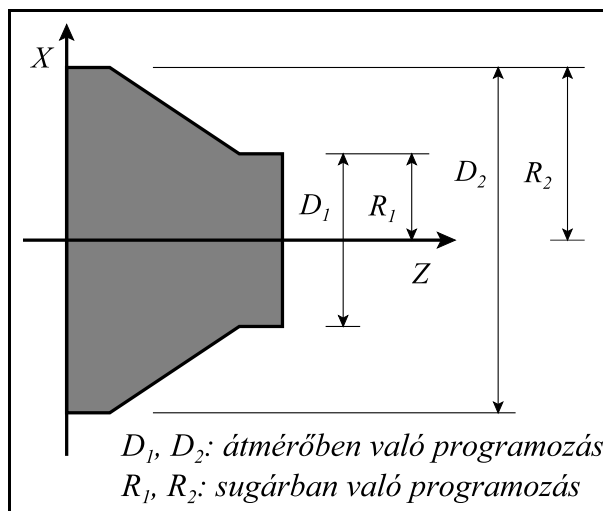
rádiuszban történő programozás esetén:

4762 DIAM=0

átmérőben történő programozás esetén:

4762 DIAM=1

Átmérőben történő programozás esetén a legkisebb beadható méret 1 inkremens. Ekkor sugárban számítva 1/2 inkremenst lép. Pl. INCRSYSTB esetén a legkisebb beadható méret 0.001 mm, és 0.0005 mm-t lép rádiuszban a vezérlő.



5.4-1 ábra

Ha a paramétert átmérő programozására állítottuk a következőket vegyük figyelembe:

eset	megjegyzés
X irányú abszolút mozgásparancs	átmérőben adjuk meg
X irányú inkrementális mozgásparancs	átmérőben adjuk meg (ábránkon $D_2 - D_1$)
koordináta és nullpont eltolás X irányban	átmérőben adjuk meg
X irányú szerszámhossz korrekció	átmérőben adjuk meg
X tengelyre vonatkozó paraméterek ciklusokban, mint pl. fogásmélység	mindig sugárban adjuk meg
körinterpoláció megadásánál R és I értéke	mindig sugárban adjuk meg
X tengely pozíció kijelzése	átmérőben történik
X irányú eltolás oldalazásnál	mindig sugár/ford, vagy sugár/perc
lépésméret léptetés és kézikerek üzemmódban	1 inkremens választása esetén 1 μ m-t lép átmérőben

5.5 Forgó tengelyek átfordulás kezelése

Ez a funkció forgó tengelyek esetén használható, tehát, ha az A, a B, vagy a C cím forgó tengely kezelésére van kijelölve. Átfordulás kezelésen azt értjük, hogy az adott tengelyen a pozíciót nem plusz minusz végtelen között tartjuk nyilván, hanem a tengely periódikusságát figyelembe véve, pl.: 0° és 360° között.

Tengely kijelölése forgó tengelynek

Ezt a kijelölést az A tengely esetén a 0182 A.ROTARY, a B tengely esetén a 0185 B.ROTARY, a C tengely esetén pedig a 0188 C.ROTARY paraméter 1-be való írásával végezhetjük el. Ha ezek közül a paraméterek közül valamelyik 1

- a szóban forgó tengelyre a vezérlés nem végzi el az inch/metrikus konverziót,
- a szóban forgó tengelyen engedélyezhető a megfelelő ROLLOVEN paraméter 1-be írásával az átfordulás kezelés.

Az átfordulási funkció engedélyezése

A funkciót az A tengelyre a 0241 ROLLOVEN_A, a B tengelyre a 0242 ROLLOVEN_B, a C tengelyre a 0243 ROLLOVEN_C paraméter 1-be írása élesíti, feltéve, hogy a megfelelő tengelyt forgó tengelynek jelöltük ki. Ha a megfelelő ROLLOVEN_x paraméter

- =0: a forgó tengelyt úgy kezeli, mint a lineáris tengelyeket, és a további paraméterek kitöltése hatástalan,
- =1: a forgó tengelyre alkalmazza az átfordulás kezelést, amelynek lényegét az alábbiakban leírtak határozzák meg.

Az egy fordulatra eső út megadása

A 0261 ROLLAMNT_A paraméteren adjuk meg az A tengelyre, a 0262 ROLLAMNT_B paraméteren a B tengelyre, valamint a 0263 ROLLAMNT_C paraméteren a C tengelyre a tengely egy körülfordulására eső utat bemeneti inkremensben. Tehát, ha a vezérlés a B inkremensrendszerben dolgozik, és a tengely egy körülfordulására 360° -ot forog, a megfelelő ROLLAMNT paraméterre írandó érték: 360000.

A fenti paraméterbeállításokkal a forgó tengely pozícióját mindig a $0^\circ - +359.999^\circ$ közötti tarto-

mányban jelzi ki a vezérlő, függetlenül attól, hogy melyik irányban forgott, és hány fordulatot tett meg a forgó tengely.

Forgó tengely mozgása abszolút programozás esetén

Ha a forgó tengelyre az átfordulás kezelés engedélyezve van (ROLLOVEN_x=1), abszolút adatmegadás esetén a tengely soha nem mozog a megfelelő ROLLAMNT_x paraméteren beállított elmozdulásnál többet. Vagyis, ha pl.: ROLLAMNT_C=360000 (360°), a legnagyobb elmozdulás 359.999°.

A 0244 ABSHORT_A, a 0245 ABSHORT_B és a 0246 ABSHORT_C paraméter alapján lehet beállítani, hogy a mozgás iránya mindig a tengelycímen megadott pozíció előjele szerint, vagy a rövidebb úton történjék. Ha a megfelelő ABSHORT_x paraméter

- =0: mindig a programozott pozíció előjelének irányában mozog, ha
- =1: mindig a rövidebb irányban mozog.

0188 C.ROTARY=1, 0243 ROLLOVEN_C=1 0263 ROLLAMNT_C= =360000	Abszolút koordinátamegadás- sal programozott mondat	A mondat hatására megtett elmozdulás	Pozíció a mondat végén
0246 ABSHORT_C=0 mindig a C címen programozott előjel szerinti irányban mozog			C=0
	G90 C450	90	C=90
	G90 C0 (a 0 pozitív szám!)	270	C=0
	G90 C-90	-90	C=270
	G90 C-360	-270	C=0
0246 ABSHORT_C=1 mindig a rövidebb úton mozog			C=0
	G90 C450	90	C=90
	G90 C0	-90	C=0
	G90 C-90	-90	C=270
	G90 C-360	90	C=0

Forgó tengely mozgása inkrementális programozás esetén

Inkrementális adatmegadás programozása esetén az elmozdulás iránya mindig a programozott előjel szerint történik.

A 0247 RELROUND_A paraméteren az A tengelyre, a 0248 RELROUND_B paraméteren a B tengelyre, a 0249 RELROUND_C paraméteren a C tengelyre lehet beállítani, hogy az elmozdulás nagyságára alkalmazza-e, vagy sem a megfelelő ROLLAMNT_x paramétert. Ha a megfelelő RELROUND_x paraméter:

- =0: nem alkalmazza a ROLLAMNT_x paramétert, tehát az elmozdulás nagyobb lehet, mint 360°,
- =1: alkalmazza a ROLLAMNT_x paramétert. Ha pl.: ROLLAMNT_C=360000 (360°), a legnagyobb elmozdulás 359.999° lehet a C tengelyen.

0188 C.ROTARY=1 , 0243 ROLLOVEN_C=1 0263 ROLLAMNT_C= =360000	Inkrementális koordinátamegadásal programozott mondat	A mondat hatására megtett elmozdulás	Pozíció a mondat végén
0249 RELROUND_C=0 nem alkalmazza a ROLLAMNT_C paramétert			C=0
	G91 C450	450	C=90
	G91 C0	0	C=90
	G91 C-90	-90	C=0
	G91 C-360	-360	C=0
0249 RELROUND_C=1 alkalmazza a ROLLAMNT_C paramétert			C=0
	G91 C450	90	C=90
	G91 C0	0	C=90
	G91 C-90	-90	C=0
	G91 C-360	0	C=0

6 Az előtolás

6.1 A gyorsmeneti előtolás

A pozicionálás gyorsmenettel történik G00 parancs hatására.

A gyorsmenet tengelyenkénti értékét a gép építője a paramétermezőben állítja be. A gyorsmenet nagysága tengelyenként különböző lehet.

Ha több tengely végez egyidejűleg gyorsmeneti mozgást, az eredő előtolás értékét a vezérlő úgy számítja ki, hogy a tengelyekre vetített sebességkomponens egyik tengelyen se haladja meg az arra a tengelyre érvényes, paraméterben megadott gyorsmeneti értéket, és a pozicionálás minimális idő alatt menjen végbe.

A gyorsmeneti előtolást módosítja a gyorsmeneti százalék kapcsoló, amely a következő állásokat vehet fel:

F0: a 1204 RAPOVER paraméterre írt érték %-ban értelmezve, illetve 25%, 50%, 100%.

100% fölé nem megy a gyorsmenet értéke.

A gyorsmeneti előtolást mindig leállítja az előtolás százalék kapcsoló 0% állása.

Érvényes referenciapont híján a paramétermezőben a gép építője által definiált csökkentett gyorsmeneti értékek lesznek érvényben tengelyenként, mindaddig, amíg a referenciapontfelvétel meg nem történt.

A fenti gyorsmeneti % értékeket veheti a vezérlés az előtolás százalék kapcsolóról is.

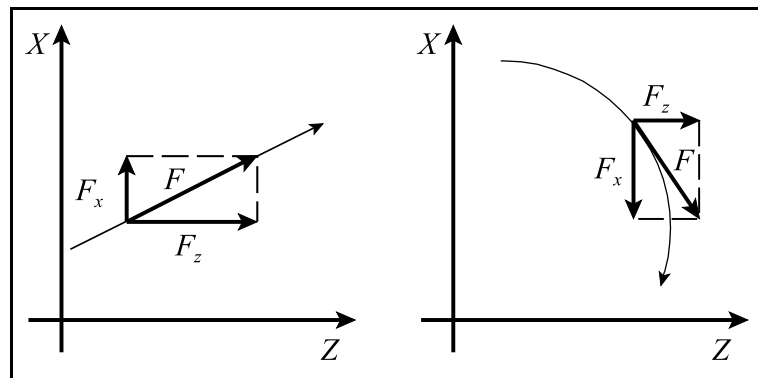
A tengelymozgató billentyűkkel végzett szánmozgatáskor a gyorsjárat sebesség a pozicionálási gyorsmenettől különböző, szintén a paramétermezőben beállított, tengelyenként különböző érték. Értelemszerűen a pozicionálási sebességnél kisebb érték, hogy a megálláshoz az emberi reakcióidőt is be lehessen kalkulálni.

6.2 A munkaelőtolás

Az előtolást **F** címen programozzuk.

A programozott előtolás lineáris (G01) és körinterpolációs (G02, G03) mondatokban érvényesül.

Az előtolás a programozott pálya mentén tangenciálisan érvényesül.



6.2-1 ábra

F : előtolás érintő irányú nagysága (programozott érték)

F_x : előtoláskomponens az X irányban

F_z : előtoláskomponens a Z irányban

$$F = \sqrt{F_x^2 + F_z^2}$$

A programozott előtolást az előtolásszázalék kapcsolóval a 0-120% tartományban lehet módosítani kivéve a G63, százalékkapcsoló és stop tiltás, állapotot.

Az F előtolás értéke öröklődő. Bekapcsolás után a paramétermező FEED paraméterén beállított előtolásérték kerül érvényesítésre.

6.2.1 Percenkénti (G94) és fordulatonkénti (G95) előtolás

Az előtolás mértékegységét a G94 és G95 kódokkal lehet megadni a programban:

G94: percenkénti előtolás

G95: fordulatonkénti előtolás

Percenkénti előtoláson a mm/perc, inch/perc, vagy fok/perc dimenzióban megadott előtolást értjük.

Fordulatonkénti előtoláson az egy főorsó fordulatra végzett előtolást értjük mm/ford, inch/ford, vagy fok/ford dimenzióban. Fordulatonkénti előtolást csak abban az esetben lehet programozni, ha a főorsóra jeladó van szerelve.

Öröklődő értékek. Bekapcsolás után a paramétermező *CODES* paramétercsoportja alapján kerül kiválasztásra a G94, vagy G95 állapot. A G94/G95 állapot nem befolyásolja a gyorsmeneti előtolást, az mindig percenkénti dimenzióban értendő.

Az alábbi táblázatok az F címen programozható abszolút maximális értékeket mutatják a különböző esetekre.

bemeneti mértékrendszer	kimeneti mértékrendszer	inkremensrendszer	F cím értékhatára	dimenzió
mm	mm	IR-A	0.001 - 250000	mm vagy fok/perc
		IR-B	0.0001 - 25000	
		IR-C	0.00001 - 2500	
		IR-A	0.0001 - 5000	mm vagy fok/ford
		IR-B	0.00001 - 500	
		IR-C	0.000001 - 50	
inch	mm	IR-A	0.0001 - 9842.5197	inch vagy fok/perc
		IR-B	0.00001 - 984.25197	
		IR-C	0.000001 - 98.25197	
		IR-A	0.00001 - 196.85039	inch vagy fok/ford
		IR-B	0.000001 - 19.685039	
		IR-C	0.0000001 - 1.9685039	
inch	inch	IR-A	0.0001 - 25000	inch vagy fok/perc
		IR-B	0.00001 - 2500	
		IR-C	0.000001 - 250	
		IR-A	0.00001 - 500	inch vagy fok/ford
		IR-B	0.000001 - 50	
		IR-C	0.0000001 - 5	
mm	inch	IR-A	0.001 - 250000	mm vagy fok/perc
		IR-B	0.0001-25000	
		IR-C	0.00001-2500	
		IR-A	0.0001 - 5000	mm vagy fok/ford
		IR-B	0.00001-500	
		IR-C	0.000001-50	

6.2.2 A munkaelőtolás értékének behatárolása

Az adott gépen programozható maximális előtolást a gép gyártója a paramétermezőben tengelyenként bekorlátozhatja. Az itt beállított érték mindig percenkénti dimenzióban értelmezett. Ez az érték egyben a SZÁRAZ FUTÁS kapcsoló bekapcsolt állapotában az előtoló mozgások sebessége. Ha ennél nagyobb előtolást programozunk, a program végrehajtása során a vezérlő a paraméter alapján bekorlátozza a sebességet.

A kézi mozgatás során az előtolás maximális értéke a paramétermezőben külön behatárolható, hogy az emberi reakcióidőt is be lehessen kalkulálni a megálláshoz.

6.3 Gyorsítás/lassítás. Az F előtolás figyelembe vétele

Mozgások indításánál a gyorsításra, megállításánál a lassításra azért van szükség, hogy az ilyenkor fellépő, a gépet mechanikusan igénybe vevő erők hatását minimalizáljuk, illetve elfogadható szinten tartsuk.

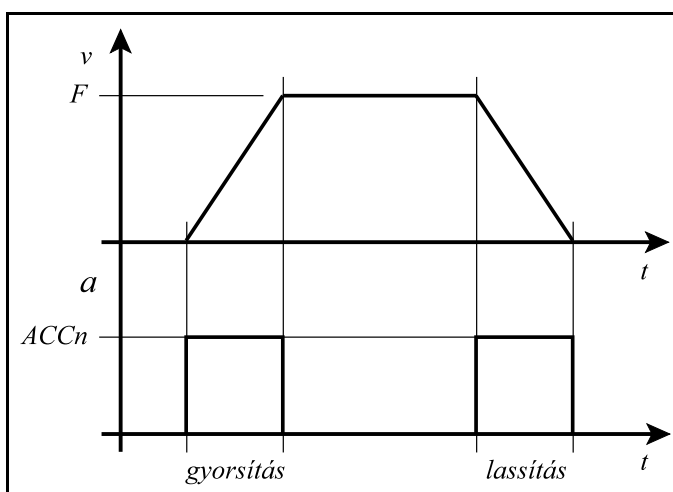
Normál körülmények között a vezérlő a következő esetekben gyorsít, illetve lassít:

- kézi mozgások esetén,
- gyorsmeneti pozicionálás (G0) során a mondat elején a mozgás mindig 0 sebességről indul, és a pozicionálás végén mindig 0 sebességre lassít,
- előtoló mozgások (G1, G2, G3) esetén G9, vagy G61 állapotban a mondat elején a mozgás mindig 0 sebességről indul, és a mozgás végén mindig 0 sebességre lassít,
- előtoló mozgások (G1, G2, G3) és több egymást követő előtoló mondat esetén a mondat sor elején gyorsít, a végén lassít,
- a fenti esetben az előtoló mondatok között is gyorsít, illetve lassít, ha sarkot detektál,
- a fenti esetben akkor is gyorsít, vagy lassít, ha az előtolást (F) valamelyik mondatban megváltoztatjuk, vagy az adott mondatban valamilyen előtolást korlátozó funkció érvényesül,
- lassít, ha az előtolást STOP gombbal megállítjuk, illetve gyorsít, ha az előtolást START-tal elindítjuk,
- lassítással áll meg, ha a mozgás után funkció végrehajtása következik és a mondat végén, ha a *MONDATONKÉNTI* kapcsoló hatásos.

A vezérlő mindig a közös (vektoriális) előtolás értékét gyorsítja, a tengelyekre eső előtoláskomponenseket nem. Kétféle gyorsítás állítható be:

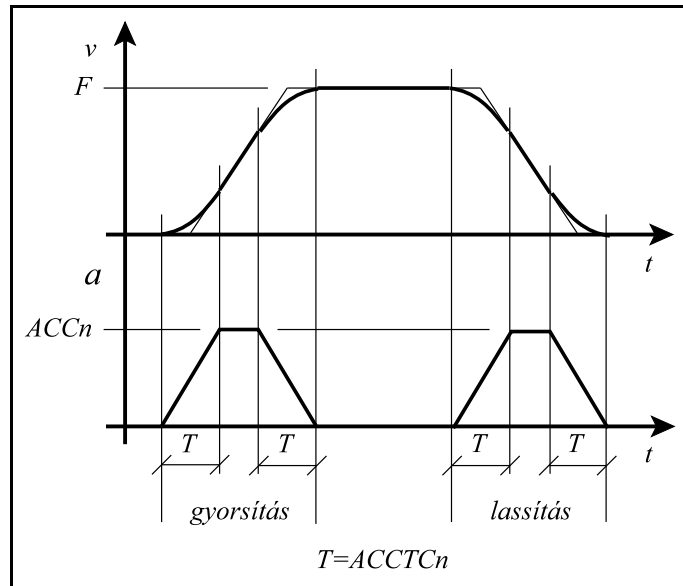
- lineáris és
- haranggörbe alakú.

Lineáris gyorsítás esetén a gyorsítás, illetve a lassítás alatt a gyorsulás értéke állandó, a vezérlő az előtolást lineáris függvény szerint növeli induláskor, illetve csökkenti megálláskor. Különböző gyorsítási érték állítható be tengelyenként az $ACCn$ paraméteren mm/sec^2 dimenzióban, igény szerint. Ha a mozgásban több tengely vesz részt mindig a legkisebb gyorsításra állított tengely paramétere alapján gyorsít, illetve lassít.



6.3-1 ábra

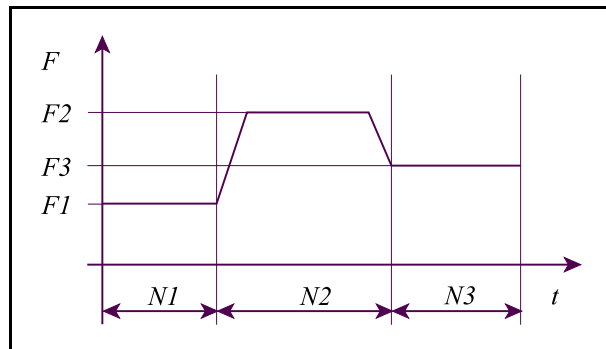
Haranggörbe alakú gyorsítás esetén a gyorsítás, illetve a lassítás alatt a gyorsulás értéke is változik, lineárisan nő, amíg el nem éri a beállított gyorsítási értéket ($ACCn$ paraméter) illetve lineárisan csökken, mielőtt eléri a célsebességet. Ennek következtében az előtolás fel-, lefutásának alakja az idő függvényében haranggörbe (másodfokú görbe), ezért nevezzük haranggörbe alakú gyorsításnak. Az a T idő, amely alatt a beállított gyorsulási értéket eléri a vezérlő, különbözőre állítható be tengelyenként az $ACCTCn$ paraméteren msec dimenzióban, igény szerint. Ha a mozgásban több tengely vesz részt mindig a legnagyobb időállandóra állított tengely paramétere alapján gyorsít, illetve lassít.



6.3-2 ábra

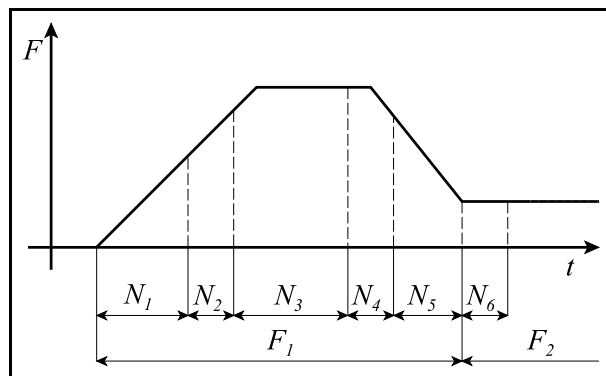
A gyorsítások és az időállandók értékét mindig a gép építője határozza meg tengelyenként, a gép dinamikus teherbírásának függvényében.

Új, az előzőnél nagyobb előtolásértékre való gyorsítást a vezérlő mindig annak a mondatnak a végrehajtása során kezdi el, amelyikben az új előtolást megadták. Ez a folyamat szükség esetén több mondaton is átnyúlhat. Az új, az előzőnél kisebb, előtolásértékre való lassítást a vezérlő egy olyan megfelelő megelőző mondatban kezdi el, hogy abban a mondatban, ahol az új előtolást megadták, már az abban a mondatban programozott sebességgel kezdje a megmunkálást.



6.3-3 ábra

A vezérlés a tangenciális sebességváltozásokat előre figyeli, és nyilvántartja. Erre azért van szükség, hogy a kívánt célsebességet akár több mondat végrehajtásán átnyúló folyamatos gyorsítással érje el. Ez a funkció csak a többszörös mondat előfeldolgozás (MULTIBUFFER) üzemmódban él ($MULBUF$ paraméter értéke 1).



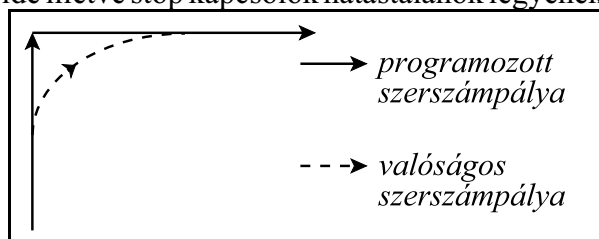
6.3-4 ábra

6.4 Az előtolásvezérlő funkciók

Az előtolásvezérlő funkciókra sarkok megmunkálásakor van szükség, illetve olyan esetben, amikor a technológia azt kívánja meg, hogy az *override* illetve *stop* kapcsolók hatástalanok legyenek.

Sarkok megmunkálásakor, ha folyamatos forgácsolás üzemmódot használunk, a szánok tehetetlenségük folytán nem képesek követni a vezérlés által kiadott útparancsokat. Ekkor a szerszám az előtolás függvényében kisebb, vagy nagyobb mértékben lekerekíti a sarkot.

Ha a munkadarabnál éles sarkokra van szükség, akkor a vezérlővel közölni kell, hogy a mozgás végén lassítson le, várja meg amíg a tengelyek megállnak, és a következő mozgást csak ezután indítsa.



6.4-1 ábra

6.4.1 G09: pontos megállás

Ez a funkció nem öröklődik, csak abban a mondatban érvényes amelyikben programozták. Annak a mondatnak a végén, ahol megadásra került a vezérlés az interpoláció végrehajtása után lelassít, megáll és megvárja a mérőrendszer pozícióban jelet. Ha a jel 5 másodpercen belül nem jön be *1020 POZÍCIÓ HIBA* üzenetet ad a vezérlés. Ez a funkció éles sarkok pontos kerülésére szolgál.

6.4.2 G61: pontos megállás üzemmód

Öröklődő funkció. Törlésre kerül G62, G63, G64 paranccsal. A vezérlés minden interpoláció végrehajtása után lelassít, megáll és megvárja a mérőrendszer pozícióban jelet, és csak ezután indítja a következő interpolációs ciklust. Ha a "pozícióban" jel 5 másodpercen belül nem érkezik meg *1020 POZÍCIÓ HIBA* jelzést ad a vezérlő.

6.4.3 G64: folyamatos forgácsolás üzemmód

Öröklődő funkció. Bekapcsolás után a vezérlő ezt az állapotot veszi fel. A következő kódok szüntetik meg ezt az állapotot: G61, G62, G63.

Ebben az üzemmódban az interpoláció végrehajtása után nem áll meg a mozgás, nem lassítanak le a szánok, hanem azonnal elkezdődik a következő mondat interpolációja.

Ebben az üzemmódban éles sarkokat nem lehet megmunkálni, mert az átmenetknél lekerekíti azokat a szánok tehetetlensége folytán.

6.4.4 G63: a százalékos kapcsoló és stop tiltás üzemmód

Öröklődő funkció. A G61, G62, G64 kódok megszüntetik ezt az állapotot.

Ebben az üzemmódban az előtolás- és főorsó százalékkapcsoló, valamint az előtolás stop hatástalan. A százalékos értékeket függetlenül azok állásától 100%-nak veszi. Az interpoláció végrehajtása után nem lassít le, hanem azonnal indítja a következő interpolációs ciklust.

Ezt az üzemmódot különböző menetmegmunkálások esetén lehet használni.

6.4.5 Automatikus előtoláscsökkentés belső sarkoknál. (G62)

Öröklődő funkció. A G61, G63, G64 kódok megszüntetik ezt az állapotot.

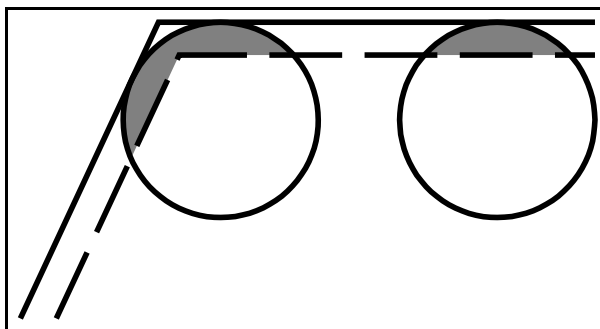
Belső sarkok megmunkálása esetén a szerszámmra ható erő megnövekszik a sarok előtti és utáni szakaszon. Annak érdekében, hogy a szerszám ne rezegjen be, és a felület megfelelő maradjon, a vezérlő G62 bekapcsolt állapotában a belső sarkok előtti és utáni szakaszon az előtolást automatikusan csökkenti.

Az előtoláscsökkentés a következő feltételek mellett hatásos:

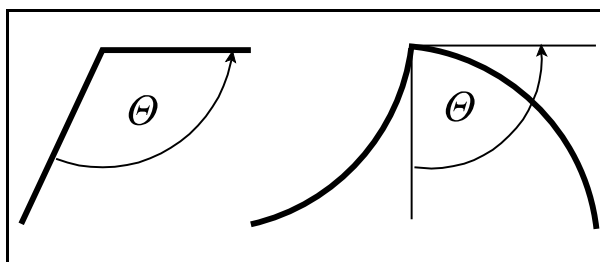
- 1. a síkbeli szerszámsugár korrekció bekapcsolt állapotában (G41, G42),
- 2. G0, G1, G2, G3 mondatok között,
- 3. a kiválasztott síkban végzett mozgásoknál,
- 4. ha a sarkot belülről kerüli a szerszám,
- 5. ha a sarok szöge kisebb, mint egy a paramétermezőben meghatározott szög,
- 6. a sarok előtt, és után a paramétermezőben meghatározott távolságra.

Az előtoláscsökkentés funkció mind a négy lehetséges átmenetre: egyenes–egyenes, egyenes–kör, kör–egyenes, kör–kör működik.

A Θ belső szög értékét a *CORNANGL* paraméteren lehet beállítani az 1–180° szögterületben.

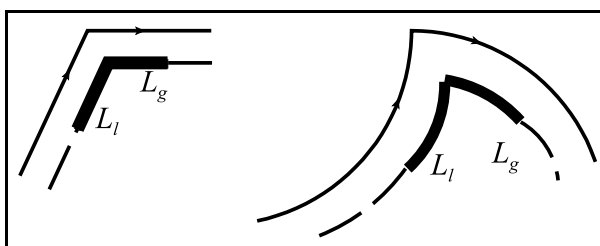


6.4.5-1 ábra



6.4.5-2 ábra

A sarok előtt L_1 távolságra kezd lassítani, a sarok után pedig L_g távolságra gyorsítani. Körívek esetén L_1 és L_g távolságot az ív mentén veszi figyelembe a vezérlő. L_1 távolság megadása a *DECDIST* paraméteren, L_g távolságé pedig az *ACCDIST* paraméteren történik.



6.4.5-3 ábra

A százalék értékét, amire az előtolást csökkenteni akarjuk, a *CORNOVER* paraméteren százalékosan lehet beállítani. Az override a sarok előtt L_1 távolságtól kezdve lesz hatásos, és a sarok után L_g távolságig tart. Az előtolás % és a sarok % értékeket együtt veszi figyelembe a vezérlő:

$$F * \text{előtolás \%} * \text{sarok \%}$$

Ha G62 állapotban pontos megállást akarunk programozni az adott mondatba G09-et kell írni.

6.4.6 Automatikus előtoláscsökkentés belső körívекnél.

A síkbeli szerszámsugár korrekció bekapcsolt állapotában (G41, G42) körívek belső megmunkálásakor a vezérlő automatikusan csökkenti az előtolás értékét, hogy a forgácsolási sugáron legyen hatásos a programozott előtolás.

Az előtolás nagysága a szerszámsugár középpontján:

ldn

$$F_c = \frac{R_c}{R} F$$

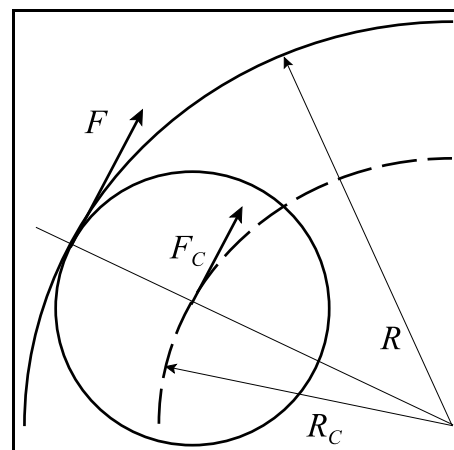
F_c : a szerszámsugár középpont előtolása (korrigált előtolás)

R : a programozott körsugár

R_c : a korrigált körsugár

F : a programozott előtolás.

Az automatikus előtoláscsökkentésnek alsó határt szab a *CIRCOVER* paraméter, ahol százalékosan meg lehet adni azelőtolás csökkentés minimumát. A körsugár miatti override összeszorozódik az előtolás és sarokoverride értékekkel, és így kerül kiadásra.



6.4.6-1 ábra

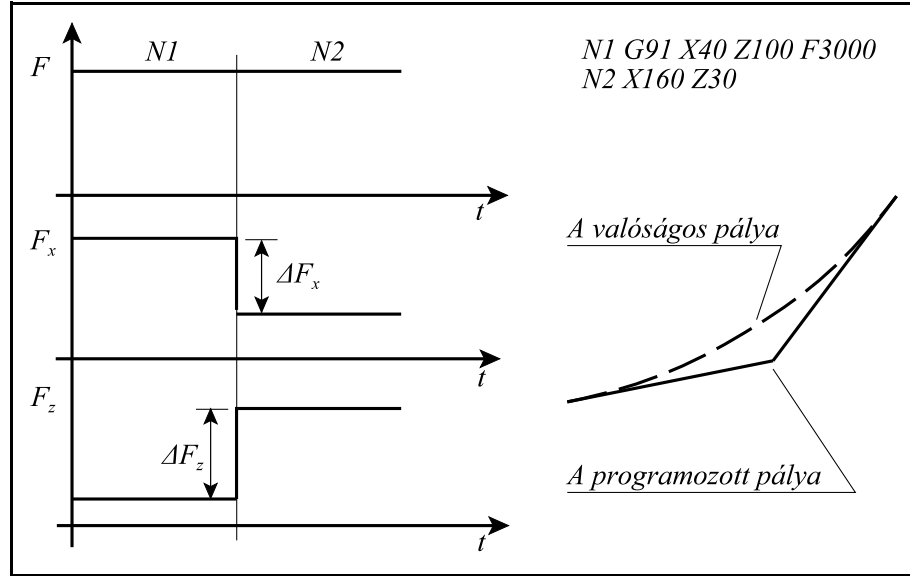
6.5 Automatikus lassítás sarkoknál

A G0 gyorsmeneti pozícionálás a mondat végpontján mindig 0 sebességre lelassít és a következő mozgásmondat végrehajtása 0-ról való gyorsítással indul. Előtoló mondatok között (G1, G2, G3) a mondat végpontjában a vezérlő csak megfelelő paraméterállítás, előtolásnagyság és a pálya erőteljes irányváltása, "sark" esetén lassít le. Ha a pályában nincs törés, azaz erőteljes irányváltás, nem kell, illetve fölösleges és káros lassítani.

Két okból van szükség az előtolásváltozások (sarkok) detektálására, és egyúttal az előtolás lassítására:

- A pálya hirtelen irányváltozásából adódó, tengelyenkénti előtolásváltozások olyan nagyok lehetnek, hogy lassítás nélkül a hajtások nem tudják lengés nélkül követni azt, és ez a pontosság rovására megy, illetve mechanikusan túlzottan igénybe veszi a szerszámgépet.
- A pálya hirtelen irányváltozása sarkot jelent és ha a sarkot "élessé" akarjuk tenni a forgácsolás során, szintén le kell lassítani. Minél jobban lelassítjuk az előtolást, a sark annál élesebb lesz.

Ha két, egymást követő N1, N2 mondatban a saroknál nem lassítunk, akkor az egyes tengelyek mentén az ábrán látható előtoláskülönbségek (ΔF_x , ΔF_z) lépnek fel, amely azt eredményezi, hogy a valóságban a sarkot lekerékíti a szerszám.



6.5-1 ábra

Ahhoz, hogy a saroklassítás funkció működjön a 2501 CDEN paramétert 1-be kell írni.

Sarkok detektálását a vezérlő kétféleképp végezheti: a pálya irányszögének változását, illetve a tengelyenkénti előtoláskomponensek változását figyelve. Paraméter alapján ki lehet választani, hogy melyik módszer alapján működjön.

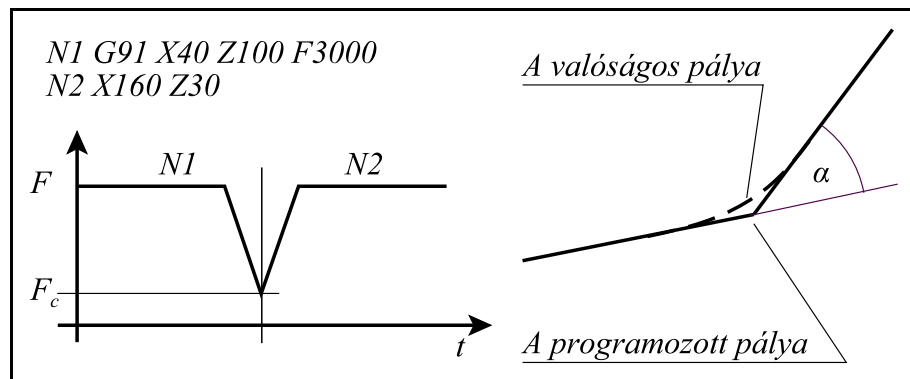
Lassítás sarkoknál a pálya irányszögének változását figyelve.

A 2501 CDEN=1 és a 2502 FEEDDIF=0 paraméterállásnál, a lassítás a pálya irányszögének változását figyelve történik. Ez a beállítás G94 (percenkénti előtolás), és G95 (fordulatonkénti előtolás) állapotban is működik.

☞ *Figyelem: Mivel esztergáknál túlnyomórészt fordulatonkénti előtolást programozunk ezért itt ezt a beállítást kell használni.*

Ha a mellékelt ábrán látható N1, N2 mondat találkozásánál az α szög túllépi a paraméteren engedélyezett értéket a vezérlő lelassítja az előtolást F_c értékre.

A 2511 CRITICAN paraméteren állítható be a kritikus szög értéke fokban. A 2512



6.5-2 ábra

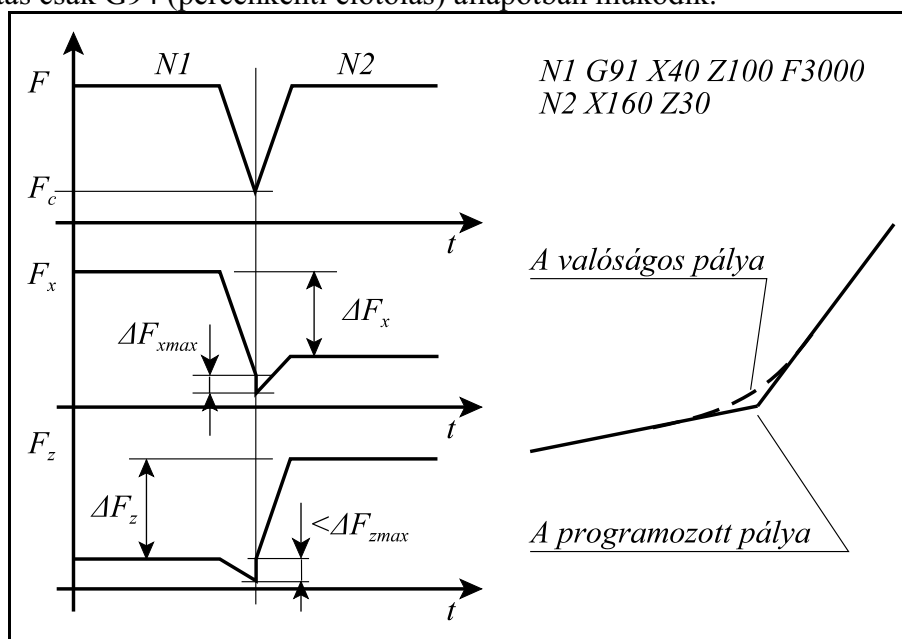
FEEDCORN paraméter értéke adja meg, hogy a kritikus szöget túllépve mekkora előtolásra lassítson le a vezérlő: $F_c = \text{FEEDCORN}$.

Lassítás sarkoknál a tengelyenkénti előtoláskomponensek változását figyelve.

Ha a 2501 CDEN=1 és a 2502 FEEDDIF=1, a lassítás az előtoláskomponensek változását figyelve történik. Ez a beállítás csak G94 (percenkénti előtolás) állapotban működik.

Ha a mellékelt ábrán látható módon az N1, N2 mondat találkozásánál az előtolást lelassítjuk úgy, hogy az előtolásváltozás mértéke egyik tengelyen se lépje túl az arra a tengelyre paraméteren engedélyezett (ΔF_{xmax} , ΔF_{zmax}) kritikus előtoláskülönbséget, akkor a szerszám a kritikus előtolás függvényében kihegyezi a sarkot.

A sarokpontban az előtolást úgy kapjuk,



6.5-3 ábra

hogya a kritikus előtolás értékeket tengelyenként elosztjuk a kiadódó előtolásváltozások értékével és a minimum értéket megszorozzuk a programozott előtolással. Az így kiadódó F_c előtolásra lassít le a saroknál:

$$F_c = \min \left\{ \frac{\Delta F_{x \max}}{\Delta F_x}, \frac{\Delta F_{z \max}}{\Delta F_z}, \dots \right\} \times F$$

ahol:

$\Delta F_{xmax}, \Delta F_{zmax}, \dots$: az X, Z, ... tengelyekre beállított megfelelő 252n CRITFDIFn paraméter,
 $\Delta F_x, \Delta F_z, \dots$: az X, Z, ... tengelyeken fellépő előtoláskülönbség.

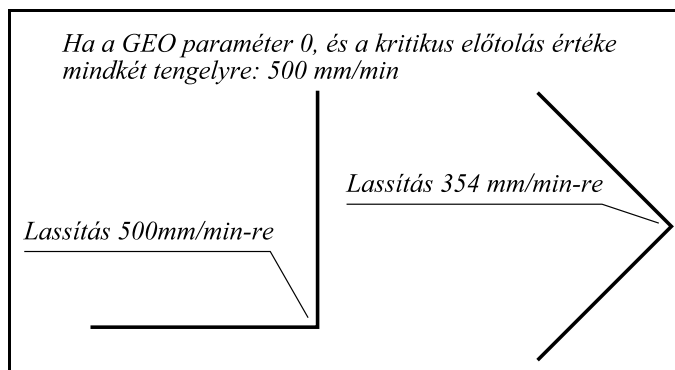
Az így lecsökkentett előtolás mértéke függ a sarok geometriai elhelyezkedésétől. Tekintsük a következő példát:

Ha a tengelyekkel párhuzamos irányokban kerülünk egy 90°-os sarkot, és mindkét tengelyen a kritikus előtolás értéke 500mm/min, akkor erre a sebességre kell lassítani az előtolást a saroknál.

Ha viszont a derékszögű sarok szárai 45°-os szöget zárnak be a tengelyekkel, akkor 354 mm/min-re kell lassítani.

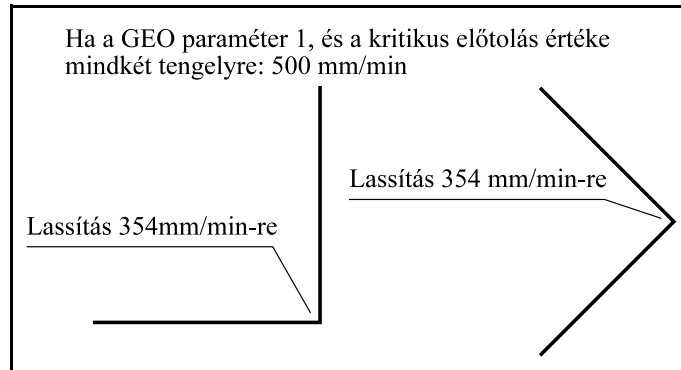
A 2503 GEO=0 paraméterállásnál a fentiek szerint jár el a vezérlő.

Így az előtolás mindig a lehető legmagasabb lesz.



6.5-4 ábra

A 2503 GEO=1 paraméterállásnál, a vezérlő a legrosszabb (45°-os esetből) indul ki, és a szög szárainak geometriai elhelyezkedésétől függetlenül a 45°-os esetről érvényes előtolással dolgozik. Ez max. 30%-os előtoláscsökkenést eredményezhet.



6.5-5 ábra

Figyelem:

A sarkoknál történő automatikus előtolásllassítás nem tévesztendő össze a pontos megállás (G9, G61) funkcióval. Az utóbbi esetben a vezérlő mindig, minden mondat végpontjában 0 sebességre lassít és megvárja a "pozícióban" jelet.

Úgyszintén nem tévesztendő össze a belső sarkoknál történő automatikus előtoláscsökkenéssel (G62), amely csak G41, G42 állapotban hatásos. Ekkor már a sarok előtt paraméteren meghatározott távolságra lecsökkenti az előtolás értékét.

6.6 A pálya mentén normális irányban fellépő gyorsulások korlátozása köríveknél

A vezérlés a megmunkálás során az előtolást a pálya érintője mentén (tangenciális irányban) állandó értéken tartja. Ennek az a következménye, hogy tangenciális irányban nem lépnek fel gyorsuláskomponensek. Nem úgy normális (a pályára, illetve a sebességre merőleges) irányban. A normális irányú gyorsulás tengelyekre eső komponensei az egyes tengelyeken túlléphetik az adott tengelyre megengedett értéket. Ezt elkerülendő a pálya menti sebességet a pálya görbületének mértékében korlátozni kell.

Körívek megmunkálása során az előtolás F nagyságát az

$$F = \sqrt{a \cdot r}$$

összefüggés alapján bekorlátozza, ahol:

a : a körinterpolációban részt vevő tengelyekre megadott gyorsulásértékek (470n ACCn paraméterek) közül a kisebb,

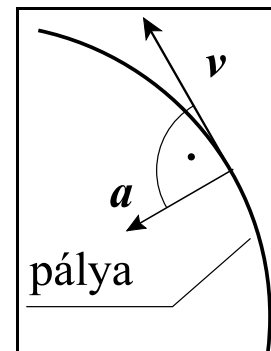
r : a kör sugara.

A körinterpolációt már az így kiszámított sebességgel kezdi el.

A 2513 CIRC FMIN paraméteren megadott előtolásnál kisebbre nem csökkenti a sebességet, függetlenül a fenti összefüggéstől.

Figyelem:

Ez a funkció nem tévesztendő össze a körívek belső megmunkálásánál G41, G42 állapotban történő automatikus előtoláscsökkenéssel.



6.6-1 ábra

7 A várakozás

A

(G94) G04 P....

parancsal várakozási időt programozhatunk másodpercben.

P értékhatára: 0.001 - 99999.999 másodperc.

A

(G95) G04 P....

parancsal várakozási időt programozhatunk főorsó fordulatban.

P értékhatára: 0.001 - 99999.999 fordulat.

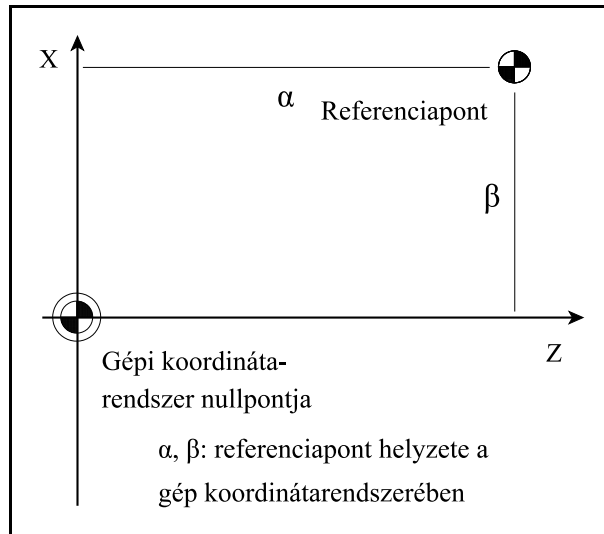
A *SECOND* paraméter függvényében a késleltetés vonatkozhat mindig másodpercre is G94, és G95 állapottól függetlenül.

A várakozás mindig a következő mondat végrehajtásának programozott késleltetését jelenti. Nem öröklődő funkció.

A várakozás alatt az 5. interpolációs állapotot kijelző státuszablakban a **VÁR** felírat jelenik meg, hogy felhívja a figyelmet a szánok mozdulatlanságának az okára.

8 A referenciapont

A referenciapont egy kitüntetett pozíció a szerzőgépen, amelyre könnyen rá tud állni a vezérlő. A referenciapont helyzetét a gép koordináta-rendszerében paraméter alapján lehet meghatározni. Referenciapontfelvétel után lehet bemérni a munkadarab koordináta-rendszereket, és abszolút pozícióra állni. Referenciapont felvétel után hatásosak csak a paraméteres végállások és a programozott munkatérbehatárolás.



8-1 ábra

8.1 Automatikus referenciapont felvétel (G28)

A

G28 v

utasítás a v vektorban meghatározott tengelyeken referenciapontot vesz fel. A mozgás két részből tevődik össze.

Először a v vektor által meghatározott koordinátákat közbülső pontnak véve, gyorsmenettel a v vektor által meghatározott közbülső koordinátákra áll lineáris mozgással. A megadott koordinátaértékek lehetnek abszolút, illetve inkrementális értékek is. A mozgás mindig az aktuális koordináta-rendszerben történik.

A lineáris mozgás végpontjára úgy áll rá, hogy a síkbeli szerszámsugár korrekciósvektor törlődik. A közbülső pont koordinátáit a v vektor által meghatározott tengelyeken eltárolja.

A második fázisban a közbülső pontról a v vektor által meghatározott tengelyeken egyidejűleg, a kézi referenciapontfelvétel által meghatározott menet szerint, refpontra áll.

A referenciapontfelvétel tengelyenként meghatározott sebességgel, nemlineáris mozgással történik. A referenciapont felvétele után, mint kézi mozgásnál, a paramétermezőben meghatározott módon felveszi az alaphelyzetet.

Nem öröklődő kód.

☞ Megjegyzések:

- G28 parancs mindig a kézi referenciapontfelvételnek megfelelően veszi fel a referenciapontot.
- Ha még nincs érvényes referenciapont, a G28 parancsban szereplő v közbülső koordinátáknak inkrementális értéket kell adni.
- A G28 mondatban programozott v közbülső koordináták eltárolódnak, és kikapcsolásig megőrződnek. Más szavakkal, azoknál a koordinátáknál, amelyeknek a pillanatnyi G28 parancsban nem adtunk értéket, a korábbi G28 parancsban meghatározott közbülső érték marad érvényben. Például:

G28 X100 közbülső pont: X=100, Z=0

G28 Z200 közbülső pont: X=100, Z=200

8.2 Az 1., 2., 3., 4. referenciapontra állás (G30)

A

G30 v P

utasítással a P címen meghatározott referenciapontra küldi a v vektor címein meghatározott koordinátájú tengelyeket.

P1: 1. referenciapont

P2: 2. referenciapont

P3: 3. referenciapont

P4: 4. referenciapont

A referenciapontok a szerszámgépen a gép koordinátarendszerében paramétereken meghatározott koordinátájú pontok (*REFPOS1*, ..., *REFPOS4*), amelyeket általában a különböző cserepozíciók meghatározására használunk. Például: szerszámcserehely, vagy palettacsere helyzete. Az első referenciapont mindig a gépi referenciapont helyzete, azaz az a pont ahová a vezérlés referenciapontra futás után áll.

Az utasítás csak a gépi referenciapont felvétele után használható.

A mozgás két részből tevődik össze. Először a v vektor által meghatározott koordinátákat közbülső pontnak véve, gyorsmenettel a v vektor által meghatározott közbülső koordinátákra áll lineáris mozgással. A megadott koordinátaértékek lehetnek abszolút, illetve inkrementális értékek is. A mozgás mindig az aktuális koordinátarendszerben történik. A lineáris mozgás végpontjára úgy áll rá, hogy a síkbeli szerszámugár korrekciósvektor törlődik. A közbülső pont koordinátáit a v vektor által meghatározott tengelyeken az aktuális koordinátarendszerben eltárolja. Az így eltárolt koordináták felülírják a G28 utasításban eltároltakat.

A második fázisban a közbülső pontról a v vektor által kijelölt tengelyek lineáris gyorsmeneti mozgással a P címen kiválasztott referenciapontra állnak.

A referenciapontra állás a korrekciós vektorok (hossz, eltolás, 3 dimenziós sugár) figyelmen kívül hagyásával történik, azokat a G30 utasítás kiadása előtt törölni nem kell, viszont a további mozgások programozásánál a vezérlés érvényesíti azokat. A síkbeli szerszámugár korrekció automatikusan visszakapcsolódik az első mozgásmondatban.

Nem öröklődő kód.

8.3 Automatikus visszatérés a referenciapontról (G29)

A

G29 v

utasítással a v vektoron meghatározott tengelyek mentén visszatér a vezérlés a referenciapontról. A G29 utasítás G28-at és G30-at követően ugyanúgy hajtódik végre. A visszatérés két ütemben zajlik le.

Az első fázisban a referenciapontról a G28, vagy G30 utasítás végrehajtása közben bejegyzett közbülső pontra áll a v vektoron meghatározott tengelyeken. A közbülső pont koordinátája öröklődik, vagyis, ha olyan tengelyre hivatkozunk, amelyikre nem történt koordinátaadás a G29-et megelőző G28 vagy G30 mondatban az eggyel korábbi értéket veszi figyelembe. A közbülső pont koordinátáinak értékét a kézi vagy az automatikus referenciapontfelvétel inicializálja, vagyis a refpontfelvétel megkezdése előtti koordinátákat jegyzi be közbülső pontként. A közbülső pontra a szerszámhossz, szerszámeltolás és a 3 dimenziós szerszámugár korrekció figyelembe vételével áll rá.

A közbülső pont koordinátája mindig az aktuális munkadarab koordinátarendszerben érvényes. Tehát, ha például a refpontra állás után és a G29 utasítás előtt munkadarab koordinátarendszer váltást programoztak a közbülső pontot az új koordinátarendszerben veszi figyelembe.

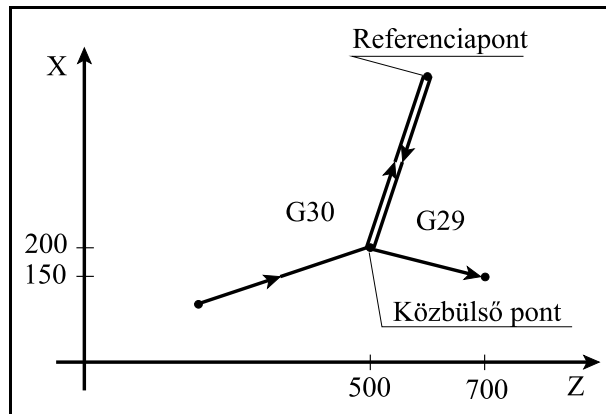
A második fázisban a közbülső pontról a G29 utasításban meghatározott v koordinátájú pontra áll. Ha a v koordináta értéke inkrementális, az elmozdulást a közbülső ponttól számítja.

A síkbeli szerszámsugár korrekció bekapcsolt állapotában a mozgás végpontjára a korrekciós vektor figyelembevételével áll rá.

Nem öröklődő kód.

Alkalmazási példa G30 és G29 használatára:

```
...  
G90  
...  
G30 P1 X200 Z500  
G29 X150 Z700  
...  
...
```



8.3-1 ábra

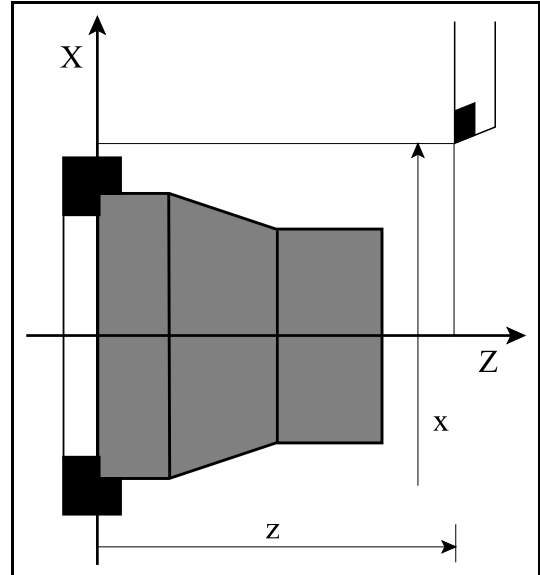
9 Koordinátarendszerek, síkválasztás

A programban egy pozíciót, ahova a szerszámot akarjuk mozgatni, koordinátaadatokkal adunk meg. Ha 2 tengelyünk van (X, Z) a szerszám pozícióját két koordinátaadat X ___ Z ___ fejezi ki.

Ahány tengely van a gépen a szerszám pozícióját annyi különböző koordinátaadat fejezi ki. A koordinátaadatok mindig egy adott koordinátarendszerben értendők.

A vezérlés háromféle koordinátarendszert különböztet meg:

1. a gépi koordinátarendszert,
2. a munkadarab koordinátarendszert,
3. a lokális koordinátarendszert.

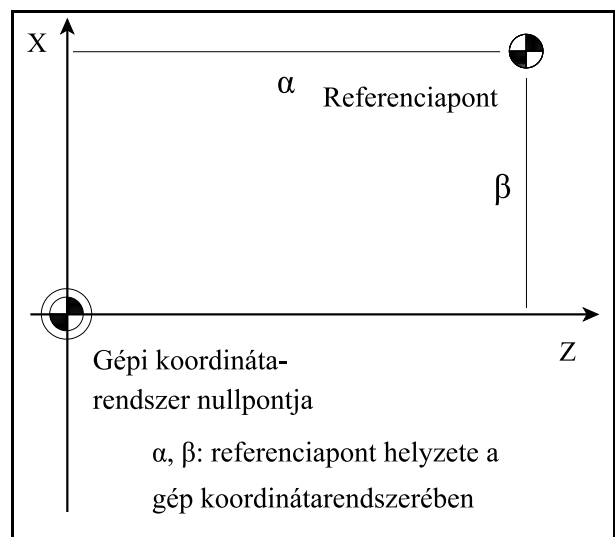


9-1 ábra

9.1 A gépi koordinátarendszer

A gép nullpontja, azaz a gépi koordinátarendszer nullpontja, egy olyan pont az adott szerzsámgépen, amit általában a gép építője határoz meg. A gépi koordinátarendszert a vezérlés a referenciapont felvétel alkalmával határozza meg.

Miután a gépi koordinátarendszer meghatározásra került, azt sem a munkadarab koordinátarendszer váltás (G54...G59), sem egyéb koordinátatranszformáció (G52, G92) nem változtatja meg, csak a vezérlés kikapcsolása.



9.1-1 ábra

9.1.1 A gépi koordinátarendszer beállítása

A gépi koordinátarendszert referenciapontfelvétel után a paramétermezőben lehet beállítani. A 7021 REFPOS1 paraméter csoportra a referenciapont távolságát kell beírni tengelyenként a gépi koordinátarendszer nullpontjától számítva.

9.1.2 A gépi koordinátarendszer kiválasztása (G53)

A

G53 v

utasítás hatására a szerszám a gépi koordinátarendszer v pozíciójú pontjára mozog.

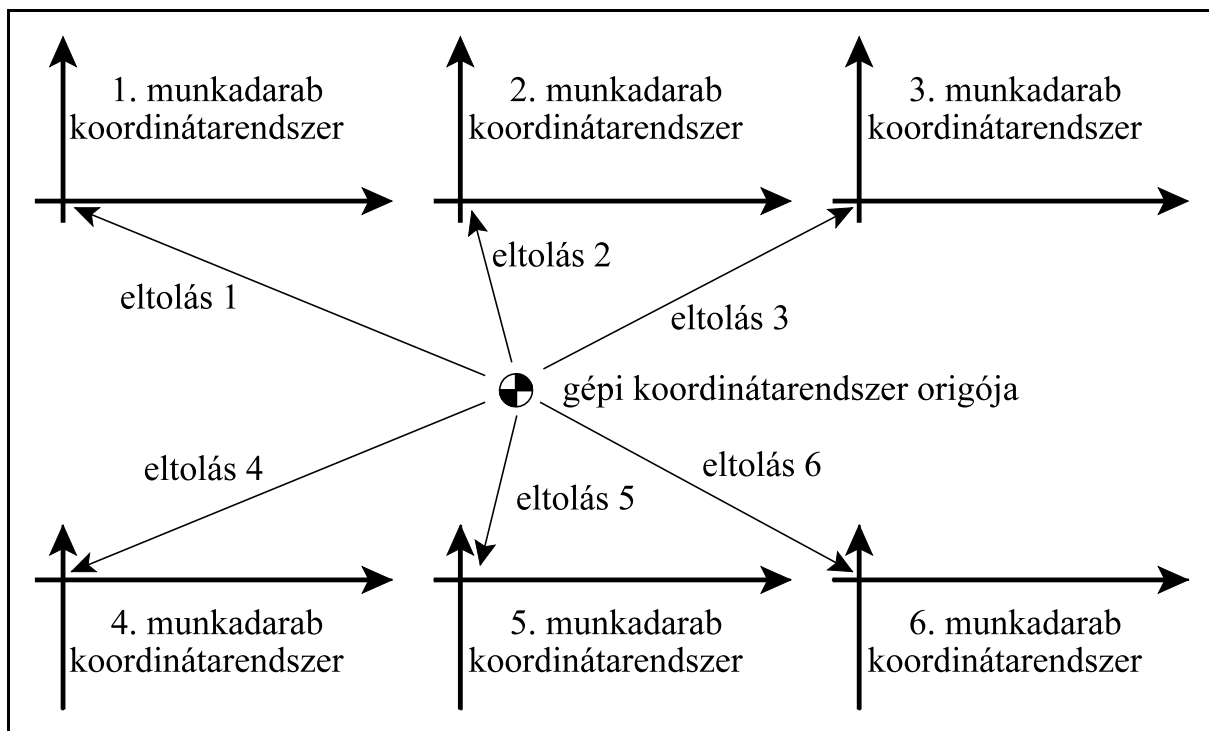
- G90, G91 állapottól függetlenül a v koordinátákat mindig abszolút koordinátaként kezeli,
- I operátor a koordináták címe után, vagy U, V, W cím (ha U, V, W nincsenek tengelynek kijelölve) hatástalan
- a mozgás mindig gyorsmenettel történik G00 utasításhoz hasonlóan,
- a pozícionálás mindig a beállított szerszámkorrekciók figyelembe vételével történik.

G53 utasítást csak referenciapontfelvétel után lehet végrehajtani. A G53 parancs egylovétű, csak abban a mondatban hatásos, ahol megadásra került.

9.2 A munkadarab koordinátarendszerek

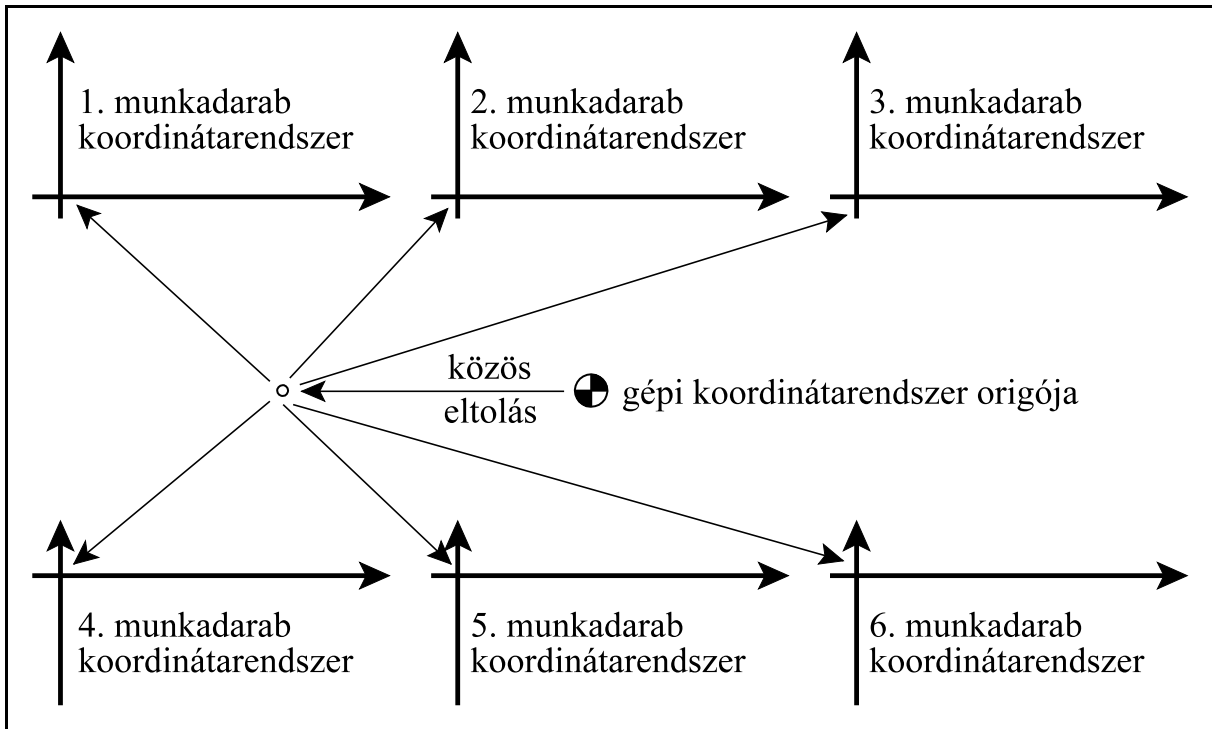
Azt a koordinátarendszert, amelyet a munkadarab forgácsolásakor használunk munkadarab koordinátarendszernek nevezzük. Hat különböző munkadarab koordinátarendszert lehet definiálni a vezérlőben.

9.2.1 A munkadarab koordinátarendszerek beállítása



9.2.1-1 ábra

Beállítás üzemmódban meg lehet állapítani a különböző munkadarab koordinátarendszerek helyzetét a gépi koordinátarendszerben, és a szükséges eltolásokat beállítani.



9.2.1-2 ábra

Ezen túlmenően az összes munkadarab koordinátarendszert el lehet tolni egy közös értékkel, amelyet szintén a beállítás üzemmódban adhatunk be.

9.2.2 A munkadarab koordinátarendszer kiválasztása

A G54...G59 utasításokkal lehet a különböző munkadarab koordinátarendszereket kiválasztani.

- G54.....1. munkadarab koordinátarendszer
- G55.....2. munkadarab koordinátarendszer
- G56.....3. munkadarab koordinátarendszer
- G57.....4. munkadarab koordinátarendszer
- G58.....5. munkadarab koordinátarendszer
- G59.....6. munkadarab koordinátarendszer

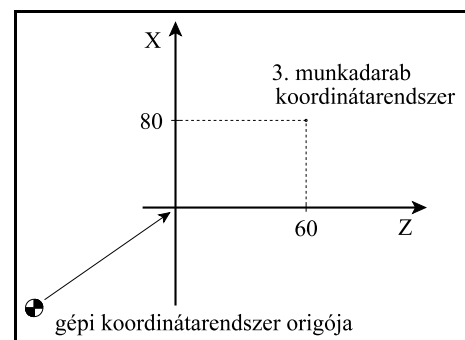
Öröklődő funkciók. Referenciapontfelvétel előtt választásuk hatástalan. Referenciapontfelvétel után a G54 1. munkadarab koordinátarendszer kerül kiválasztásra.

Az interpolációs mondatok abszolút koordinátaadatait a vezérlés az aktuális munkadarab koordinátarendszerben veszi figyelembe.

Például a

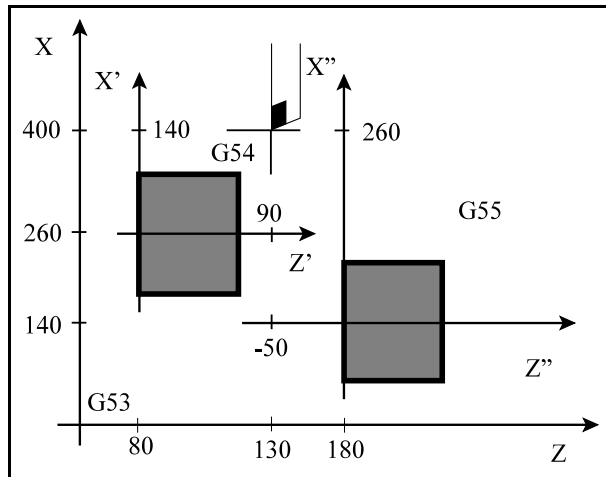
```
G56 G90 G00 X80 Z60
```

utasítás esetén a 3. munkadarab koordinátarendszer $X=80$, $Z=60$ pontjára áll rá. A munkadarab koordinátarendszerek eltolásait a be lehet mérni, és az eltolás értékek kikapcsolás után is megőrződnek.



9.2.2-1 ábra

A munkadarab koordinátarendszer váltással a szerszám pozíciója az új koordinátarendszerben kerül kijelzésre. Például az ábra szerint a G54 munkadarab koordinátarendszer eltolása a gép koordinátarendszerében $X=260, Z=80$. A G55 munkadarab koordinátarendszer eltolása a gép koordinátarendszerében számítva $X=140, Z=180$. A G54 X', Z' koordinátarendszerben a szerszám pozíciója $X'=140, Z'=90$. A G55 utasítás hatására a szerszám pozíciója az X'', Z'' koordinátarendszerben kerül értelmezésre: $X''=260, Z''=-50$.



9.2.2-2 ábra

9.2.3 A munkadarab koordinátarendszerek eltolásának állítása programból

A munkadarab koordinátarendszereket, és a munkadarab koordinátarendszerek közös eltolását be lehet állítani programutasítással is.

A

G10 v L2 P_p

utasítással, ahol

$p = 0$ közös eltolás állítása

$p = 1 \dots 6$ 1.- 6. munkadarab koordinátarendszer állítása

$v (X, Z, \dots)$: tengelyenkénti eltolási érték

A koordinátaadatok mindig derékszögű, abszolút értékűként kerülnek beolvasásra. G10 utasítás egylővetű.

9.2.4 Új munkadarab koordinátarendszer létrehozása (G92)

A

G92 v

utasítás hatására új munkadarab koordinátarendszer képződik úgy, hogy egy kijelölt pont, például a szerszám hegye, ha van hosszkorrekció programozva, vagy a szerszámtartó bázispontja, ha nincs hosszkorrekció, lesz az új munkadarab koordinátarendszer v koordinátájú pontja. Ezután bármely következő abszolút parancs ebben az új munkadarab koordinátarendszerben értendő, és a pozíciókijelzés is ebben a koordinátarendszerben képződik. A G92 parancsban megadott koordináták mindig derékszögű, abszolút értékűként kerülnek értelmezésre.

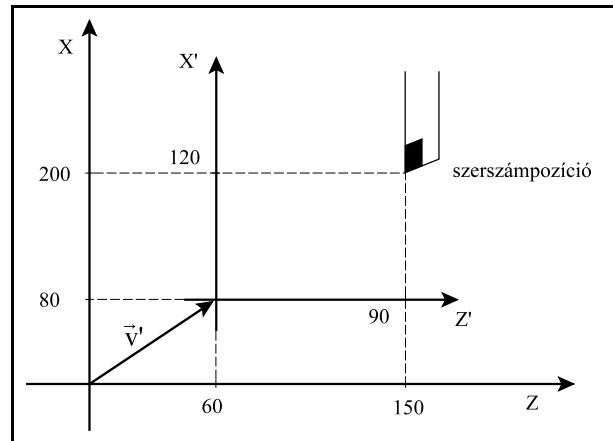
Ha például a szerszám az $X=200$, $Z=150$ koordinátájú ponton tartózkodik az aktuális X , Z munkadarab koordináta-rendszerben, a

G92 X120 Z90

utasítás hatására egy új X' , Z' koordináta-rendszer képződik, amelyben a szerszám az $X'=120$, $Z'=90$ koordinátájú pontra kerül. Az X , Z - X' , Z' koordináta-rendszerek közötti \vec{V}' eltolásvektor tengelyirányú komponensei:

$$V'_x = 200 - 80 = 120, \text{ illetve}$$

$$V'_z = 150 - 90 = 60.$$

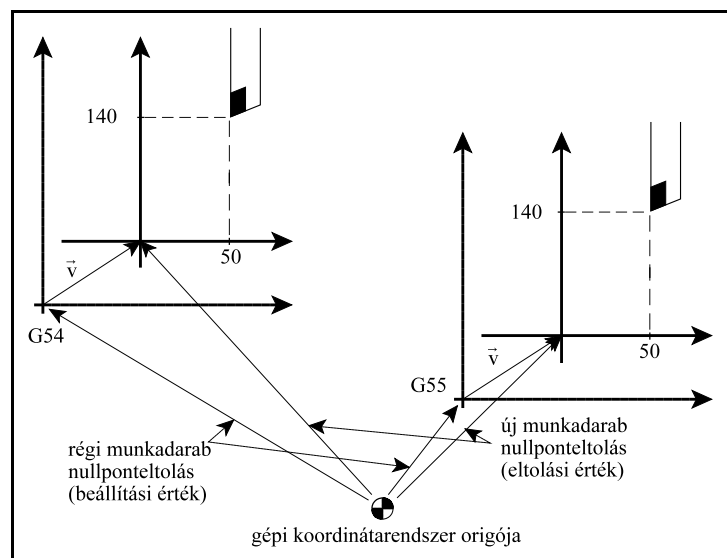


9.2.4-1 ábra

A G92 parancs mind a hat munkadarab koordináta-rendszerben érvényesül, azaz az egyikben kiszámított V eltolást a többi ötben is figyelembe veszi.

☞ Megjegyzések:

- A G92 utasítással beállított munkadarab koordináta-rendszer eltolását az M2, M30 programvége utasítás végrehajtása, és a program elejére való resetelés törli.
- G92 utasítás azokon a tengelyeken, amelyek az utasításban szerepelnek, törli a lokális koordináta-rendszer G52 utasítással programozott eltolásait.



9.2.4-2 ábra

9.3 A lokális koordináta-rendszer

Az alkatrészprogram írása közben könnyebb bizonyos esetekben a koordinátaadatokat nem a munkadarab koordináta-rendszerben megadni, hanem egy másik, úgynevezett lokális koordináta-rendszerben.

A

G52 v

utasítás egy lokális koordináta-rendszert hoz létre.

- Ha v koordináta abszolút értéként van megadva, a lokális koordináta-rendszer origója a munkadarab koordináta-rendszer v koordinátájú pontjára esik.
- Ha v koordináta inkrementális értéként van megadva a lokális koordináta-rendszer origóját v -vel eltolja, ha már előzőleg volt lokális koordináta-rendszer definiálva, illetve ha nem, akkor az eltolás a munkadarab koordináta-rendszer origójához képest képződik.

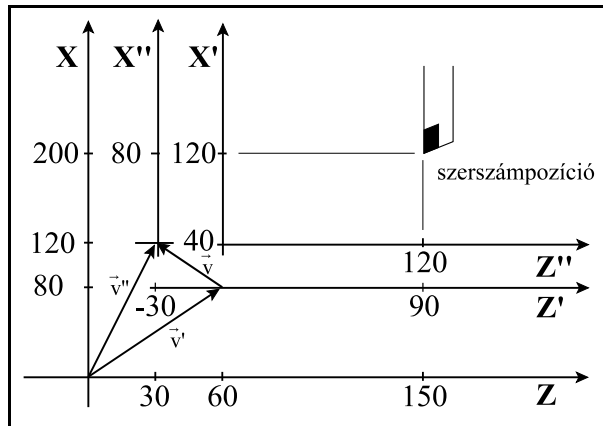
Ettől kezdve minden abszolút koordinátákkal megadott mozgásparancs az új koordináta-rendszerben kerül végrehajtásra. A pozíciókijelzés is az új koordináta-rendszerben történik. v koordináták értékeit mindig derékszögű koordinátaértéként kezeli.

Ha például a szerszám az $X=200, Z=150$ koordinátájú ponton tartózkodik az aktuális X, Z munkadarab koordinátarendszerben, a

`G90 G52 X80 Z60`

utasítás hatására egy új X', Z' lokális koordinátarendszer képződik, amelyben a szerszám az $X'=120, Z'=90$ koordinátájú pontra kerül. Az $X, Z-X', Z'$ koordinátarendszerek közötti V eltolásvektor tengelyirányú komponenseit határozzuk meg a `G52` utasítással: $V'_x=80$, illetve $V'_z=60$.

Ha most a lokális koordinátarendszert az X'', Z'' pozíciójára pontra akarjuk vinni az eljárás kétféle lehet:



9.3-1 ábra

- abszolút adatmegadással: `(G90) G52 X120 Z30` utasítás az X, Z munkadarab koordinátarendszerben az X'', Z'' koordinátarendszer origóját az $X=120, Z=30$ koordinátájú pontra állítja. V'' vektor komponensei $V''_x=120, V''_z=30$ értékadással képződnek.
- inkrementális adatmegadással: `G91 G52 X40 Z-30` utasítás az X', Z' koordinátarendszerben az X'', Z'' koordinátarendszer origóját az $X'=40, Z'=-30$ koordinátájú pontra állítja. V vektor komponensei $V_x=40, V_z=-30$ értékadással képződnek. A V'' vektor, amely az új lokális koordinátarendszer helyzetét mutatja az X, Y munkadarab koordinátarendszerben: $V''=V'+V$. Ennek komponensei: $V''_x=80+40=120, V''_z=60+(-30)=30$.

A szerszám pozíciója az X'', Y'' koordinátarendszerben: $X''=80, Z''=120$.

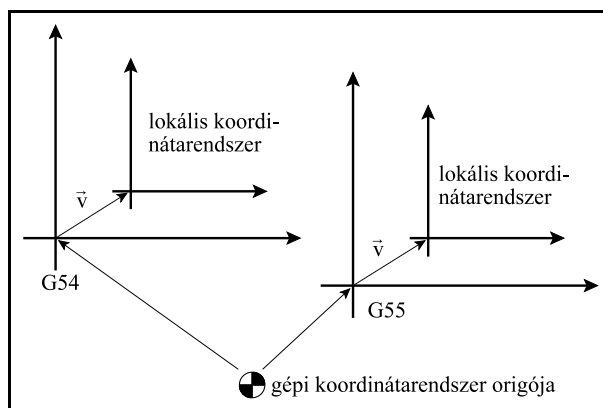
A

G90 G52 v0

utasítás törli a v koordinátájú pontokon az eltolásokat.

A lokális koordinátarendszer eltolása az összes munkadarab koordinátarendszerben érvényesül.

A `G92` utasítás programozása azokon a tengelyeken, amelyeknek értéket adtunk törli a `G52` utasítás által képzett eltolásokat, mintha `G52 v0` parancsot adtunk volna ki.



9.3-2 ábra

Ha a szerszám az X, Z munkadarab koordinátarendszer $X=240, Z=200$ koordinátájú pontján tartózkodik, a

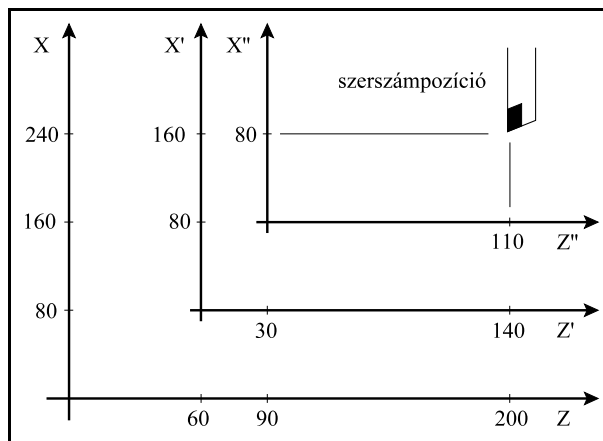
`G52 X80 Z60`

utasítás hatására az X', Z' lokális koordinátarendszerben a pozíciója $X'=160, Z'=140$ lesz.

Ezután a

`G92 X80 Z110`

parancs hatására az X'', Z'' új munkadarab koordinátarendszerben a szerszám pozíciója $X''=80, Z''=110$ lesz. Tehát az X', Z' lokális koordinátarendszer `G92` parancs hatására törlődik, mintha `G52 X0 Z0` parancsot adtunk



9.3-3 ábra

volna ki.

☞ *Megjegyzés:*

- M2, M30 parancs végrehajtása, illetve a program elejére való reszettelés törli a lokális koordinátarendszer eltolását.

9.4 Síkválasztás (G17, G18, G19)

A sík, amelyben

- a körinterpoláció,
- a síkbeli szerszámsugár korrekció,
- a fűróciklusok pozicionálása

érvényesül a következő G kódokkal választható ki:

G17..... $X_p Y_p$ sík

G18..... $Z_p X_p$ sík

G19..... $Y_p Z_p$ sík,

ahol: X_p : X, vagy a vele párhuzamos tengely,

Y_p : Y, vagy a vele párhuzamos tengely,

Z_p : Z, vagy a vele párhuzamos tengely.

A kiválasztott síkot nevezzük a fősíknak.

Az hogy a párhuzamos tengelyek közül melyik kerül kiválasztásra a G17, G18, vagy a G19 utasítással egy mondatba programozott tengelycímetől függ:

Ha X és U, Y és V, Z és W párhuzamos tengelyek:

G17 X___ Y___ az XY síkot,

G17 X___ V___ az XV síkot,

G17 U___ V___ az UV síkot,

G18 X___ W___ az XW síkot,

G19 Y___ Z___ az YZ síkot,

G19 V___ Z___ a VZ síkot,

választja ki.

Ha G17, G18, G19 nincs megadva egy mondatban a síkkijelölés változatlan marad:

G17 X___ Y___ XY sík

U___ Y___ marad az XY sík.

Ha a G17, G18, G19 mondatban nincs tengelycím megadva, akkor a főtengelyeket választja ki a vezérlés:

G17 az XY síkot,

G17 X az XY síkot,

G17 U az UY síkot,

G17 V az XV síkot,

G18 a ZX síkot,

G18 W a WX síkot

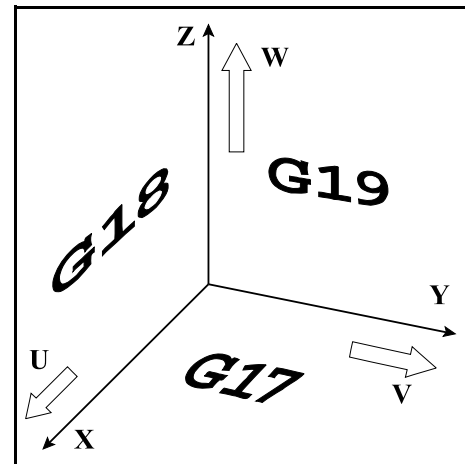
választja ki.

A mozgásparancs nem befolyásolja a síkválasztást:

(G90) G17 G00 Z100

hatására az XY sík kerül kiválasztásra, és a Z tengely a 100 koordinátájú pontra mozog.

Bekapcsolás után a paraméterező *CODES* paraméterén megadott sík: G17, vagy G18 kerül kiválasztásra.



9.4-1 ábra

Egy programon belül többször is válthatunk síkot.

U, V, W cím párhuzamos tengelyként való kijelölése a paramétermezőben történhet.

10 A főorsó funkció

10.1 A főorsó fordulatszám parancs (S kód)

S címre egy legfeljebb ötjegyű számot írva az NC egy kódot ad át a PLC-nek. S címet a PLC, az adott szerszám gép felépítésének függvényében értelmezheti kódként, vagy fordulatszám/perc dimenziójú értéként is.

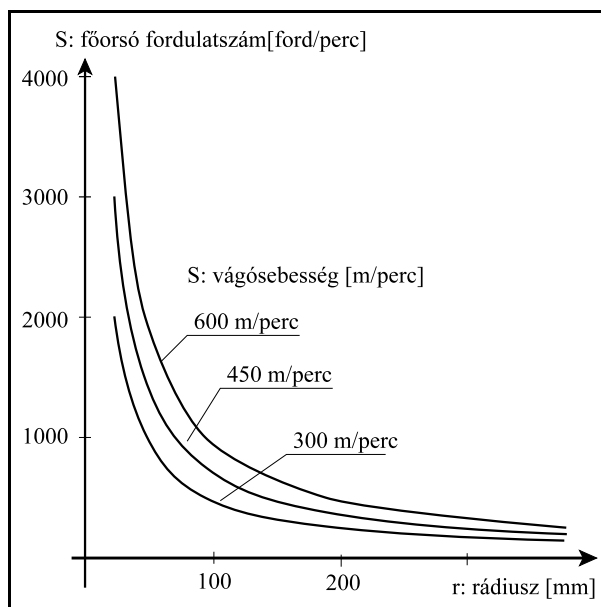
Ha mozgásparancsot és főorsó fordulatszámot (S) programozunk ugyanabba a mondatba az S funkció a mozgásparancs végrehajtása alatt, vagy után kerül végrehajtásra. A végrehajtás mikéntjét a gép építője határozza meg.

Az S címen megadott fordulatszámértékek öröklődnek. Bekapcsolásra a vezérlés S0 kóddal áll fel. A különböző főorsó áttételi tartományokban a főorsó fordulatszámoknak van egy minimális és egy maximális határa. Ezeket a határokat a szerszám gép építője határozza meg a paramétermezőben, és ezen tartományon kívülre nem engedi a vezérlő a fordulatszámot.

10.2 A konstans vágósebesség számítás programozása.

A konstans vágósebesség számítási funkciót csak fokozatmentes főhajtás esetén lehet használni. A vezérlés ekkor a főorsó fordulatszámát úgy változtatja, hogy a szerszámnak a darab felületéhez viszonyított sebessége mindig állandó, és egyenlő a programozott értékkel.

A konstans vágósebesség értékét a bemenő mértékrendszer függvényében az alábbi táblázat alapján kell megadni:



10.2-1 ábra

bemenő mértékegység	konstans vágósebesség mértékegysége
mm (G21 metrikus)	m/min (méter/perc)
inch (G20 inches)	feet/min (láb/perc)

10.2.1 A konstans vágósebesség számítás megadása (G96, G97)

A

G96 S

utasítás bekapcsolja a konstans vágósebesség számítást. S címen a konstans vágósebesség értékét kell megadni a fenti táblázatban megadott mértékegységben.

A

G97 S

utasítás kikapcsolja a konstans vágósebesség számítást. S címen a kívánt főorsó fordulatot lehet megadni (fordulat/perc mértékegységben).

- A konstans vágósebesség számításához a koordinátarendszert úgy kell beállítani, hogy a koordinátarendszer nullpontja egybeessen a forgástengellyel.
- A konstans vágósebesség számítás csak azután hatásos, hogy a főorsót M3-mal, vagy M4-gyel elindítottuk.
- A konstans vágósebesség értéke öröklődik, még azután is, hogy G97 utasítással kikapcsoltuk a számítását. A konstans vágósebesség értékét bekapcsolás után az 1081 CTSURFSP paraméter határozza meg.

G96 S100 (100m/min, vagy 100 láb/min)

G97 S1500 (1500 ford/min)

G96 X260 (100m/min, vagy 100 láb/min)

- A konstans vágósebességszámítás érvényes G94 (előtolás/perc) módban is.
- Ha a konstans vágósebesség számítást kikapcsoltuk G97 paranccsal és nem adtunk meg új főorsó fordulatot akkor a G96 állapotban felvett utolsó főorsófordulat marad érvényben.

G96 S100 (100m/min, vagy 100 láb/min)

⋮

⋮

G97 (a kiadódó X átmérőhöz tartozó fordulatszám)

- Gyorsmeneti pozicionálás esetén (G00 mondat) a konstans vágósebesség nem kerül folyamatosan kiszámításra, hanem a vezérlő a pozicionálás végpontjában esedékes pozícióhoz tartozó fordulatszámot állítja be. Erre azért van szükség, hogy fölöslegesen ne változtassuk a főorsó fordulatszámát.
- A konstans vágósebesség számításához annak a tengelynek a nullpontját, amelynek a pozíciója alapján a főorsó fordulatszámát változtatjuk, a főorsó forgástengelyére kell állítani.

10.2.2 A konstans vágósebesség értékének behatárolása (G92)

A

G92 S

utasítással a konstans vágósebesség számításakor megengedhető legmagasabb főorsó fordulatszámot állíthatjuk be. A vezérlés a konstans vágósebességszámítás bekapcsolt állapotában az itt megadott értéknél nagyobb főorsó fordulatot nem enged kiadni. S mértékegysége ebben az esetben: ford/perc.

- Bekapcsolás után, illetve, ha az S értékét nem határoltuk be G92 paranccsal a főorsó fordulatszám felső határa konstans vágósebességszámítás esetén az adott tartományra megengedhető maximális érték.
- A maximális fordulatszám értéke öröklődik, mindaddig amíg újat nem programozunk, a vezérlés program végére nem fut, vagy üzemmódot nem váltunk.

10.2.3 Tengely kijelölése a konstans vágósebesség számításához

Azt a tengelyt, amelyikről a konstans vágósebességet számítja a vezérlő az 1182 AXIS paraméter jelöli ki. A paraméterre a logikai tengelyszámot kell írni.

Ha a kijelölt tengelytől el akarunk térni, a

G96 P

utasítással adhatjuk meg azt a tengelyt, amelyikről a vágósebességet számítani akarjuk.

A P cím értelmezése:

P1: X, P2: Y, P3: Z,

P4: U, P5: V, P6: W,

P7: A, P8: B, P9: C

- A P címen beállított érték öröklődik. Bekapcsolásra a vezérlés az 1182 AXIS paraméteren beállított tengelyre érvényesíti a konstans vágósebesség számítást.

10.3 A főorsó pozícióvisszacsatolás

Normál megmunkálás esetén az NC a főorsó hajtásának a programozott fordulatszámmal (S címen megadott érték) arányos fordulatszám parancsot ad ki. A főhajtás ekkor fordulatszám szabályzott üzemmódban dolgozik.

Bizonyos technológiai feladatoknál azonban szükség lehet arra, hogy a főorsót meghatározott szöghelyzetbe állítsuk. Ezt nevezük főorsó pozícionálásnak, vagy indexálásnak.

A pozícionáltatás előtt az NC a főhajtást pozíciószabályzott üzemmódba hozza. Ez a gyakorlatban azt jelenti, hogy többé már nem az S kóddal arányos fordulatszám parancsot ad ki az NC, hanem a főorsóra szerelt szöghelyzetadó (jeladó) segítségével méri a főorsó helyzetét és a kívánt szögelfordulás függvényében ad ki parancsot a főhajtásnak, mint a többi szabályzott tengelyen. Ez a pozícióvisszacsatolás.

Ahhoz, hogy egy adott gépen a főorsót pozícionálni lehessen a főorsóra szöghelyzetadót kell szerelni, illetve a főhajtásnak olyannak kell lennie, hogy pozícióvisszacsatolásos üzemmódban is működjön.

10.4 Orientált főorsó megállás.

Főorsó orientálásnak, vagy orientált főorsó megállásnak nevezük azt a funkciót amikor a főorsót adott szöghelyzetben állítjuk meg. Erre például automata szerszámcsere esetén, illetve bizonyos fűróciklusok végrehajtásához lehet szükség. Azt, hogy egy adott gépen lehetséges-e az orientálás a paramétermezőben az *ORIENTI* paraméteren kell közölni a vezérlővel. A főorsó orientálás parancsot M19 funkcióval adjuk ki, de a konkrét szerszám gép függvényében más funkció is kiválthatja. Az orientáció műszakilag kétféleképp történhet.

Ha a főorsó nem csatolható vissza pozíciószabályozásra az orientáció a gépre szerelt helyzetkapcsolóra való főorsó ráfordulás segítségével történhet.

Ha a főorsó visszacsatolható pozíciószabályozásra az M19 parancs hatására a vezérlés a főorsón referenciapontfelvételt hajt végre a többi tengelyhez hasonlóan. Azt hogy keressen-e reftpontkapcsolót, vagy rácsponi zérót vegyen fel a paramétermezőben lehet megadni. A vezérlés automatikusan elvégzi a pozíciószabályzó kör zárását.

10.5 A főorsó pozícionálás (indexálás)

Főorsó pozícionálás csak a főorsó pozíciószabályozó hurok zárása, azaz orientálás után lehetséges. Ez a funkció szolgál tehát a hurokzárásra. A hurok nyitása M3, vagy M4 forgásparancsra történik.

Abban az esetben, ha az *INDEXI* paraméter értéke =1 (ez a paraméter azt jelzi, hogy a főhajtás visszacsatolható pozíciószabályozásra) és az *INDEX_CI* paraméter értéke =0 a főorsó indexálás M funkcióra történik.

Ebben az esetben az *M_NUMBI* paraméteren beállított küszöbértéktől kezdődő (*M_NUMBI*+360)-ig tartó M funkciókat főorsó indexálási parancsnak értelmezi, vagyis a programozott M értékből levonja a küszöbszámot és az így kapott számot, mint inkrementális, fokban megadott elmozdulást kezeli.

Tehát ha például *M_NUMBI*=100 az M160 parancs azt jelenti, hogy a jelenlegi helyzethez képest a főorsó forogjon el 160-100=60 fokot. A mozgás iránya a *CDIRSI* paraméteren kijelölt érték, sebessége pedig a *RAPIDSI* paraméteren beállított érték.

Ha az *INDEX_CI* paraméter értéke =1 a főorsó indexálást C címen lehet megadni.

10.6 A főorsó fordulatszám ingadozás figyelése (G25, G26)

A

G26

utasítás bekapcsolja a főorsó fordulatszám ingadozás figyelését, a

G25

utasítás kikapcsolja azt. Bekapcsolás, vagy RESET után a vezérlés G26 állapotba kerül, vagyis a fordulatszám ingadozás figyelés be van kapcsolva. Ez a funkció a főorsó forgása során bekövetkező abnormalitásokra ad jelzést, amellyel például a főorsó beragadása elkerülhetővé válik.

A fordulatszám ingadozás figyelését 4 paraméter befolyásolja. Ezek a paraméterek programból a G26 kód után következő címekkel átírhatók. Az átírásra került paraméterek megőrződnek kikapcsolásra is. A

G26 Pp Qq Rr Dd

utasítás hatására íródnak át a paraméterek. A paraméterek értelmezését az alábbi táblázat tartalmazza:

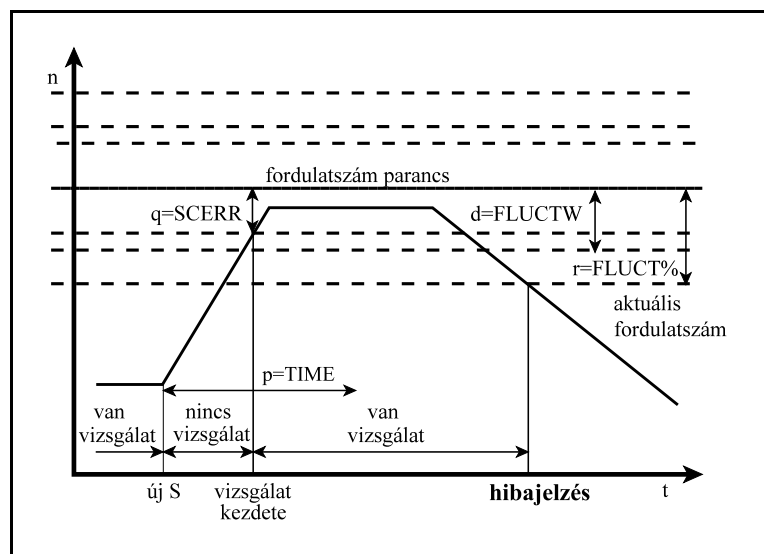
név	paraméter	jelentése	egység	értékhatár
p	5001 TIME	a fordulatszám parancs kiadása és az ellenőrzés megkezdése közötti maximális időtartam	100 msec	65535
q	5002 SCERR	a fordulatszám parancs és a tényleges fordulatszám között megengedhető százalékos eltérés	%	1-50
r	5003 FLUCT%	a fordulatszám ingadozás megengedhető mértéke a kiadott fordulatszám százalékában	%	1-50
d	5004 FLUCTW	a fordulatszám ingadozás megengedhető mértéke abszolút értékben	ford/perc	65535

A fordulatszám ingadozás figyelés a következőképp történik.

A fordulatszám ingadozás figyelésének elkezdése

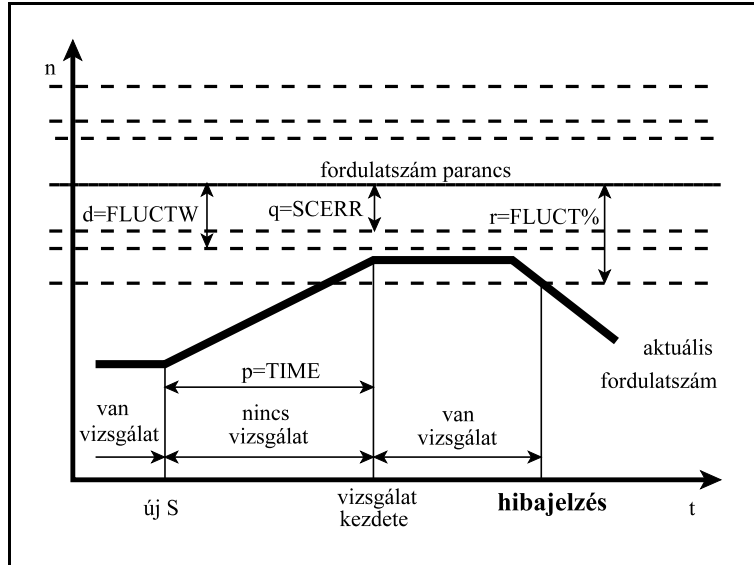
Új fordulatszám parancs hatására a figyelést felfüggeszti a vezérlő. A fordulatszám ingadozás figyelése akkor kezdődik, amikor

– az aktuális főorsó fordulatszám a "q" értéken meghatározott tűrészhatáron belül eléri a fordulatszám parancs értékét, vagy



10.6-1 ábra

– az aktuális főorsó fordulatszám nem éri el a "q" értéken megadott tűréshatáron belül a fordulatszám parancs értékét, de a parancs kiadásától kezdve a "p" értéken meghatározott idő eltelik.



10.6-2 ábra

Hiba detektálása

A figyelés során a vezérlő akkor ad hibajelzést, amikor az aktuális fordulatszám eltérése a fordulatszám parancsétól túllépi

– az "r" értéken megadott, a parancsérték százalékában kifejezett tűréshatárt, és
– a "d" értéken megadott abszolút tűréshatárt is.

Amikor mindkét tűréshatárt átlépte az aktuális fordulatszám értéke, az NC beállítja a PLC felé az I656 jelzöt.

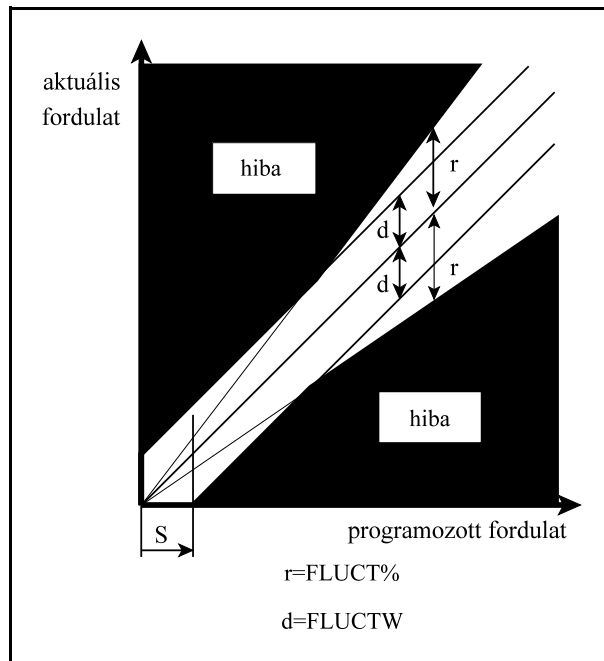
A 3. ábrán látható az a fordulatszám tartomány, amelyre az NC hibajelzést küld. Ha a programozott főorsó fordulat az ábrán látható "S" jelű érték alatt van, akkor ad az NC hibajelzést, ha az aktuális fordulatszám 1 másodpercnél nagyobb ideig 0 ford/perc.

– A fordulatszám ingadozás figyelési funkció csak akkor hatásos, ha a főorsóra jeladó van szerelve.

– A fordulatszám parancs, amihez képest az aktuális fordulatszámot figyeli, az override, a tartományi fordulatszámhatárok, és G96 állandó vágósebességszámítási állapotban a programozott maximális fordulatszám (G92 S_) figyelembe vételével kerül kiszámításra.

– A fordulatszám ingadozás figyelés csak G26 esetén és forgó főorsónál (M3, vagy M4 állapotban) hatásos.

– A G26 parancsot önálló mondatban kell programozni.

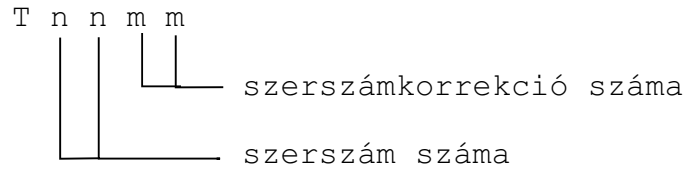


10.6-3 ábra

11 A szerszámkezelés

A T címre írt szám tartalmazza a szerszám számát az első két számjegyen, illetve a szerszámhoz rendelt korrekció számát a második két számjegyen.

A T címre írt kód értelmezése:



A T1236 parancs jelentése: a 12-es számú szerszámot váltsd be és a 36-os számú korrekciós csoportot hívd le mellé.

- T cím programozásakor a vezető nullák elhagyhatók: T101=T0101
- Ha a szerszám számára 0-t programozunk, vagy csak egy, vagy két számjegyet írunk T címre, szerszámváltás nem történik, csak új korrekciót hívunk le. T12 jelentése: a 12-es korrekciós csoportot hívd le.

Ha mozgásparancsot és szerszámszámot (T) programozunk ugyanabba a mondatba a T funkció a mozgásparancs végrehajtása közben, vagy után kerül végrehajtásra. A végrehajtás mikéntjét a gép építője határozza meg.

12 Vegyes és segédfunkciók

12.1 Vegyes funkciók (M kódok)

M cím után egy legfeljebb 3 jegyű számértéket adva az NC a kódot átadja a PLC-nek.

Ha mozgásparancsot és vegyes funkciót (M kódot) programozunk ugyanabba a mondatba a vegyes funkció a mozgásparancs végrehajtásával párhuzamosan, vagy a mozgásparancs végrehajtása után kerül végrehajtásra. A végrehajtás mikéntjét a gép építője határozza meg.

Az M kódok között vannak kijelölt funkciót ellátó kódok amelyek csak meghatározott funkcióra használhatók. Ezek a következők:

M00, M01, M02, M30, M96, M97, M98, M99: programvezérlő kódok

M03, M04, M05, M19: főorsó kezelés kódjai

M07, M08, M09: hűtővíz kezelés kódjai

M11, ..., M18: főorsó tartományváltás kódja

A többi M érték szabad felhasználású.

A főorsó indexálás M kódjai, ha az indexálás M-re működik, paraméter alapján kerülnek kijelölésre.

A vezérlés lehetővé teszi, hogy egy mondatba több, különböző csoportba tartozó M kódot írassunk. A csoportosítás és végrehajtási sorrend a következő:

1. csoport M11, ..., M18: főorsó tartományváltás
2. csoport M03, M04, M05, M19: főorsó kezelés
3. csoport M07, M08, M09: hűtővíz kezelés
4. csoport Mnnn: tetszőleges egyéb M funkció
5. csoport főorsó indexálás M kódjai
6. csoport M00, M01, M02, M30, M96, M97, M98, M99: programvezérlő kódok

Az egy mondatban programozható M funkciók száma maximum 5. Mindegyik csoportból csak egy M kód programozható egy mondatban. Ennek ellentmondó programozás *3032 ELLNTMONDÓ M KÓDOK* hibajelzést eredményez.

Az egyes M kódok pontos működését az adott szerszámgép építője határozza meg a szerszámgép felépítésének függvényében. Ez alól kivételt képeznek a programvezérlő kódok.

A programvezérlő M kódok:

M00: programozott stop

Azon mondat végén, amelyikben az M00 megadásra került stop állapot generálódik. Az összes öröklődő funkció változatlan marad. Újraindítható start hatására.

M01: feltételes stop

Hatása azonos az M00 kód hatásával. Végrehajtásra kerül a FELTÉTELES ÁLLJ gomb bekapcsolt állapotában. Ha a megfelelő gomb nincs bekapcsolva hatástalan.

M02, M30: program vége

A főprogram végét jelenti. A műveletek leállnak, és a vezérlés alaphelyzetbe kerül. A gép alaphelyzetbe hozásáról a PLC program gondoskodik. Minden végrehajtott M02 vagy M30 eggyel növeli a munkadarab-számlálót, hacsak a *PRTCNTM* paraméterrel felül nem bíráljuk ezt a szolgáltatást.

M98: alprogram hívás

Hatására alprogramhívás történik.

M99: alprogram vége

Hatására a végrehajtás visszatér a hívás helyére.

12.2 Segédfunkciók (A, B, C kódok)

A, B vagy C címen legfeljebb három számjegyet adhatunk meg, ha ezeknek a címeknek valamelyike, vagy mindegyike segédfunkciónak van kijelölve a paramétermezőben. A segédfunkción megadott érték a PLC-nek kerül átadásra.

Ha mozgásparancsot és segédfunkciót programozunk ugyanabba a mondatba a segédfunkció a mozgásparancs végrehajtásával párhuzamosan, vagy a mozgásparancs végrehajtása után kerül végrehajtásra.

A végrehajtás sorrendjét a szerszám gép építője dönti el, és a szerszám gép specifikációja tartalmazza. B címen például osztóasztal indexelése valósítható meg.

12.3 A különböző funkciók végrehajtási sorrendje

A különböző, egy mondatba írt funkciókat a vezérlés általában az alábbi sorrendben hajtja végre:

- | | | |
|-----|-------------------------|--|
| 1. | szerszámhívás: | T |
| 2. | főorsó tartományváltás: | M11, ..., M18 |
| 3. | főorsó fordulatszám: | S |
| 4. | főorsó kezelés: | M03, M04, M05, M19 |
| 5. | hűtővíz: | M07, M08, M09 |
| 6. | egyéb M funkció: | Mnnn |
| 7. | főorsó indexálás: | M funkcióval |
| 8. | A funkció: | A |
| 9. | B funkció: | B |
| 10. | C funkció: | C |
| 11. | programvezérlő kódok: | M00, M01, M02, M30, M96, M97, M98, M99 |

Amennyiben a fenti végrehajtási sorrend nem megfelelő, a mondatot több mondatra kell bontani, és az egyes mondatokba a kívánt sorrendnek megfelelően kell beírni a funkciókat.

13 Az alkatrészprogram szervezése

A bevezető részben már láttuk az alkatrészprogram felépítését, hogy a programok milyen kódokkal, és milyen formátumban helyezkednek el a tárban. Ebben a fejezetben az alkatrészprogramok szervezéséről lesz szó.

13.1 A mondatszám (N cím)

A program mondatait sorszámmal láthatjuk el. A mondatok számozása N címen történhet. N címen legfeljebb 5 számjegyen számozhatjuk a mondatokat. N cím használata nem kötelező. Egyes mondatokat beszámozhatunk, másokat nem. A mondatok számozásának nem kötelező egymás utáninak lenni.

13.2 Feltételes mondatkihagyás (/ cím)

Feltételes mondatkihagyást programozhatunk törtvonal / címen. A törtvonal / cím értéke 1-9 lehet. 1-9 számok kapcsoló sorszámként jelentenek. Az 1-es sorszámú kapcsoló a vezérlő funkciógombjai között található a Kezelőpanel, Pozíció és Állapot képekben a **FELTTL MONDAT** felírással.

A többi kapcsoló felszerelése opcionális, a vezérlés interfész felületén keresztül adhatók be a jelei.

Abban az esetben, ha egy mondat elejére /n feltételes mondatkihagyást programozunk, akkor

- ha az n-edik kapcsoló bekapcsolt állapotban van kihagyja a végrehajtásból a mondatot,
- ha az n-edik kapcsoló kikapcsolt állapotban van végrehajtja a mondatot.

Ha azt akarjuk, hogy a feltételes mondat kapcsolót akár a mondat végrehajtása előtti mondatban is figyelembe vegye a vezérlő, állítsuk a 1248 CNDBKBUF paramétert 0-ba. Ekkor a feltételes mondat utasítás (/ jellel kezdődő mondatok) **elnyomja** a mondat előreolvasást. Ebben az esetben **G41, G42** esetén a kontúr **torzul**, viszont a feltételes mondat kapcsolót elég az **előző mondat végrehajtása közben kapcsolni**, hogy hatásos legyen.

Ha azt akarjuk, hogy a / utasítás ne nyomja el a mondat előreolvasást, állítsuk a 1248 CNDBKBUF paramétert 1-be. Ekkor a feltételes mondat utasítás (/ jellel kezdődő mondatok) **nem nyomja el** a mondat előreolvasást. Ebben az esetben **G41, G42** esetén a kontúr **nem torzul**, viszont a feltételes mondat kapcsolót a biztos hatás miatt **a program végrehajtása előtt be kell állítani**.

13.3 Főprogram és alprogram

Kétféle programot különböztetünk meg: főprogramot és alprogramot. Egy alkatrész megmunkálása során adódhatnak ismétlődő tevékenységek, amelyeket ugyanazzal a programrészlettel lehet leírni. Annak érdekében, hogy az ismétlődő részeket ne kelljen többször leírni a programban, ezekből a részekből alprogramot készíthetünk, amelyet az alkatrészprogramból hívhatunk. A fő- és alprogram felépítése teljes egészében megfelel a bevezetőben mondottaknak.

Kettejük között a különbség az, hogy míg a főprogram végrehajtása után a megmunkálás befejeződik, és a vezérlés arra vár, hogy újra elindítsák, az alprogram végrehajtása után a vezérlés visszatér a hívó programba és onnan folytatja a megmunkálást.

Programozástechnikailag a különbség a kétféle program között, a program lezárásából adódik. A főprogram végét M02, vagy M30 kóddal jelezzük (használatuk nem kötelező), az alprogramot pedig M99 kóddal kell kötelezően lezárni.

13.3.1 Az alprogram hívása

Az

M98 P....

utasítássor alprogramhívást generál. Az utasítás hatására a program végrehajtása a P címen meghatározott számú alprogramon folytatódik. P cím értékhatára: 1-9999. Az alprogram végrehajtása után a hívó program alprogramhívást követő mondatán folytatódik a megmunkálás:

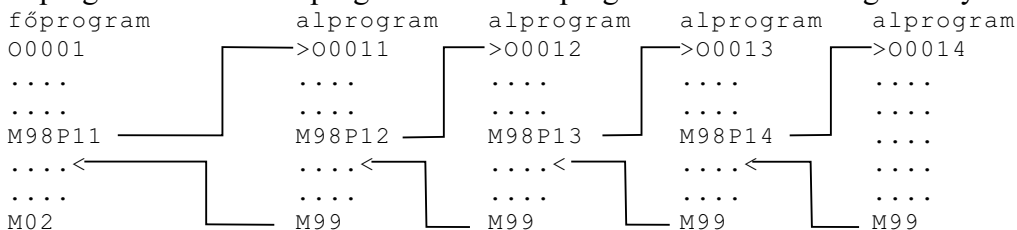
hívó program		alprogram	megjegyzés
O0010			az O0010 program végrehajtása
.....			
.....			
M98 P0011	--->	O0011	az O0011 alprogram hívása
		az O0011 alprogram végrehajtása
		
		
következő mondat	<---	M99	visszatérés a hívó programba
.....			az O0010 program folytatása
.....			

Az

M98 P.... L....

utasítássor az L címen megadott számban hívja egymás után a P címen megadott alprogramot. L cím értékhatára: 1-9999. Ha L-nek nem adunk értéket az alprogram egyszer hívódik meg, azaz L=1-et tételez fel a vezérlő.

Az M98 P11 L6 utasítás azt jelenti, hogy hívod meg az O0011-es alprogramot egymás után 6-szor. Alprogramból is lehet alprogramot hívni. Alprogramhívások 4 szintig skatulyázhatók egymásba.



☞ Megjegyzések:

- 3069 SZINTTÚLLÉPÉS hibajelzést ad, ha az alprogram hívásszint túllépi a 4-et.
- 3071 P HIÁNYZIK, VAGY HIBÁS hibajelzést ad, ha P címértéke nagyobb, mint 9999, vagy nincs megadva.
- 3072 L MEGADÁSI HIBA hibajelzés képződik, ha L értéke hibás.
- 3073 NEM LÉTEZŐ PROGRAMSZÁM hibajelzés képződik, ha a P címen megadott azonosítójú program nincs a tárban.

13.3.2 Visszatérés alprogramból

Az

M99

utasítás alprogramban történő használata az alprogram végét jelenti, és a programvezérlést átadja a hívó program hívást követő mondatára:

hívó program		alprogram	megjegyzés
00010			az 00010 program végrehajtása
.....			
.....			
N101 M98 P0011	--->	00011	az 00011 alprogram hívása
		az 00011 alprogram végrehajtása
		
N102	<---	M99	visszatérés a hívó program következő mondatára
.....			az 00010 program folytatása
.....			

Az

M99 P.....

utasítás alprogramban történő használata az alprogram végét jelenti, és a programvezérlést átadja a hívó program P cím alatt megadott számú mondatára. P cím értékhatára ebben az esetben: 1-99999.

hívó program		alprogram	megjegyzés
00010			az 00010 program végrehajtása
.....			
.....			
N101 M98 P0011	--->	00011	az 00011 alprogram hívása
		az 00011 alprogram végrehajtása
		
N250	<---	M99 P250	visszatérés a hívó program N 2 5 0 mondatára
.....			az 00010 program folytatása
.....			

Az

M99 (P.....) L....

utasítás átírja a hívó program ciklusszámlálóját. Ha L-re 0-t írunk az alprogram csak egyszer kerül meghívásra. Például ha az M98 P11 L20 utasítással hívjuk meg az O11 alprogramot és onnan M99 L5 utasítással térünk vissza az O11 alprogram összesen 6-szor hívódik meg. (L értékhatára: 1-9999)

☞ *Megjegyzés:* – 3070 NEM LÉTEZŐ MONDATSZÁM P hibajelzést ad, ha a visszatérési mondat számot (P) nem találja a hívó programban.

13.3.3 Ugrás főprogramon belül

Az

M99

utasítás főprogramban történő használata feltétel nélküli ugrást eredményez a főprogram első mondatára, és a programvégrehajtást innen folytatja. Az utasítás használata végtelen ciklust ered-

ményez:

```
O0123
N1... <
...
.....
.....
M99
```

Az

M99 P.....

utasítás főprogramban történő használata feltétel nélküli ugrást eredményez a főprogram P cím alatt megadott számú mondatára, és a programvégrehajtást innen folytatja. Az utasítás használata végtelen ciklust eredményezhet:

```
O0011
.....
.....
N128.....<
.....
.....
M99 P128
```

```
O0011
.....
M99 P225
.....
N225 <
.....
```

Az esetleges végtelen ciklusok képződése elkerülhető, ha az M99 utasítást tartalmazó mondatot /1 M99 formában adjuk meg. Ekkor a feltételes mondatkihagyás kapcsoló állásától függően vagy kihagyja az ugrást, vagy nem.

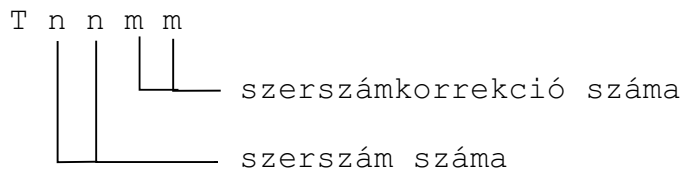
14 A szerszámkorrekció

Ahhoz, hogy az alkatrész programban a különféle szerszámokhoz tartozó kinyúlásértékeket, szerszámcsúgarakat stb. ne kelljen a koordináták megadásánál figyelembe venni, a szerszámjellemzőket egy táblázatban, az ún. korrekciós táblázatban gyűjtjük össze. Ha valahányszor egy szerszámot lehívunk az alkatrész programban, meg kell adni, hogy az adott szerszám adatai hol találhatóak a korrekciós táblázatban. Ezek után a vezérlés már a hivatkozott korrekciók figyelembe vételével vezeti a szerszámot a programozott pályán.

14.1 Hivatkozás szerszámkorrekcióra

Szerszámkorrekcióra a T címre írt szám alsó két számjegyén lehet hivatkozni.

A T címre írt kód értelmezése:



A T1236 parancs jelentése: a 12-es számú szerszámot váltsd be és a 36-os számú korrekciós csoportot hívd le mellé.

- T cím programozásakor a vezető nullák elhagyhatók: T101=T0101
- Ha a szerszám számára 0-t programozunk, vagy csak egy, vagy két számjegyet írunk T címre, szerszámváltás nem történik, csak új korrekciót hívunk le. T12 jelentése: a 12-es korrekciós csoportot hívd le.

A korrekció számmal a szerszámkorrekciós táblázat egy csoportját jelöljük ki a vezérlés számára. Ennek a táblázatnak az elemei a következők:

sorszám	X		Y		Z		R		Q
	geom.	kopás	geom.	kopás	geom.	kopás	geom.	kopás	
1	123.500	-0.234	87.450	-0.129	267.400	-0.036	1	-0.010	3
2									
3									
...									
47									

A szerszámkorrekciós táblázat tartalmazza a szerszám X és Z irányú kinyúlását (opcionálisan az Y irányút is), a szerszámcsúcs sugarát (R), illetve a szerszám állásának kódját (Q).

☞ Figyelem

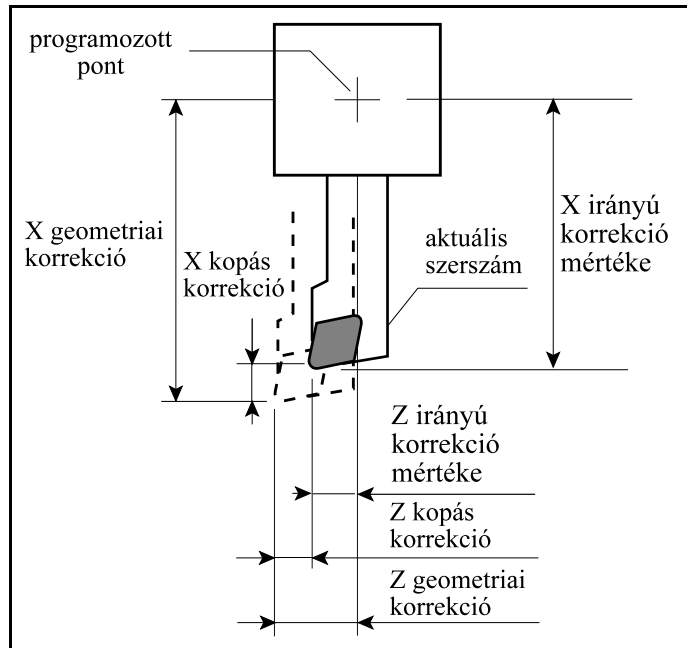
A 00 korrekciószám nem szerepel a táblázatban, az ezen lévő korrekciós értékek mindig nullák, tehát a Tnn00, vagy T0 utasítás a korrekciók törlését jelenti.

Az X, (Y), Z irányú korrekció és a sugárkorrekció (R) két részből tevődik össze: a geometriai és kopásértékből.

Geometriai érték: a bemért szerszám hossza/sugara. Előjeles szám.

Kopás érték: a megmunkálás folyamán fellépő kopások mértéke. Előjeles szám.

Ha programban T címen egy korrekciós csoportra hivatkozunk a vezérlés korrekció gyanánt mindig a geometriai-, és kopásérték összegét veszi figyelembe.

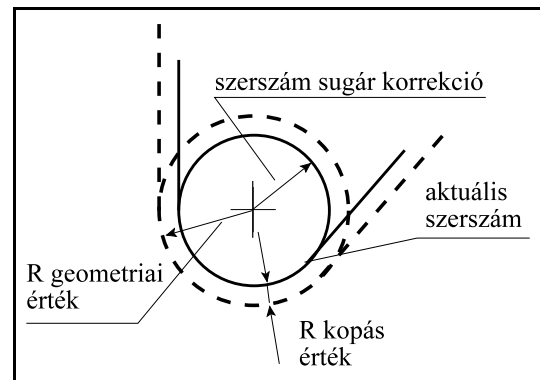


14.1-1 ábra

Például, ha Tnn01-re hivatkozunk a programban, akkor a fenti táblázat alapján az X irányú hosszkorrekció értéke a 01 sor szerint:

$$123.500 + (-0.234) = 123.266$$

Ugyanez vonatkozik a Z (ésY) irányú korrekcióra, és a szerszámsugár korrekcióra (R) is.



14.1 -2 ábra

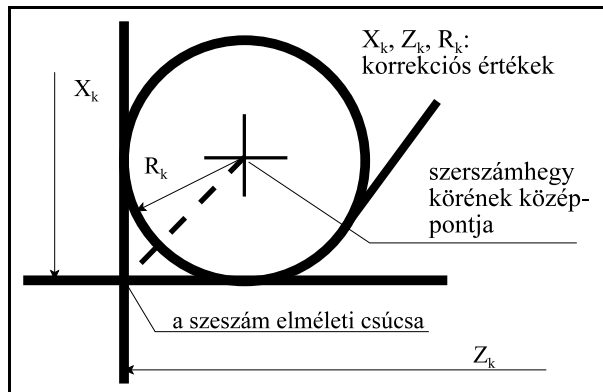
A geometriai és kopás értékek értékhatárai:

bemeneti mérték-rendszer	kimeneti mérték-rendszer	inkremens-rendszer	geometriai érték	kopás érték	dimenzió
mm	mm	IR-A	$\pm 0.01 \div 99999.99$	$\pm 0.01 \div 163.80$	mm
		IR-B	$\pm 0.001 \div 9999.999$	$\pm 0.001 \div 16.380$	
		IR-C	$\pm 0.0001 \div 999.9999$	$\pm 0.0001 \div 1.6380$	
inch	mm	IR-A	$\pm 0.001 \div 9999.999$	$\pm 0.001 \div 6.448$	inch
		IR-B	$\pm 0.0001 \div 999.9999$	$\pm 0.0001 \div 0.6448$	
		IR-C	$\pm 0.00001 \div 99.99999$	$\pm 0.00001 \div 0.06448$	
inch	inch	IR-A	$\pm 0.001 \div 9999.999$	$\pm 0.001 \div 16.380$	inch
		IR-B	$\pm 0.0001 \div 999.9999$	$\pm 0.0001 \div 1.6380$	
		IR-C	$\pm 0.00001 \div 99.99999$	$\pm 0.00001 \div 0.16380$	
mm	inch	IR-A	$\pm 0.01 \div 99999.99$	$\pm 0.01 \div 416.05$	mm
		IR-B	$\pm 0.001 \div 9999.999$	$\pm 0.001 \div 41.605$	
		IR-C	$\pm 0.0001 \div 999.9999$	$\pm 0.0001 \div 4.1605$	

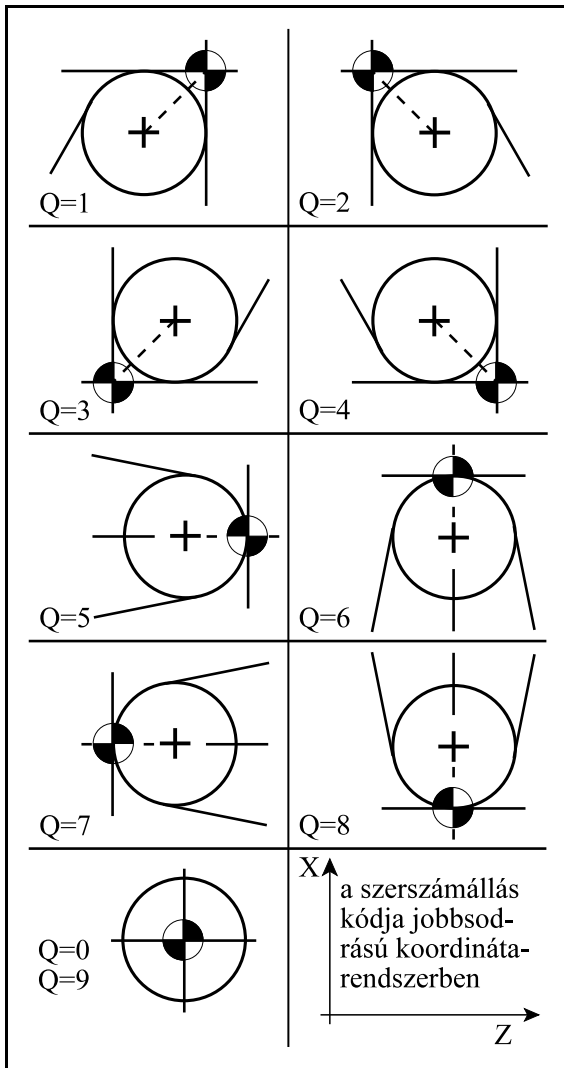
A korrekciós táblázatban meg kell adni a szerszám állásának kódját is. A szerszám állás kódja azt mutatja meg, hogy a szerszámhegy körének középpontjából nézve a szerszám elméleti csúcsa milyen irányban található. A szerszám elméleti csúcsához van az X és Z irányú hosszkorrekció bemérve.

A szerszámállás kódja (Q) egyjegyű szám, értéke 0,1,...9 lehet.

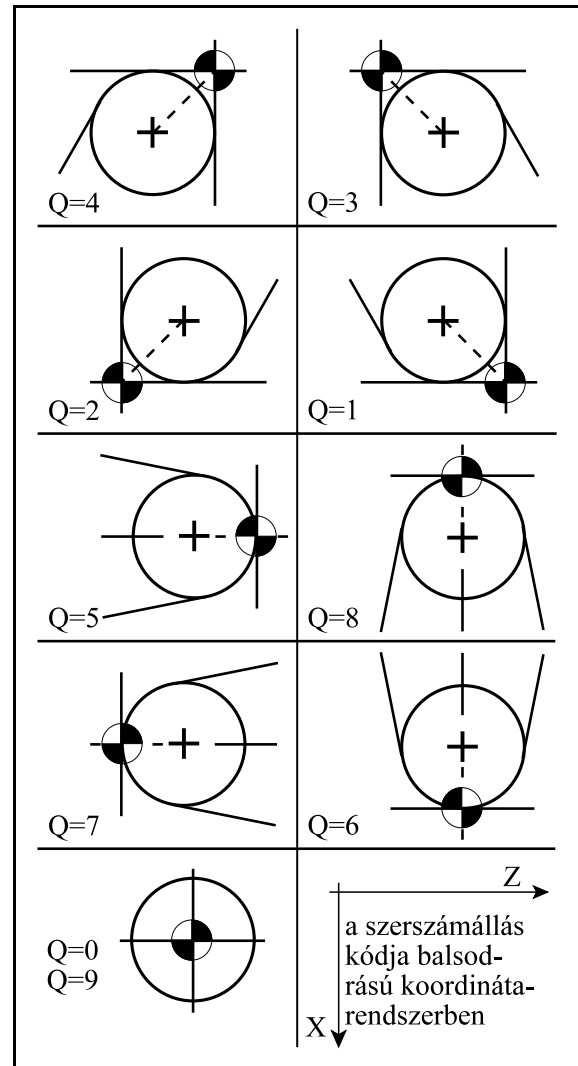
A szerszámállás kódja függ az alkalmazott koordináta-rendszer állásától. Az alábbi két ábra mutatja a jobb-, illetve balsodrású rendszerben alkalmazandó szerszámállás (Q) kódokat.



14.1-3 ábra



14.1-4 ábra



14.1-5 ábra

A lehívott korrekciós kód öröklődik, vagyis a vezérlés mindaddig ugyanazt a korrekciós értéket veszi figyelembe, amíg egy másik T parancsot nem kap, azaz amikor egy T parancssal a korrekciós értéket kiolvastuk, úgy a korrekciós táblázat módosítása (például G10 programozásával) már nincs hatással a kiolvasott értékre.

A korrekciós tábla korrekciós értékei kikapcsolásra megőrződnek.

A korrekciós tárat alkatrészprogramként a háttértárba is elmenthetjük.

A szerszámkorrekciós értékeket be lehet állítani, illetve módosítani a kezelőpanelről adatbevitellel és programból a G10 beállító utasítás használatával. Ha a G10 paranccsal módosítjuk az aktuális korrekciós értéket, akkor ismételt hivatkoznunk kell az aktuális korrekciós regiszterre, mert csak ebben az esetben kerül figyelembevételre a módosított érték.

14.2 Szerszámkorrekciós értékek módosítása programból (G10)

A

G10 L P X Y Z R Q

G10 L P XI YI ZI RI Q vagy

G10 L P U V W C Q

utasítással lehet a szerszámkorrekciók értékeit módosítani programból. A G10 utasítás egyelővetű. A címek és értékeik jelentése:

L=10: geometriai érték beírása

L=11: kopás érték beírása

P címen adjuk meg a korrekciós csoport számát.

X, Y, Z, R címen visszük be a hossz- és sugárkorrekció értékét abszolút adatmegadásnál (G90 állapotban). Ha inkrementálisan akarjuk a korrekció értékét változtatni, kapcsoljunk G91-be, vagy használjuk a megfelelő cím után az **I** inkrementális operátort, vagy X, Y, Z, R címek helyett az **U, V, W, C** címeket. Inkrementális adatmegadásnál a korrekció pillanatnyi értékéhez előjelhelyesen hozzáadódik a megadott érték.

Q címen a szerszámállás kódját adhatjuk meg (Q=0...9 értéktartományban)

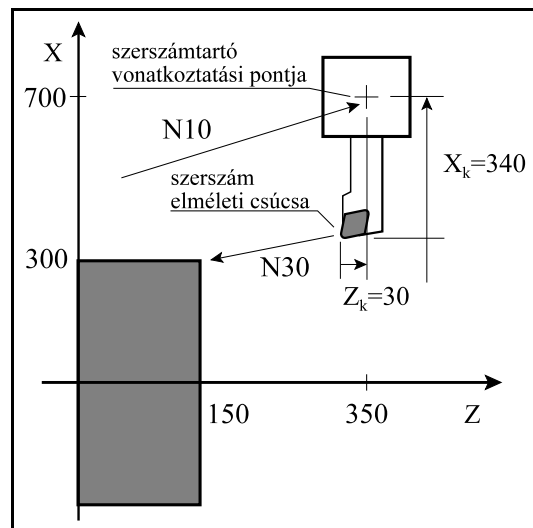
14.3 A szerszámhossz–korrekció figyelembe vétele

A T kód hatására a koordináta-rendszer eltolódik a T kódban megadott korrekciós csoporthoz tartozó X_k , Y_k , Z_k korrekciós értékkel. Ez azt jelenti, hogy a pillanatnyi X, Y, Z pozícióból levonásra kerül a kiválasztott korrekciós csoporthoz tartozó X_k , Y_k , Z_k irányú korrekciós érték. Ettől kezdve nem a szerszám-tartó vonatkoztatási pontjának, hanem a szerszám elméleti csúcsának koordinátáit látjuk a kijelzőn. Korrekciós értéken a továbbiakban mindig a geometriai-, és kopáskorrekció összegét értjük.

Például a

```
(T0000)
N10 G0 (G90) X700 Z350
N20 T202
N30 X300 Z150
```

utasítássor hatására az N10 mondatban a szerszám-tartó vonatkoztatási pontját az X700; Z350 koordinátájú pontra vezeti. Az N20 mondatban beváltja a szerszámot és levonja a pozícióból a 2-es korrekciós csoportba tartozó $X_k=340$, $Z_k=30$ korrekciós értéket. Ekkor a pozíciókijelző az $X=700-340=360$; $Z=350-30=320$ értékre vált. Az N20 mondatban *semmilyen mozgás nem történik*. Az N30 mondatban már a szerszám elméleti csúcsát vezeti a programozott X300; Z150 pontra, vagyis az elmozdulás $X=300-360=-60$; $Z=150-320=-170$ lesz.



14.3-1 ábra

Ha a T kódot mozgásmondattal együtt adjuk meg, a mondatvégi pozícióra már a T kóddal meghatározott új korrekciós érték figyelembe vételével áll rá. A szerszámcsere viszont, ha a T kód szerszámváltás parancsot is tartalmaz, a mozgással párhuzamosan, vagy a mozgás végén hajtódik végre. (A szerszámcsere időpontját a gép építője határozza meg.)

Nézzük a fenti példát:

```
(T0000)
```

```
N10 G0 (G90) X700 Z350
N20 X300 Z150 T202
```

Most az N30 mondatot elhagytuk és az N20 és N30 mondat parancsait összevontuk. Az N20 mondatban a mozgás során már a szerszám elméleti csúcsát vezeti az $X=300, Z=150$ koordinátájú pontra, mint az előző példában az N30 mondat esetén. A szerszámcsere viszont vagy a mozgás közben, vagy a mozgás végén történik. Ez utóbbi esetben viszont könnyen ütközhetünk, ha pl. az előző szerszám kinyúlása hosszabb volt, mint az új szerszámé, és a szerszámcsere a mondat végén hajtja végre a gép. A fentiekből látszik, hogy T kódot mozgással együtt programozni csak új korrekció lehívására célszerű:

```
N10 G0 (G90) X700 Z350
N20 X300 Z150 T02
```

A fenti példában szerszámváltás nem történik, csak a 2-es korrekciót hívtuk le az N20 mondatban.

Figyelem!

A fent elmondottakból látszik, hogy amennyiben a T parancs szerszámváltást, pl revolverfej forgatást is eredményez, a T kódot célszerű önálló mondatba programozni. Ha a T parancs csak korrekcióváltást végez nyugodtan programozhatjuk a T kódot mozgásmondatba is.

A hosszkorrekció törlése

A 0-ás számú korrekciós csoportnak kitüntetett szerepe van., és a korrekció törlését jelenti.

A

Tnn00, vagy a

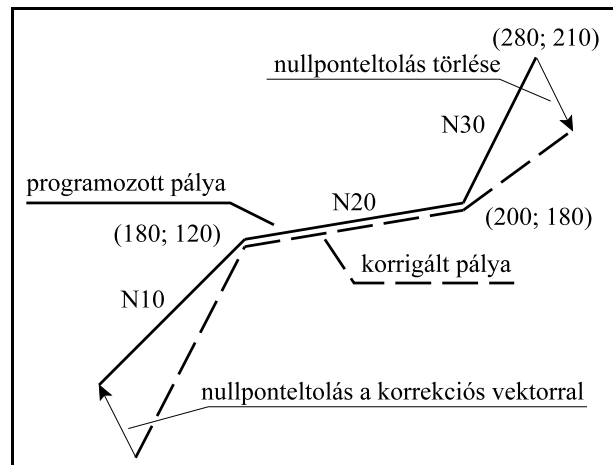
T0

utasítás hatására törlődik a hosszkorrekció (nn: tetszőleges szerszámszám). A folyamat pont fordított, mint a korrekció lehívása. Ez azt jelenti, hogy a szerszám elméleti csúcsának X, Y, Z koordinátájához hozzáadódnak az előzőleg érvényben volt X_k, Y_k, Z_k korrekciós értékek, és a továbbiakban a szerszám tartó vonatkoztatási pontjának koordinátáit látjuk a kijelzőn. Ha a korrekció törlése mozgásmondattal együtt történik a mondat programozott végpontjára a szerszám tartó vonatkoztatási pontját küldi.

Például a

```
N10 X180 Z120 T202
N20 X200 Z180
N30 X280 Z210 T200
```

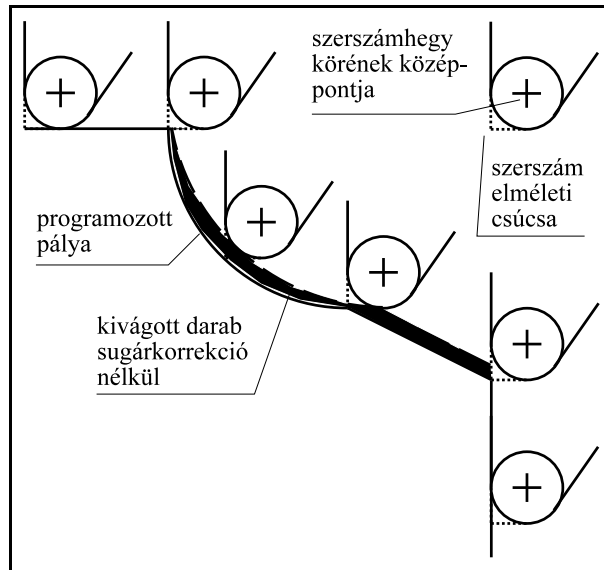
mintapéldában az N10 mondatban a lehívott korrekcióval korrigálja a mozgást a programozotthoz képest, az N30 mondatban pedig törli a korrekciót, vagyis az N10 mondatban beállított nullponteltolást.



14.3-2 ábra

14.4 A szerszámsugár korrekció (G38, G39, G40, G41, G42)

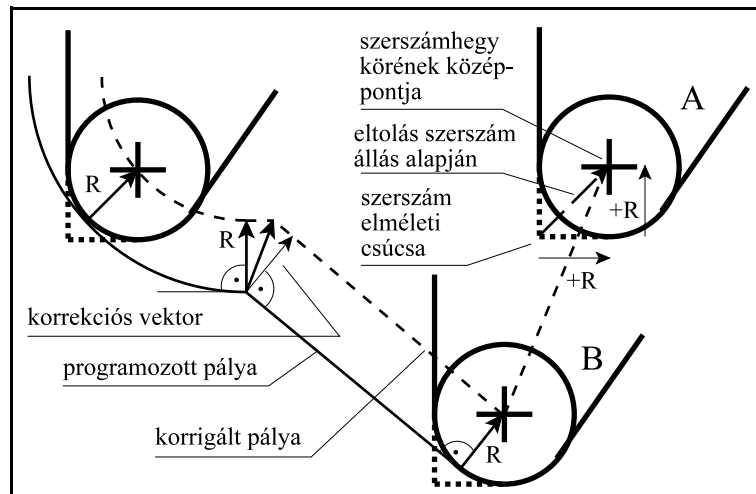
Abban az esetben, ha csak szerszámhossz korrekciót használunk, kúpos felületet, vagy körívet pontosan esztergálni nem tudunk. A szerszámhossz korrekció használata esetén a vezérlés a szerszám elméleti csúcsát vezeti a programozott pályán. Mivel minden szerszám csúcsa kisebb nagyobb lekerekítéssel rendelkezik, ezért a forgácsolt felület csak a tengelyekkel párhuzamos mozgásoknál lesz pontos, amint az a mellékelt ábrán látható. Kúpos, vagy íves felületeknél a darab átmérője minden ponton nagyobb lesz a szándékoltnál. Az eltérést az ábrán a vonalkázott terület mutatja.



14.4-1 ábra

Ahhoz, hogy egy tetszőleges alakzatot pontosan lehessen esztergálni, és az alakzatnak a rajz szerinti pontjait kelljen a programban megadni, függetlenül az alkalmazott szerszámhegy körének sugarától, a vezérlésnek a szerszámhegy körének középpontját a programozott kontúrral párhuzamosan, attól sugárnyi távolságra kell vezetnie. A vezérlés a T címen lehvívott korrekciószámom bejegyzett szerszámsugár korrekció értékében függvényében állapítja meg, hogy a szerszámközep-pont pályáját milyen távolságra vezesse a programozott kontúrtól.

A kontúrra való ráállás előtt a szerszám elméleti csúcsának pozícióját tartja nyilván a vezérlő. Azt, hogy az elméleti csúcsához képest milyen irányban található a szerszámhegy körének középpontja szükség van a szerszám állás kódjára, amit a korrekciós táblázatban Q címen adtunk meg. A szerszámállás kódja alapján az elméleti csúcs pozíciójához X és Z irányban hozzáad, vagy kivon egy szerszámsugárnyi értéket (R). (Ábrán A helyzet.) Ezután a szerszámhegy körének középpontját vezeti rá a programozott pálya kezdőpontjára merőlegesen állított R hosszúságú vektor, a korrekciós vektor, végpontjára. (Ábrán B helyzet.) Ezután a programozott pályával párhuzamosan, attól R távolságra vezeti a szerszámhegy körének középpontját.



14.4-2 ábra

A korrekciós vektor egy olyan síkbeli vektor, amit a vezérlő minden mondatban újraszámol, és a programozott elmozdulásokat a mondat eleji és végi korrekciós vektorokkal módosítja. A kiadódó korrekciós vektorok hossza és iránya a T címen lehvívott korrekciós értéktől és a két mon-

dat közti átmenet geometriájától függ.

A korrekciós vektorokat a G17, G18, G19 utasítások által kiválasztott síkban számolja. Ez a szerszámugár korrekció síkja. Ezen síkon kívüli mozgásokat a sugárkorrekció nem befolyásolja. Például: ha G18 állapotban X, Z sík van kiválasztva, akkor a korrekciós vektorok az X, Z síkban kerülnek kiszámításra.

Szerszámugár korrekció számítása közben a korrekciós sík váltása nem megengedett. Ha mégis megkíséreljük, akkor *3010 SÍKVÁLTÁS G41, G42 ALATT* üzenetet ad a vezérlő.

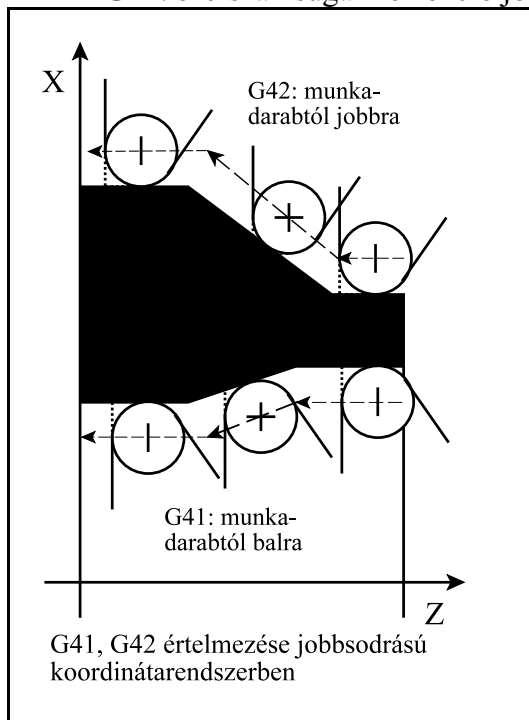
Abban az esetben, ha nem a fősíkba eső tengelyek mentén akarunk korrekciós síkot definiálni a melléktengelyeket a paramétermezőben párhuzamos tengelyekként kell definiálni. Például, ha U párhuzamos tengelynek van felvéve, és a Z, U, síkban akarjuk a szerszámugár korrekciót alkalmazni G18 U__ Z__ megadással lehet a síkot kijelölni.

A szerszámugár korrekció számítását programból be, és ki lehet kapcsolni:

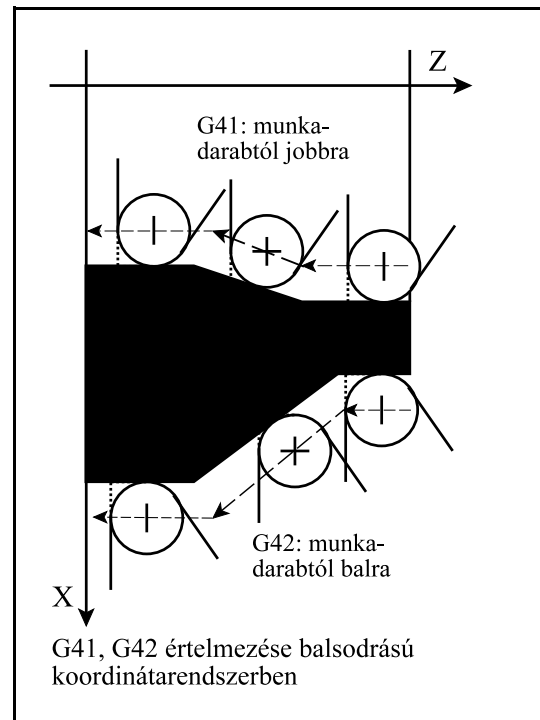
G40: szerszámugár korrekció kikapcsolása

G41: szerszámugár korrekció balról

G42: szerszámugár korrekció jobbról



14.4-3 ábra



14.4-4 ábra

A G41, vagy G42 parancs a korrekciószámítást bekapcsolja. G41 állapotban a programozott kontúr a menetirány szerint balról, G42 állapotban pedig jobbról követi jobbsodrású rendszerben, fordítva a balsodrású rendszerben. Az alkalmazott szerszámugár korrekciós értéket a T című alsó két számjegyen kell megadni. Tnn00, vagy T0 megadása mindig nulla sugárérték lehívásával egyenlő. A korrekciószámítás a G00, G01, G02, G03 interpolációs mozgásokra történik.

Az eddig elmondottak pozitív szerszámsugár korrekció megadásakor érvényesek. A szerszámsugár korrekció értéke viszont negatív is lehet. Ennek gyakorlati értelme akkor van, ha például ugyanazzal az alprogrammal akarunk egy anya, majd egy ehhez illeszkedő apa munkadarabot körbejárni. Ezt úgy is meg lehet oldani, hogy G41-gyel forgácsoljuk például az anyát, és G42-vel az apát. Nem kell ezt a váltást azonban beszerkeszteni a programba, ha az anyadarabot például pozitív, az apadarabot pedig negatív sugárkorrekcióval munkáljuk meg. Ekkor a szerszámközpont pályája a programozott G41, vagy G42-vel ellentétesre vált:

	sugárkorrekció: pozitív	sugárkorrekció: negatív
G41	balról	jobbról
G42	jobbról	balról

☞ **Megjegyzés:**

– A további leírásokban és ábrákban az egyszerűség kedvéért mindig pozitív sugárkorrekcióval dolgozunk.

G40 vagy Tnn00, T0 parancs kikapcsolja a korrekciószámítást. A két parancs közti különbség, hogy a T0 utasítás csak 0 hosszúságú korrekciós vektorral számol és a G41 vagy G42 állapotot változtatlanul hagyja. Ha ezek után új Tnnmm (mm≠0) címre történik hivatkozás a G41 vagy a G42 állapot függvényében az új szerszámsugárral kerül kiszámításra a korrekciós vektor.

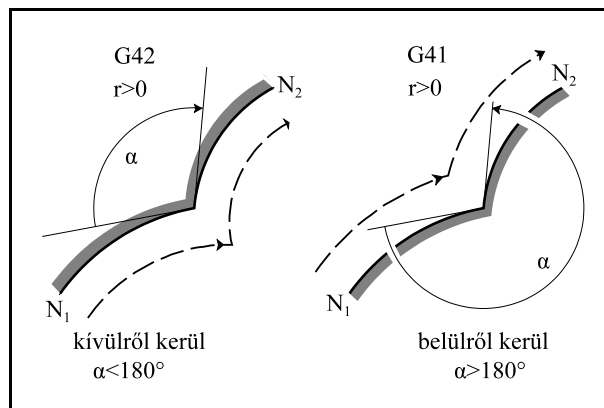
Ha viszont G40 utasítást használunk, a korrekciós vektorokat nem számolja tovább a vezérlő. A sugárkorrekció bekapcsolásának, illetve kikapcsolásának, meghatározott szabályai vannak, amit a következő fejezetek tárgyalnak részletesen.

G40, G41, G42 parancsok öröklődnek. Bekapcsolás után, program végén, vagy a program elejére történő resetelés hatására a vezérlés a G40 állapotot veszi fel, a sugárkorrekciós vektorok törlődnek.

A sugárkorrekciós utasításokat csak automata üzemmódban hajtja végre a vezérlés. Kézi üzemmódban egyedi mondatokon nem határos. Ennek oka a következő. Ahhoz, hogy egy mondat végpontjában a korrekciós vektort ki tudja számítani a vezérlő a következő, a kiválasztott síba eső mozgást tartalmazó mondatot is be kell olvasnia. A két mondat közötti átmenet függvénye a korrekciós vektor. Ebből látható, hogy a korrekciós vektor számításához több mondat előfeldolgozására van szükség.

Mielőtt a korrekciószámítás részleteinek tárgyalásába fognánk bele, be kell vezetni egy segédadatot. Két szakasz, azaz két mondat metszéspontjában a két görbéhez húzott érintők által bezárt szöget: α -t. α iránya attól függ, hogy a kontúrt balról, vagy jobbról járjuk körül.

A vezérlés az α szög függvényében választja ki a metszéspontoknál a fordulási stratégiát. Ha $\alpha > 180^\circ$, azaz belül dolgozik a szerszám, a két szakasz között metszéspontot számít. Ha $\alpha < 180^\circ$, azaz a szerszám kívülről kerül, akkor további egyenes szakaszokat *iktathat be* a ke-



14.4-5 ábra

14.4.1 A sugárkorrekció számítás bekapcsolása. Ráállítás a kontúrra.

A vezérlés bekapcsolás, program vége hatására, vagy a program elejére történő resetelés hatására G40 állapotot vesz fel. A sugárkorrekciós vektor törlődik, és a szerszám elméleti csúcának pályája egybeesik a programozott pályával.

G40 állapotból G41, vagy G42 utasítás hatására a vezérlő belép a sugárkorrekció számítási üzemmódba. A G41 vagy G42 állapotot csak egyenes interpolációt (G00, vagy G01) tartalmazó mondatban veszi fel. Ha körmondatban (G02, G03) akarjuk a korrekciószámítást bekapcsolni a vezérlés 3043 G2, G3 ALATT G41, G42 hibajelzést ad. A kontúrra való ráállítás stratégiáját csak akkor választja a vezérlő, ha G40 állapotból G41, vagy G42 állapotba kapcsolunk. Másképp fogalmazva, ha T00-lal töröljük a korrekciót és utána Tnn-nel visszakapcsoljuk (nn 0-tól különböző szám), nem a kontúrra való ráállítás stratégiáját választja a vezérlő.

A korrekció bekapcsolásának alapesetei α szög és a lehetséges átmenetek: egyenes–egyenes, egyenes–kör függvényében alább láthatók. Az ábrák G42 esetre vannak felrajzolva, pozitív sugárkorrekciót tételezve föl.

☞ *Megjegyzés:* Az ábrák jelöléseinek jelentése most, és a továbbiakban:

r: a sugárkorrekció értéke,

L: egyenes szakasz,

C: körív,

S: mondatonkénti üzemmódban a megállás helye,

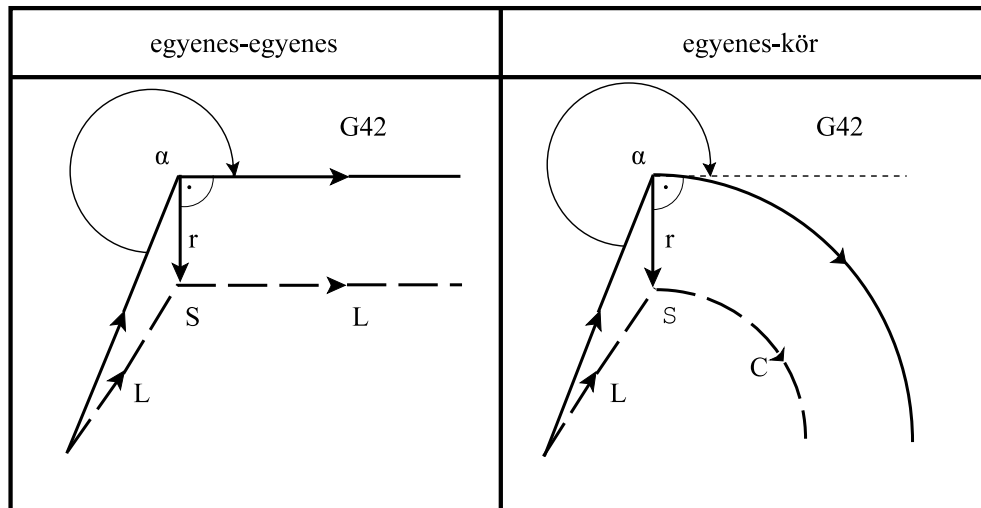
szaggatott vonal: a szerszámközéppont pályája,

folymatos vonal: a programozott pálya.

A sugárkorrekció bekapcsolásának alapesetei:

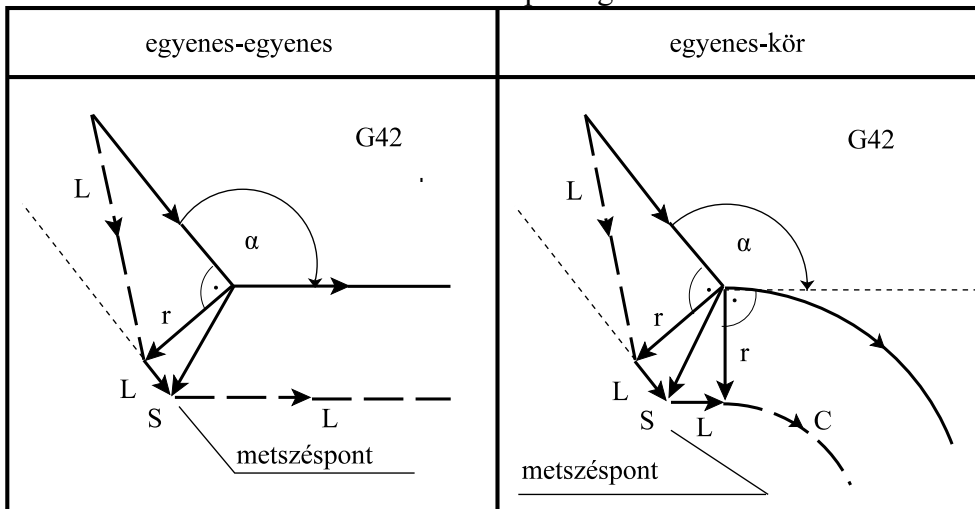
(G40) G42 G01 X_ Z_ X_ Z_	(G40) G42 G01 X_ Z_ G2 X_ Z_ R_
---------------------------------	---------------------------------------

Belső sarokra való ráállítás: $180^\circ < \alpha < 360^\circ$



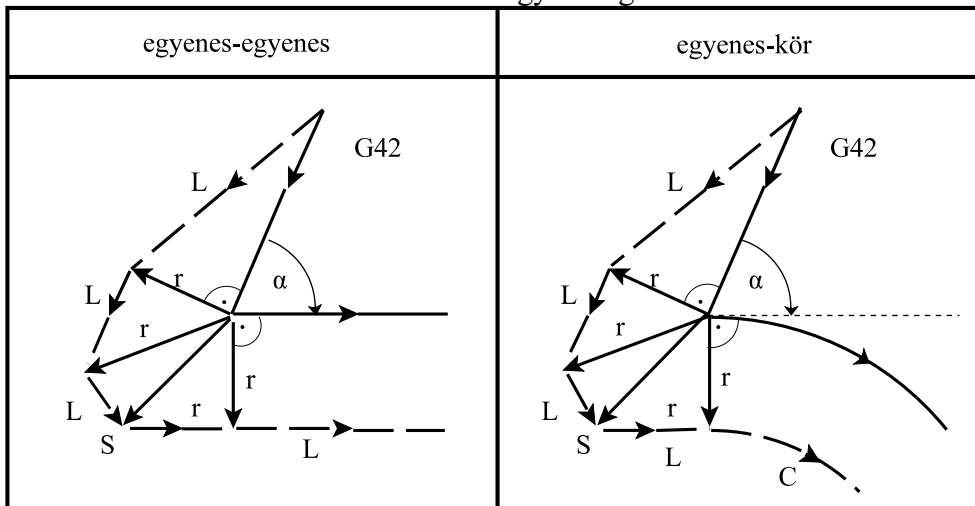
14.4.1-1 ábra

Külső sarokra való ráállás tompaszög alatt: $90^\circ \leq \alpha \leq 180^\circ$



14.4.1-2 ábra

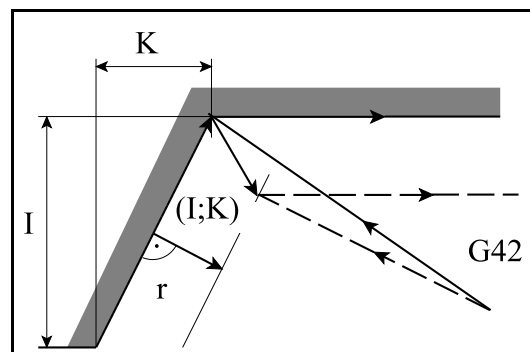
Külső sarokra való ráállás hegyesszög alatt: $0^\circ \leq \alpha < 90^\circ$



14.4.1-3 ábra

A sugárkorrekció bekapcsolásának speciális esetei:

Ha a korrekció bekapcsolását végző mondatban (G41, vagy G42) I, J, K-nak értéket adunk, de csak a kiválasztott síkban lévőeknek (például: G18 esetén I, K-nak), akkor a következő mondat és az I, J, K által meghatározott egyenes közti metszéspontra áll a vezérlő, a sugárkorrekció figyelembe vételével. I, J, K értéke mindig inkrementális, és az általuk megadott vektor annak a mondatnak a végpontjára mutat, amelyikben programoztuk. Ez a lehetőség például belső sarokra való ráállás esetén hasznos.



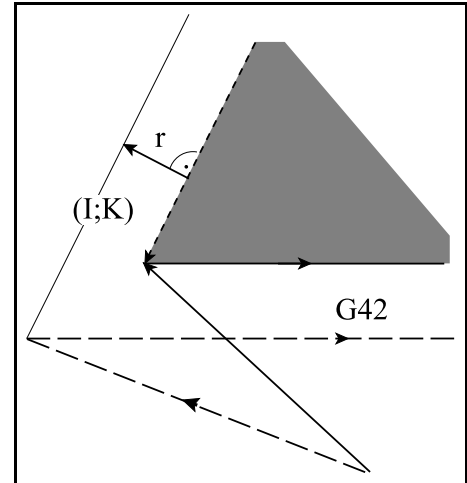
14.4.1-4 ábra


```

...
G91 G18 G40
...
N110 G42 G1 X120 Z-80 I70 K50
N120 Z100
...

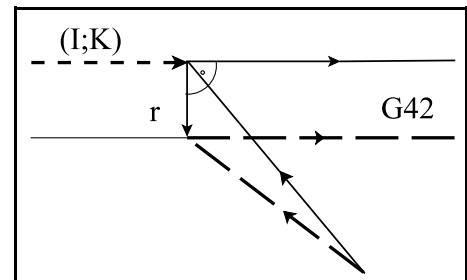
```

Ebben az esetben a vezérlés mindig metszéspontot számol, függetlenül attól, hogy belső, vagy külső sarkot munkálunk meg.



14.4.1-5 ábra

Ha nem talál metszéspontot, a következő mondat kezdőpontjára merőlegesen áll rá.



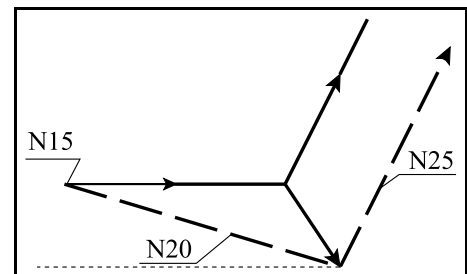
14.4.1-6 ábra

Ha a korrekció bekapcsolását külön mondatban végezzük, ahol a kiválasztott síkban mozgást nem programozunk, akkor a korrekció bekapcsolódása mozgás nélkül megy végbe, a kiszámított korrekciós vektor 0 hosszúságú. A következő mozgásmondat végén a korrekciós vektor a sugárkorrekciószámítás bekapcsolt állapotának megfelelő stratégia szerint (lásd következő pont) számítódik ki:

```

...
N10 G40 G18 G0 X0 Z0
N15 G42
N20 G1 Z80
N25 X120 Z110
...

```



14.4.1-7 ábra

Ha a korrekció bekapcsolását (G41, G42) tartalmazó mondatban nulla elmozdulást programozunk, vagy nulla elmozdulás adódik ki, akkor a vezérlő nem végez semmi mozgást, hanem a fent említett stratégia szerint folytatja a megmunkálást.

```

...

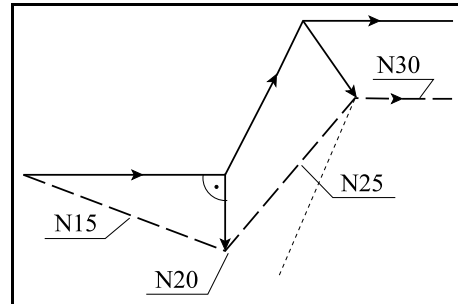
```

14 A szerszámkorrekció

```
N10 G40 G18 G0 X0 Z0  
N15 G91 G42 Z0  
N20 G1 Z80  
N25 X120 Z30  
...
```

Ha a korrekció bekapcsolását követő mondatban a kiválasztott síkban 0 elmozdulás adódik, a korrekciós vektort a bekapcsolást végző mondatra merőlegesen állítja. Az utána következő mondatban a szerszám pályája nem lesz párhuzamos a programozott kontúrral:

```
...  
N10 G40 G18 G0 X0 Z0  
N15 G91 G42 Z80  
N20 G1 Z0  
N25 X120 Z30  
N30 Z60  
...
```

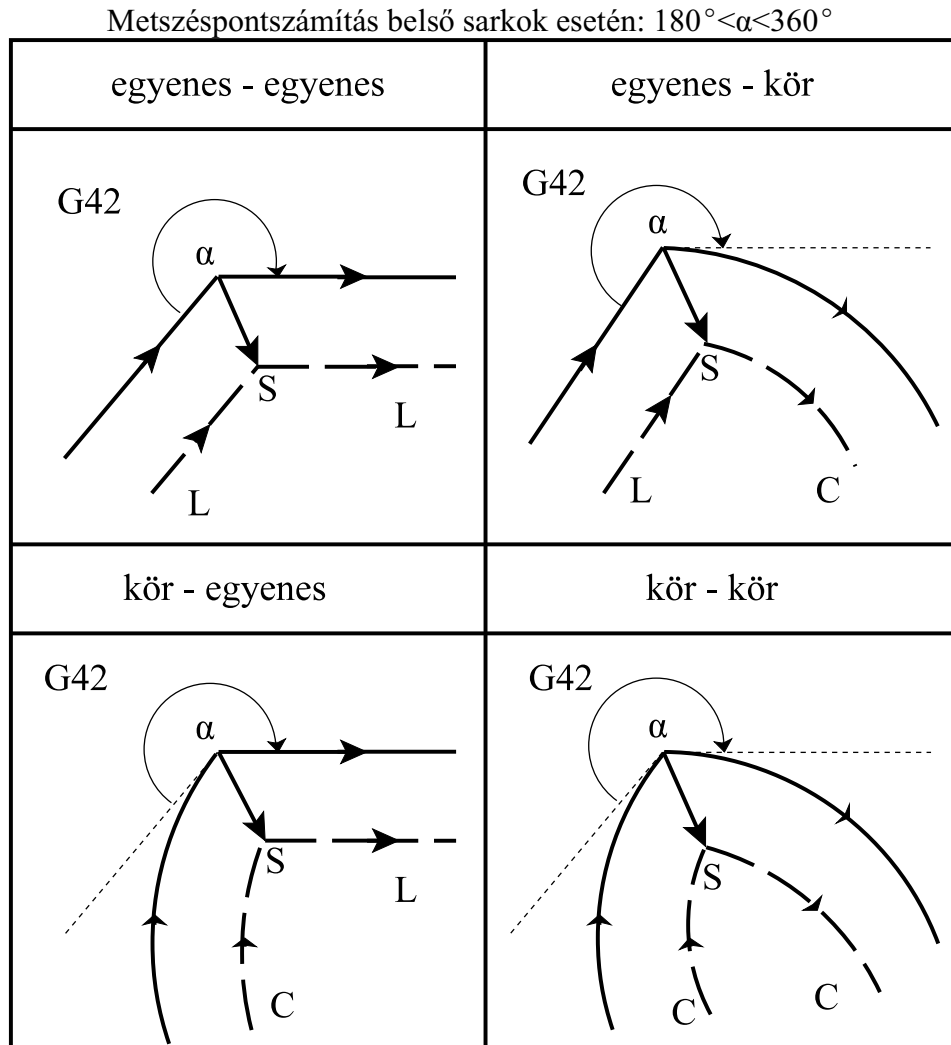


14.4.1-8 ábra

14.4.2 A sugárkorrekció számítás bekapcsolt állapot. Haladás a kontúron.

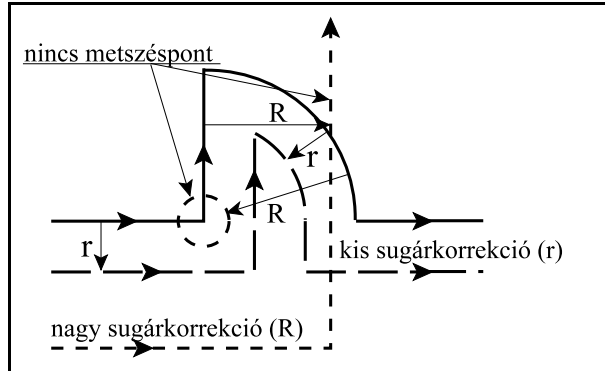
A korrekciószámítás bekapcsolt állapotában a korrekciós vektorok folyamatosan kiszámításra kerülnek az alapeseteknek megfelelően a G00, G01, G02, G03 mondatok között, amíg egynél több olyan mondat nem iktatódik közbe, amelyik nem tartalmaz a kiválasztott síkban elmozdulást. Ilyenek közé soroljuk a várakozást, illetve a tisztán funkciót tartalmazó mondatot is.

A sugárkorrekció számítás bekapcsolt állapotának alapesetei:



14.4.2-1 ábra

Előfordulhat, hogy bizonyos szerszámsugár értékeknél nem adódik metszéspont. Ekkor a vezérlés az előző mondat végrehajtása alatt megáll és 3046 NINCS METSZÉSPONT G41, G42 hibajelzést ad.



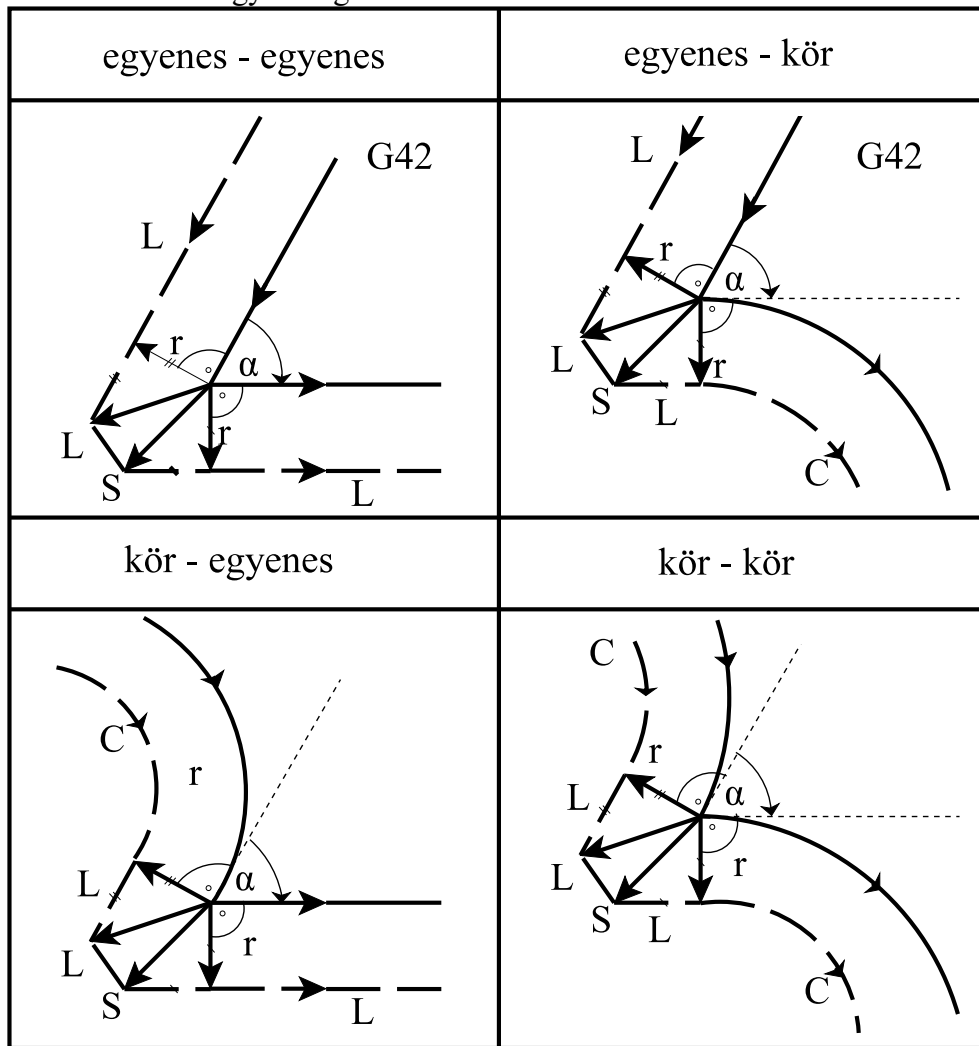
14.4.2-2 ábra

Tomposzögű külső sarkok kerülése: $90^\circ \leq \alpha \leq 180^\circ$

egyenes - egyenes	egyenes - kör
<p>G42</p>	<p>G42</p>
kör - egyenes	kör - kör
<p>G42</p>	<p>G42</p>

14.4.2-3 ábra

Hegyszögű külső sarkok kerülése: $0^\circ \leq \alpha < 90^\circ$



14.4.2-4 ábra

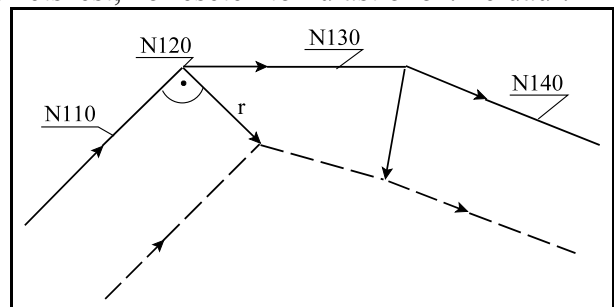
A sugárkorrekció számítás bekapcsolt állapotának speciális esetei:

Abban az esetben, ha G41, vagy G42 bekapcsolt állapotában a kiválasztott síkban az egyik mondatban nulla elmozdulást programozunk, vagy nulla elmozdulás adódik, az előző mondat végpontjára állít egy merőleges vektort, amelynek hossza megegyezik a sugárkorrekcióval. Az ilyen esetekre vigyázni kell, mert szándékolatlan alámetszést, kör esetén torzulást okoz. Például:

```

...G91 G18 G42...
N110 G1 X100 Z40
N120 Z0
N130 Z90
N140 X-40 Z50
...

```



14.4.2-5 ábra

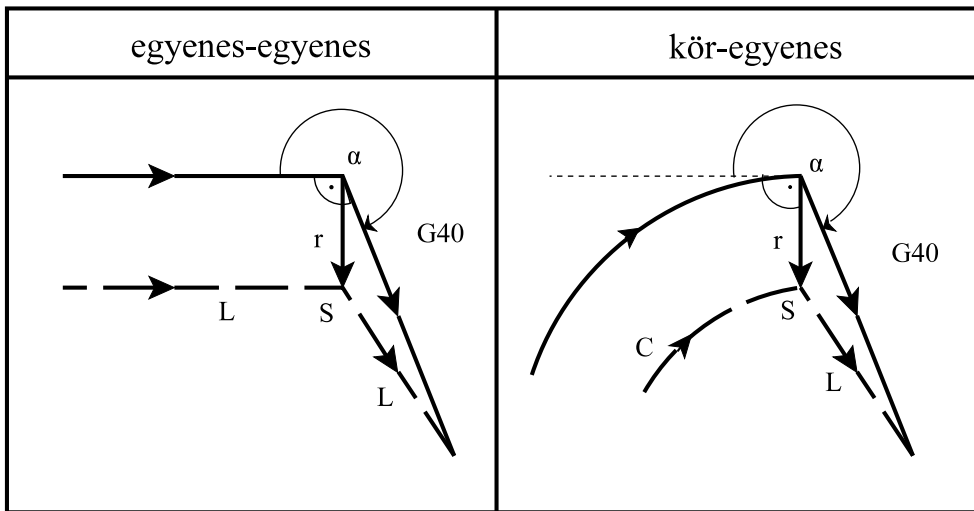
14.4.3 A szerszámsugár korrekciószámítás kikapcsolása. Leállítás a kontúrról.

A G40 parancs kikapcsolja a szerszámsugár korrekciószámítást. G40 parancsot csak lineáris interpolációval lehet kiadni. Ha körmondattban programozunk G40-et *3042 G2, G3 ALATT G40* hibajelzést ad a vezérlő.

A sugárkorrekció kikapcsolásának alapesetei:

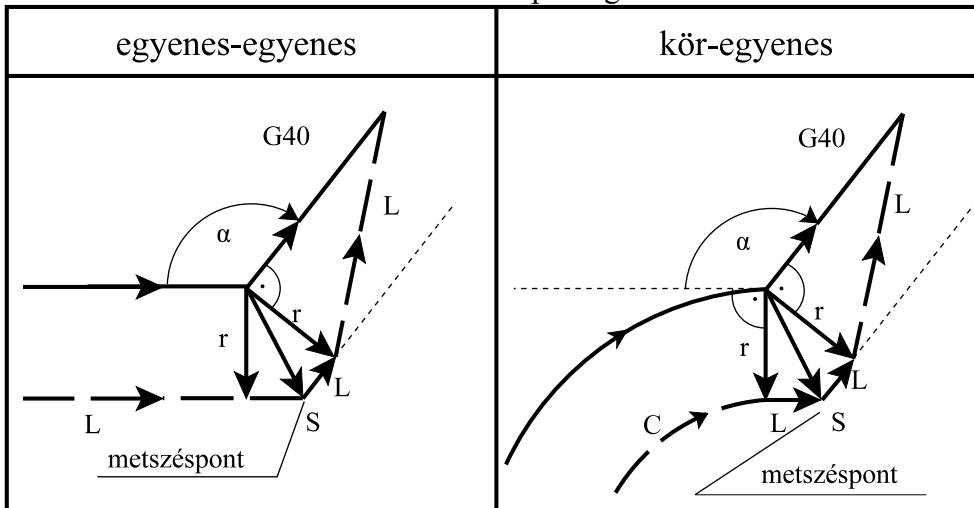
(G42)	(G42)
G01 X_ Z_	G02 X_ Z_ R_
G40 X_ Z_	G40 G1 X_ Z_

Belső sarokból való kiállás: $180^\circ < \alpha < 360^\circ$

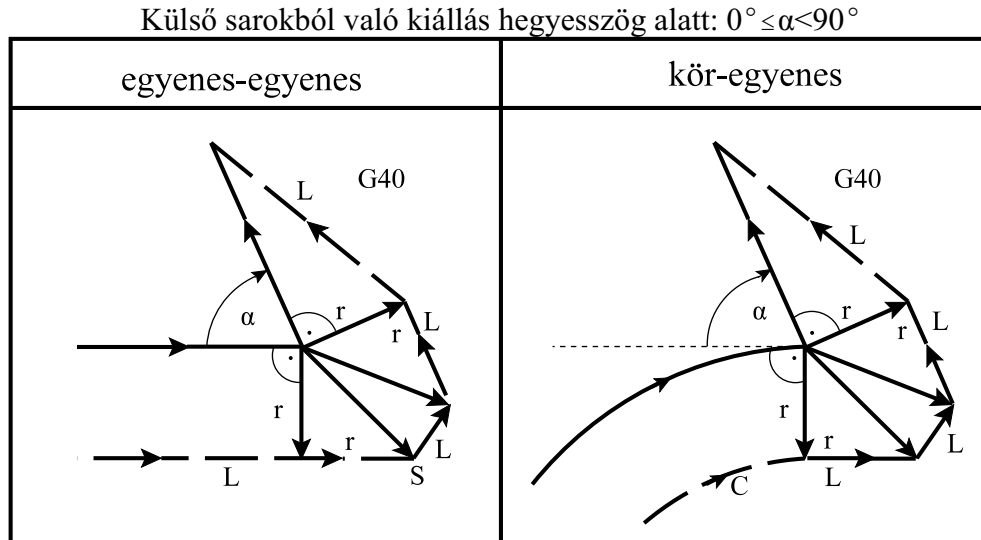


14.4.3-1 ábra

Külső sarokból való kiállás tompaszög alatt: $90^\circ \leq \alpha \leq 180^\circ$



14.4.3-2 ábra



14.4.3-3 ábra

A sugárkorrekció kikapcsolásának speciális esetei:

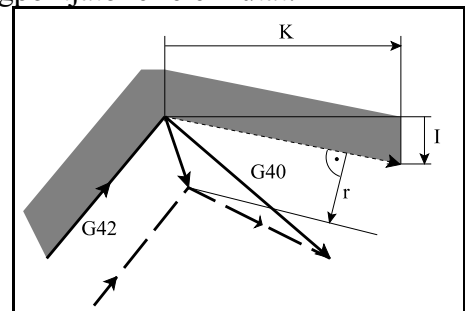
Ha a korrekció kikapcsolását végző mondatban (G40) I, J, K-nak értéket adunk, de csak a kiválasztott síkban lévőknek (például: G17 esetén I, J-nek), akkor a megelőző mondat és az I, J, K által meghatározott egyenes közti metszéspontra áll a vezérlő. I, J, K értéke mindig inkrementális, és az általuk megadott vektor a megelőző mondat végpontjától elfele mutat.

Ez a lehetőség például belső sarokból való kiállás esetén hasznos.

```

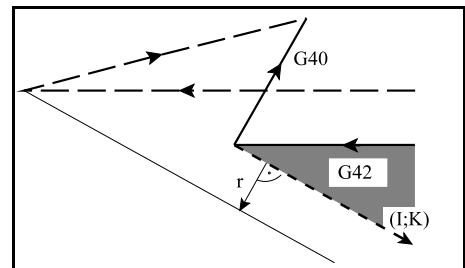
...
...G91 G18 G42...
N100 G1 X120 Z50
N110 G40 X-120 Z70 I-20 K100
...

```



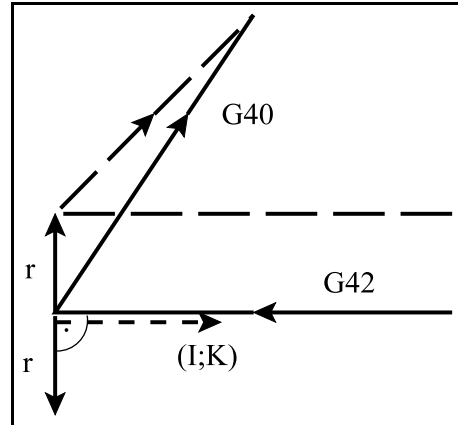
14.4.3-4 ábra

Ebben az esetben a vezérlés mindig metszéspontot számol, függetlenül attól, hogy belső, vagy külső sarokot munkálunk meg.



14.4.3-5 ábra

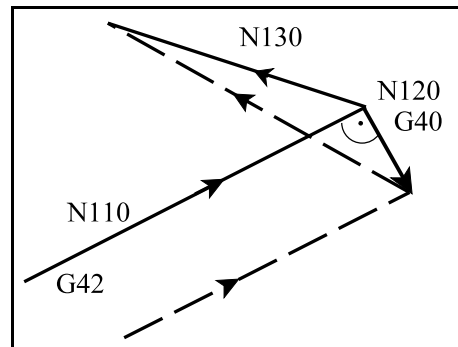
Ha nem talál metszéspontot, az előző mondat végpontjára merőlegesen áll rá.



14.4.3-6 ábra

Ha a korrekció kikapcsolását olyan mondatban végezzük, ahol a kiválasztott síkban mozgást nem programozunk, akkor az előző mondat végpontjára merőlegest állít, és a korrekciós vektor törlése a következő mozgásmondat végére történik meg.

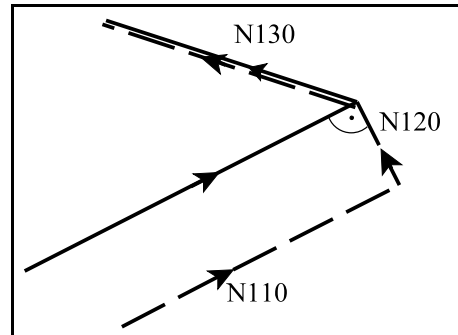
```
...G42 G18 G91...
N110 G1 X80 Z80
N120 G40
N130 X40 Z-70
...
```



14.4.3-7 ábra

Ha a korrekció kikapcsolását (G40) tartalmazó mondatban nulla elmozdulást programoztunk, vagy nulla elmozdulás adódik ki, az előző mondat végpontjára sugárkorrekciójnyi hosszúságú merőlegest állít, majd a G40-es mondatban lemozogja azt. Például:

```
...G42 G18 G91...
N110 G1 X80 Z80
N120 G40 X0
N130 X40 Z-70
...
```



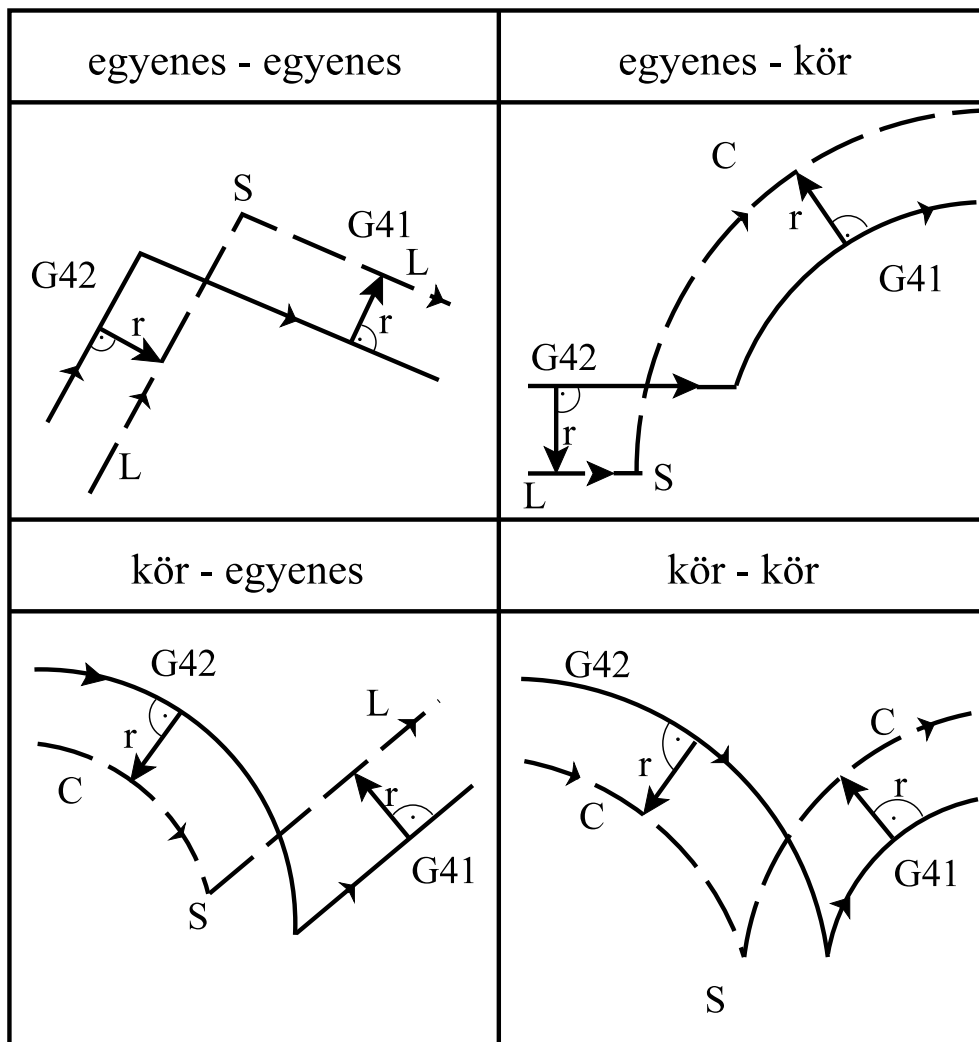
14.4.3-8 ábra

14.4.4 Irányváltás a sugárkorrekció számításban.

A szerszámsugár korrekció számításának, vagyis a kontúr követésének irányát az alábbi táblázat tartalmazza:

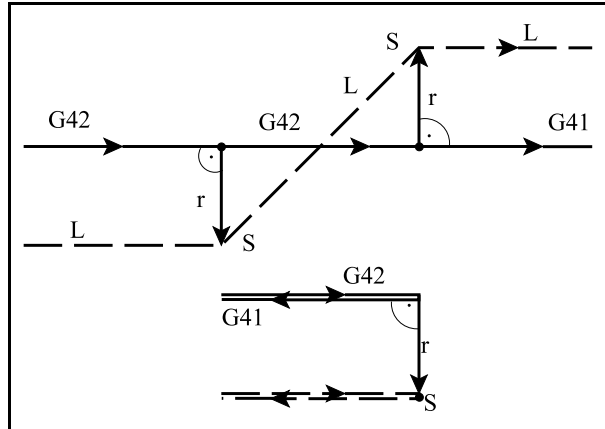
	sugárkorrekció: pozitív	sugárkorrekció: negatív
G41	balról	jobbról
G42	jobbról	balról

A kontúrkövetés iránya megfordítható a szerszámsugár korrekció számítás bekapcsolt állapotában is. Ez történhet G41, vagy G42 programozásával, vagy az eddigiekkel ellentétes előjelű szerszámsugár korrekció lehívásával T címen. Amikor a kontúrkövetés iránya megfordul a vezérlés nem vizsgálja, hogy "kívül", vagy "belül" van, hanem első lépésben mindig metszéspontot számít. Az alábbi ábrákon pozitív szerszámsugarat tételeztünk fel és G42-ből G41-be történő kapcsolást:



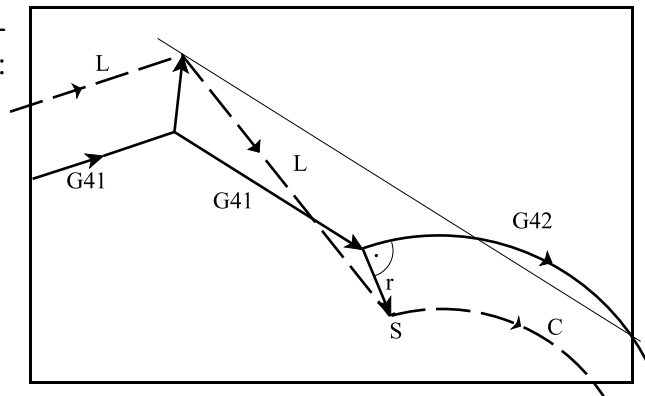
14.4.4-1 ábra

Ha egyenes–egyenes átmenetnél nem adódik metszéspont, a szerszám pályája a következő lesz:



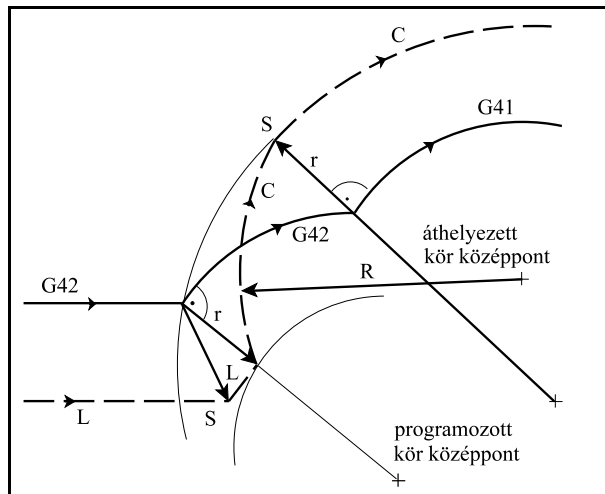
14.4.4-2 ábra

Ha egyenes–kör átmenetnél nem adódik metszéspont, a szerszám pályája a következő lesz:



14.4.4-3 ábra

Ha kör–egyenes, vagy kör–kör átmenetnél nem adódik metszéspont, akkor az első körmondat kezdőpontjában kiadódó korrekciós vektor végpontját és a második mondat kezdőpontjára merőleges korrekciós vektor végpontját összeköti egy korrigálatlan, programozott R sugarú körívvel. Ekkor az összekötő körív középpontja nem fog egybeesni a programozott körív középpontjával. Ha az irányváltás még a fent vázolt kör-középpont áthelyezéssel sem végezhető el **3047 NEM KAPCSOLHATÓ ÁT** hibajelzést ad a vezérlő.



14.4.4-4 ábra

14.4.5 A vektor megtartás programozása (G38)

A

G38 v

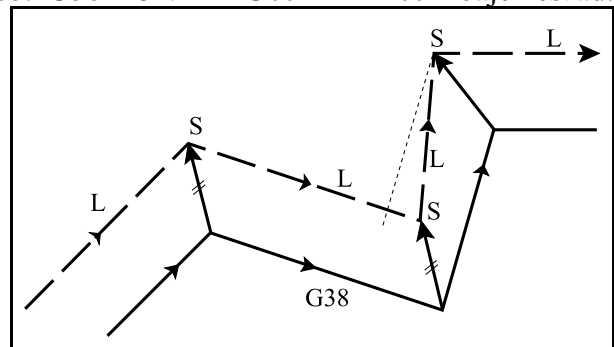
parancs hatására a vezérlés a síkbeli szerszámsugár korrekció számítás bekapcsolt állapotában az előző mondat és a G38 mondat közötti utolsó korrekciós vektort megtartja és azt érvényesíti a G38 mondat végén, függetlenül a G38 mondat, és a következő mondat közti átmenettől.

A G38 kód egylovétű, azaz nem öröklődik. Ha több egymást követő mondatban szükséges a vektor megtartása a G38-at újra programozni kell.

G38 programozása csak G00, vagy G01 állapotban lehetséges, azaz a vektor megtartó mondatnak mindig egyenes interpolációnak kell lenni. Ellenkező esetben *3040 G38 NEM G0, G1 ALATT* hibajelzést ad a vezérlő. Ha a G38 kódot nem a síkbeli szerszámsugár korrekció bekapcsolt állapotában (G41, G42) használjuk a vezérlés *3039 G38 MONDAT G40 ALATT40* hibajelzést ad.

Példa a G38 működésére:

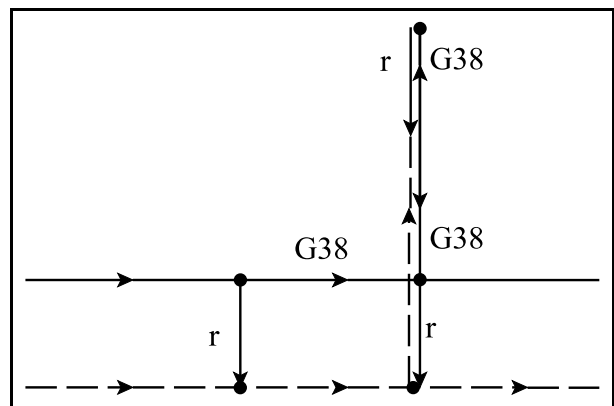
```
...G18 G41 G91...
N110 G1 X120 Z60
N120 G38 X-80 Z90
N130 X140 Z20
N140 Z120
...
```



14.4.5-1 ábra

Ha beszúrást akarunk programozni a kontúrkövetés kikapcsolása nélkül:

```
...G18 G42 G91...
N110 G1 Z40
N120 G38 Z50
N130 G38 X140
N140 G38 X-140
N150 Z60
...
```



14.4.5-2 ábra

14.4.6 Sarokív programozása (G39)

A

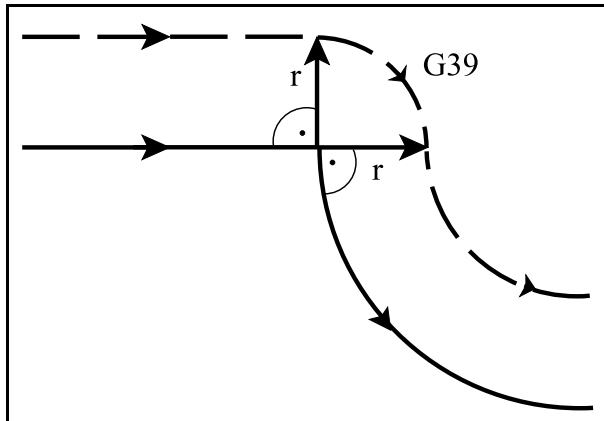
G39 (I J K)

mondat programozásával a síkbeli szerszámsugár korrekciószámítás bekapcsolt állapotában elérhető, hogy külső sarkok kerülése esetén a vezérlő ne számoljon automatikusan metszéspontot, vagy ne iktasson be egyenes szakaszokat a kerüléshez, hanem egy szerszámsugárnyi köríven mozogjon a szerszám középpontja.

G41 állapotban G02, G42 állapotban G03 irányú szerszámradiusnyi sugarú kört iktat be.

A kör kezdőpontját az előző mondat pályájának végpontjára merőleges szerszámradiusnyi hosszúságú vektor, végpontját pedig a következő mondat kezdőpontjára merőleges szerszámradiusnyi hosszúságú vektor adja. G39-et külön mondatban kell programozni:

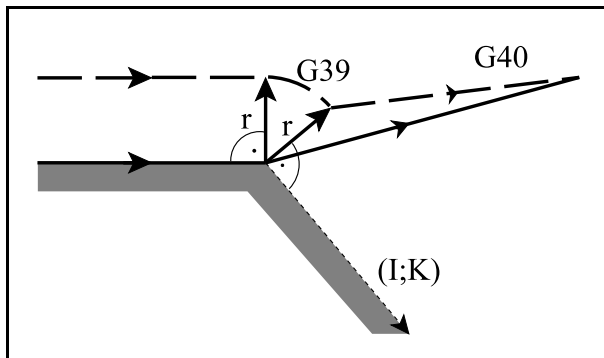
```
...G18 G91 G41...
N110 G1 Z100
N120 G39
N130 G3 X-160 Z80 K80
...
```



14.4.6-1 ábra

Ha a G39-es mondatban I-t, J-t vagy K-t programozunk, a kiválasztott síknak megfelelően, a kör végpontját az előző mondat végpontjától az I, J vagy K által meghatározott vektorra merőleges irányú, szerszámradiusnyi hosszúságú vektor adja:

```
...G18 G91 G41...
N110 G1 Z100
N120 G39 I-60 K50
N130 G40 X60 Z110
...
```



14.4.6-2 ábra

Az I, J vagy K által meghatározott vektorra érvényesek az előzőleg beállított tükrözési parancsok. A léptékezési parancs értelemszerűen az irányt nem befolyásolja.

A G39-es típusú mondatban semmilyen mozgásparancsot nem lehet programozni. Ha G39 parancsot adunk ki G40 állapotban, vagy térbeli korrekciószámítási üzemmódban a vezérlés 3036 G39 MONDAT G40 ALATT hibajelzést ad.

14.4.7 Általános tudnivalók a síkbeli sugárkorrekció alkalmazása esetére

A kontúrkövetés bekapcsolt állapotában (G41, G42) a vezérlés mindig két, a kiválasztott síkba eső mozgásmondattal közötti korrekciósvektorokat kell, hogy számolja. A gyakorlatban szükség lehet arra, hogy két síkbeli mozgásmondattal közbe mozgást nem tartalmazó mondatot, vagy nem a kiválasztott síkba eső mozgást tartalmazó mondatot programozzunk. Ezek a következők lehetnek:

funkciók: M, S, T

várakozás: G4 P

a kiválasztott síkon kívüli mozgás: (G18) G1 Y

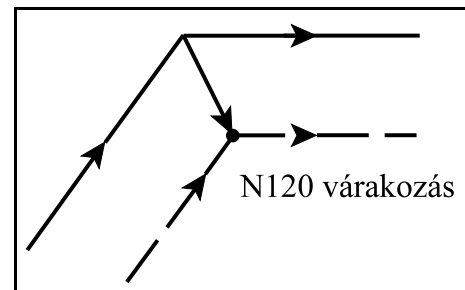
alprogram hívás: M98 P

különleges transzformációk be-, kikapcsolása: G50, G51, G50.1, G51.1, G68, G69.

☞ *Megjegyzés:* ha alprogramot hívunk, akkor ügyeljünk arra, hogy az alprogram a kiválasztott síkba eső mozgásmondattal kezdődjen, mivel ellenkező esetben a pálya torzulni fog.

A vezérlés a programban két síkbeli mozgásmondattal között **egy** fentebb felsorolt mondat programozását elfogadja, az a szerszám pályáját nem befolyásolja:

```
...G18 G42 G91...
N110 G1 X140 Z50
N120 G4 P2
N130 Z60
...
```

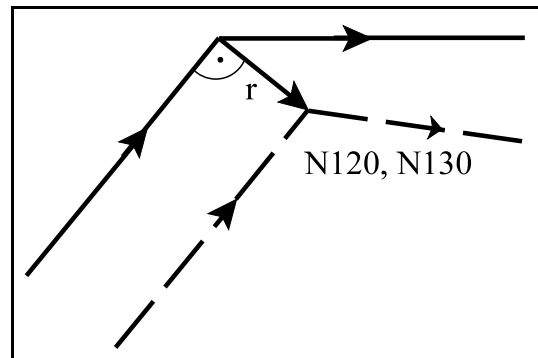


14.4.7-1 ábra

Ha két kontúrmondat között egy, vagy több kerülőmondatot iktat a vezérlő a kontúrmondatok közé programozott egyéb mondat végrehajtási helye a mondatonkénti megállás helye, amit az ábrákban "S"-sel jelöltünk.

Ha két, nem a kiválasztott síkba eső mozgást tartalmazó, vagy mozgást nem tartalmazó mondatot írunk a programba, a vezérlés az utolsó kontúrmondat végpontjára merőlegest állít, és ez lesz a korrekciós vektor, a pálya torzul:

```
...G18 G42 G91...
N110 G1 X140 Z50
N120 G4 P2
N130 S400
N140 Z60
...
```

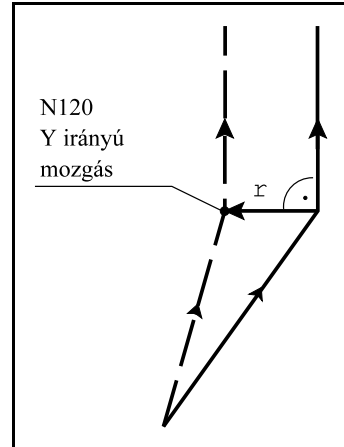


14.4.7-2 ábra

Ha a sugárkorrekció bekapcsolása után lehet csak fogást venni **Y** irányban a következőképp járhatunk el:

```
...G18 G91...
N110 G41 G0 X140 Z50
N120 G1 Y-40
N130 X80
...
```

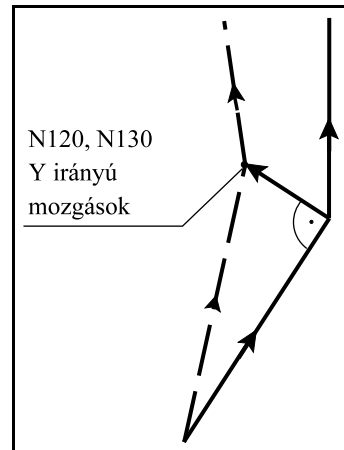
Ebben az esetben a szerszám pályája korrekt lesz, amint az ábrán látszik.



14.4.7-3 ábra

Ha azonban az **Y** irányú mozgást felbontjuk egy gyorsmeneti és egy előtolásos szakaszra, a pálya torzulni fog, mert két, nem a kontúrkövetés síkjában levő mondat jön egymás után:

```
...G18 G91...
N110 G41 G0 X140 Z50
N120 Y-35
N130 G1 Y-5
N140 X80
...
```



14.4.7-4 ábra

Áthidaló megoldásként iktassunk a két **Y** irányú mozgás közé egy kis **X** irányú mozgást:

```
...G18 G91...
N110 G41 G0 X138 Z50
N120 Y-35
N130 X2
N140 G1 Y-5
N150 X80
...
```

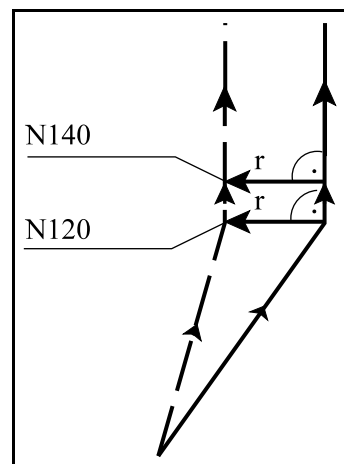
A fenti fogással élve már helyesen épül föl a korrekciós vektor.

Ha két kontúrmondat közé:

```
G22, G23, G52, G54-G59, G92
G53
G28, G29, G30
```

utasításokat iktatunk a szerszám pályája az alábbiak szerint alakul.

Abban az esetben, ha G41, vagy G42 bekapcsolt állapotában két mozgásmondat között G22, G23, G52, G54-G59, vagy G92 parancsot programozunk, az ezt megelőző mondat végpontján törli a korrekciós vektort, végrehajtja a parancsot, majd a következő mozgásmondat végpontjában visszaállítja azt. Abban az esetben, ha ezt a parancsot megelőző, vagy követő mozgásmondat

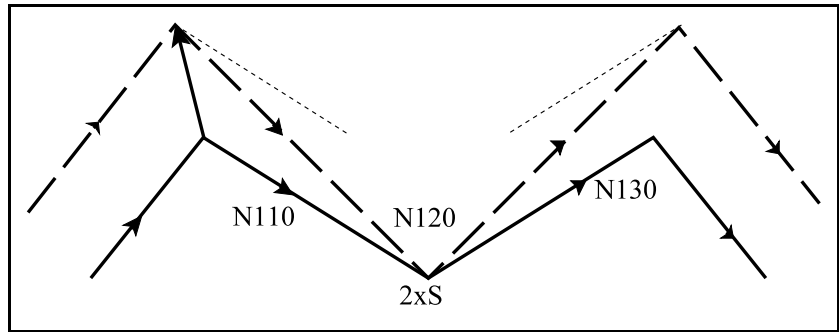


14.4.7-5 ábra

körinterpoláció, 3041 G2, G3 UTÁN ILLEG. MONDAT hibajelzést ad a vezérlő.

Például:

```
...G91 G18 G41...
N110 G1 X-100 Z80
N120 G92 X0 Z0
N130 X100 Z80
...
```

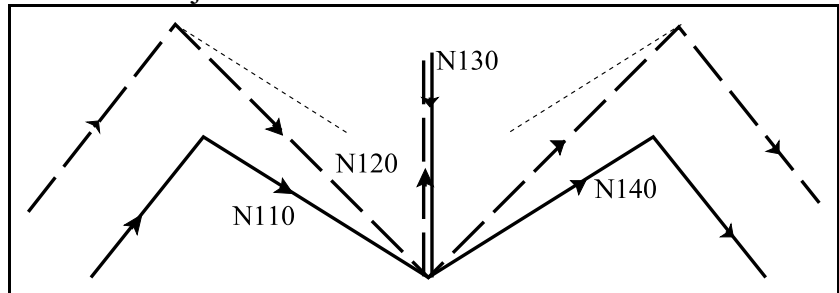


14.4.7-6 ábra

Abban az esetben, ha G41, vagy G42 bekapcsolt állapotában két mozgásmondat között G53 parancsot programozunk, az ezt megelőző mondat végpontján törli a korrekciós vektort, végrehajtja G53-ban a pozicionálást, majd a következő, nem G53 mozgásmondat végpontjában visszaállítja azt. Abban az esetben, ha ezt a parancsot megelőző, vagy követő mozgásmondat körinterpoláció, 3041 G2, G3 UTÁN ILLEG. MONDAT hibajelzést ad a vezérlő.

Például:

```
...G91 G18 G41...
N110 G1 X-100 Z80
N120 G53 X160
N130 G53 X0
N140 X100 Z80
...
```

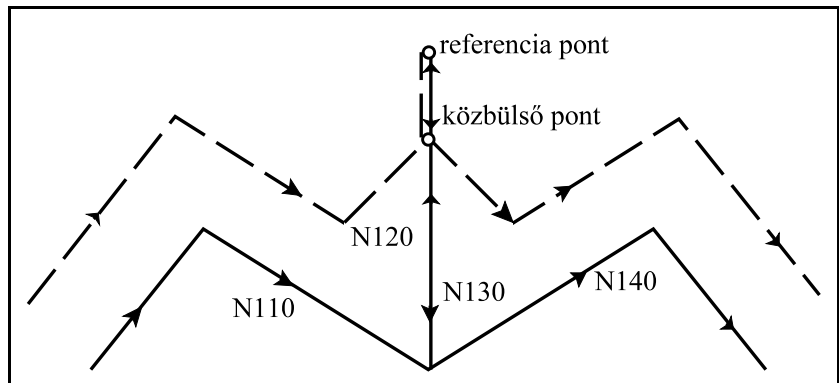


14.4.7-7 ábra

Abban az esetben, ha G41, vagy G42 bekapcsolt állapotában két kontúrmondat között G28-at, vagy G30-at, utána G29-et programozunk, a közbülső pontra pozicionáló mondat végpontján törli a korrekciós vektort, végrehajtja a refpontra állást, majd a visszatérő G29 mondat végpontjában visszaállítja azt.

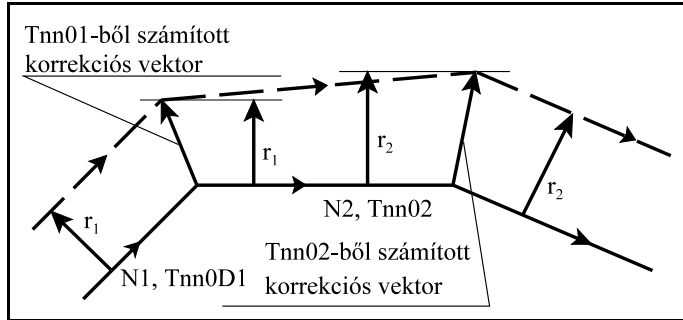
Például:

```
...G91 G18 G41...
N110 G1 X-100 Z80
N120 G28 X160
N130 G29 X0
N140 X100 Z80
...
```



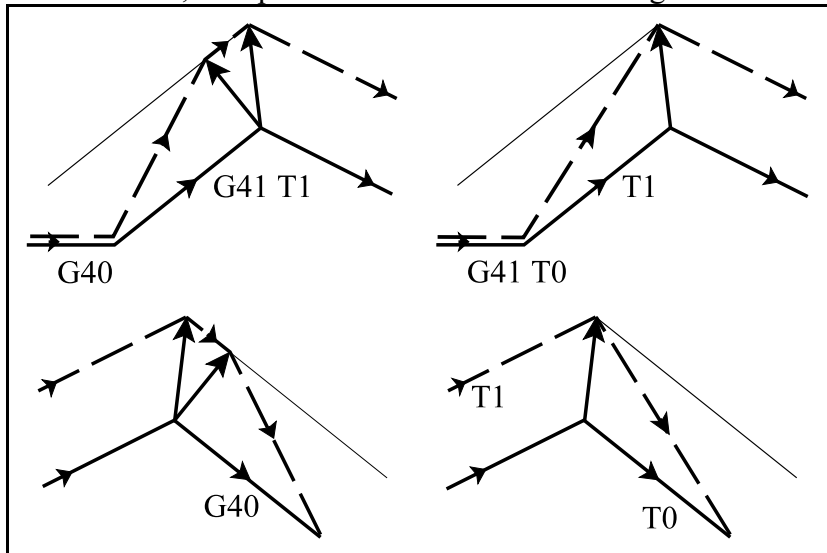
14.4.7-8 ábra

A sugárkorrekció számítás bekapcsolt állapotában (G41, G42) T címen új korrekciós sugárérték is lehívható. Ha a sugár értéke előjelet vált a kontúron való irányváltás folyamata zajlik le, amit már tárgyaltunk. Ha a sugárérték nem vált előjelet az eljárás a következő lesz. Az új sugárértékkal számolt korrekciós vektor annak a mondatnak a végpontjában kerül kiszámításra, amelyikben az új T címet programoztuk. Mivel ennek a mondatnak a kezdőpontjában az előző sugárértékkel lett a korrekciós vektor kiszámítva a szerszámközpont pályája nem lesz párhuzamos a programozott pályával. Körmondatban is lehívható új sugárkorrekciós érték T címen, ám ekkor a szerszámközpont egy változó sugarú köríven fog mozogni. Speciális esete a fent elmondottaknak, ha a sugárkorrekciószámítás bekapcsolt állapotában T00-lal kikapcsoljuk a korrekciót, Tnn-nel pedig bekapcsoljuk azt. Az alábbi példa alapján figyeljük meg a szerszám-pálya különbségét, ha a korrekciót G41, vagy G42-vel kapcsoljuk be, és G40-nel ki, illetve, ha a korrekció ki-, bekapcsolását T cím lehívásával végezzük:



14.4.7-9 ábra

14.4.7-9 ábra

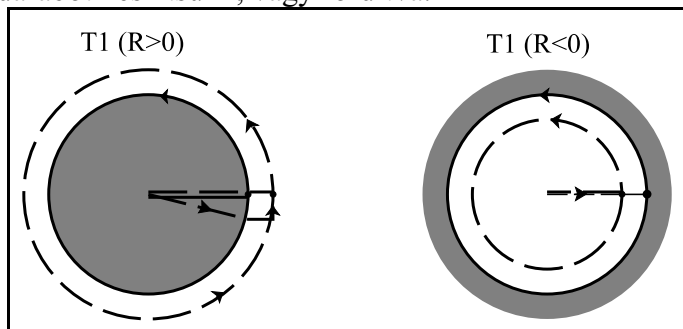


14.4.7-10 ábra

Egy adott programrészletet, vagy alprogramot használhatunk arra is, hogy pozitív sugárkorrekcióval apa, negatív sugárkorrekcióval anya darabot készítsünk, vagy fordítva. Tekintsük az alábbi kis programrészletet:

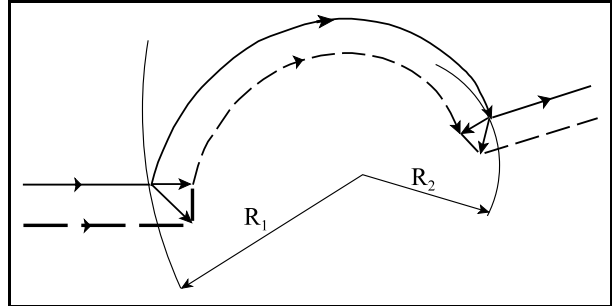
```

...
N020 G42 G1 X160 T1
N030 G1 Z-5
N040 G3 I-80
N050 G1 Z2
N060 G40 G0 X0
...
    
```



14.4.7-11 ábra

Ha változó sugarú körre alkalmazzuk a sugárkorrekciót a kör kezdőpontjában a korrekciós vektor(oka)t olyan képzeletbeli körhöz számítja ki a vezérlő, amelynek sugara megegyezik a programozott kör kezdőponti sugarával, középpontja pedig egybeesik a programozott kör kezdőponttal. A kör végpontjában a korrekciós vektor(oka)t olyan képzeletbeli körhöz számítja ki, amelynek sugara megegyezik a programozott kör végponti sugarával, középpontja pedig egybeesik a programozott kör középponttal.



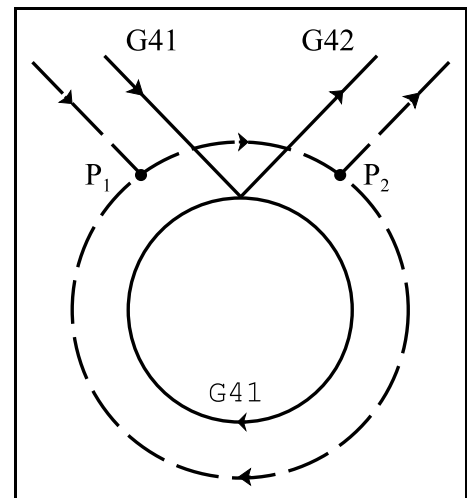
14.4.7-12 ábra

Teljes kört programozva kontúrkövetés közben számos olyan eset adódhat, amikor a szerszám pályája többet tesz meg egy teljes kör fordulatnál.

A kontúron való irányváltás programozásakor adódhat például ilyen helyzet:

```
...G17 G42 G91...
N110 G1 X60 Y-40
N120 G41 G2 J-40
N130 G42 G1 X60 Y40
...
```

A szerszámközepont tesz egy teljes körívet a P_1 ponttól P_1 pontig, majd egy ívet P_1 ponttól P_2 pontig.

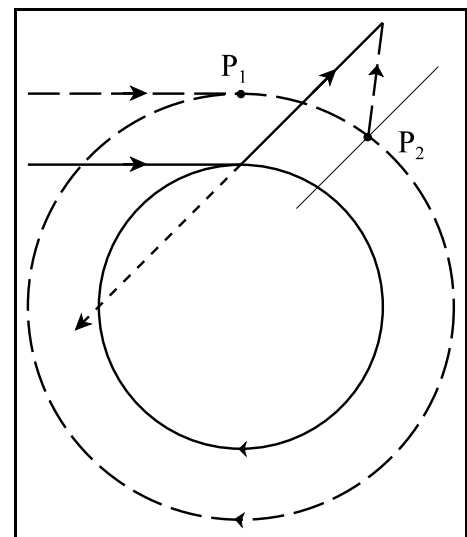


14.4.7-13 ábra

Ha a kontúrkövetést I, J, K programozásával kapcsoljuk ki, az alábbi példa hasonló helyzetet mutat:

```
...G18 G90 G41...
N090 G1 Z60
N100 G2 I-60
N110 G40 G1 X360 Z120 I-60 K-60
...
```

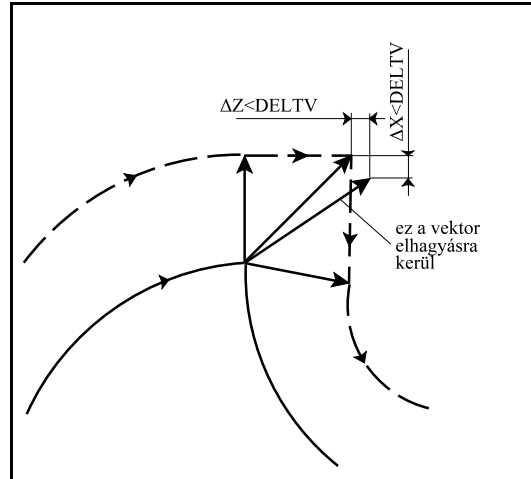
A szerszámközepont tesz egy teljes körívet a P_1 ponttól P_1 pontig, majd egy ívet P_1 ponttól P_2 pontig.



14.4.7-14 ábra

Éles sarkok kerülése esetén két vagy több korrekciós vektor is képződhet. Ha ezek végpontjai közel esnek egymáshoz, alig adódik mozgás a két pont között. Abban az esetben, ha a két vektor közötti távolság mindkét tengelyen kisebb, mint a paramétermezőben beállított *DELTV* paraméter értéke az ábrán látható vektor elhagyásra kerül, és a szerszám pályája az ábra szerint módosul.

☞ *Megjegyzés:* A *DELTV* paraméter indokolatlanul nagy értéke mellett külső hegyesszögű sarkok kerülésekor előfordulhat, hogy a sarkot megsérti a szerszám!

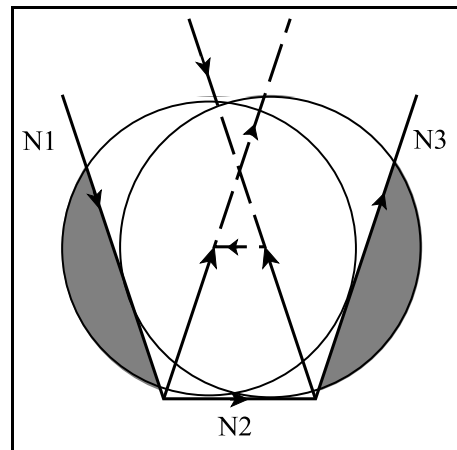


14.4.7-15 ábra

14.4.8 A kontúrkövetés zavarproblémái. Interferenciavizsgálat.

A kontúrkövetés végrehajtása során számos esetben előfordulhat, hogy a szerszám pályája ellentétes lesz a programozott pályával. Ebben az esetben a programozói szándékkal ellentétesen a szerszám belevághat a munkadarabba. Ezt a jelenséget nevezzük a kontúrkövetés zavarának, vagy interferenciának.

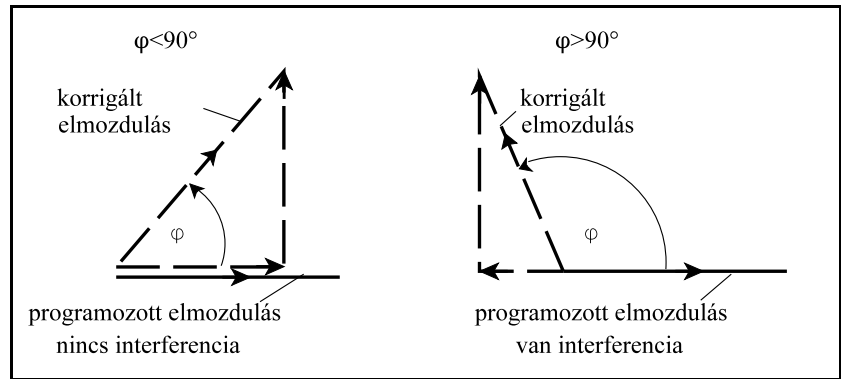
Az ábrán látható esetben a metszéspontok kiszámítása után az N2 mondat végrehajtása során a programozottal ellentétes szerszám-pálya adódik. A bevonalkázott terület jelzi, hogy a szerszám belevág a munkadarabba.



14.4.8-1 ábra

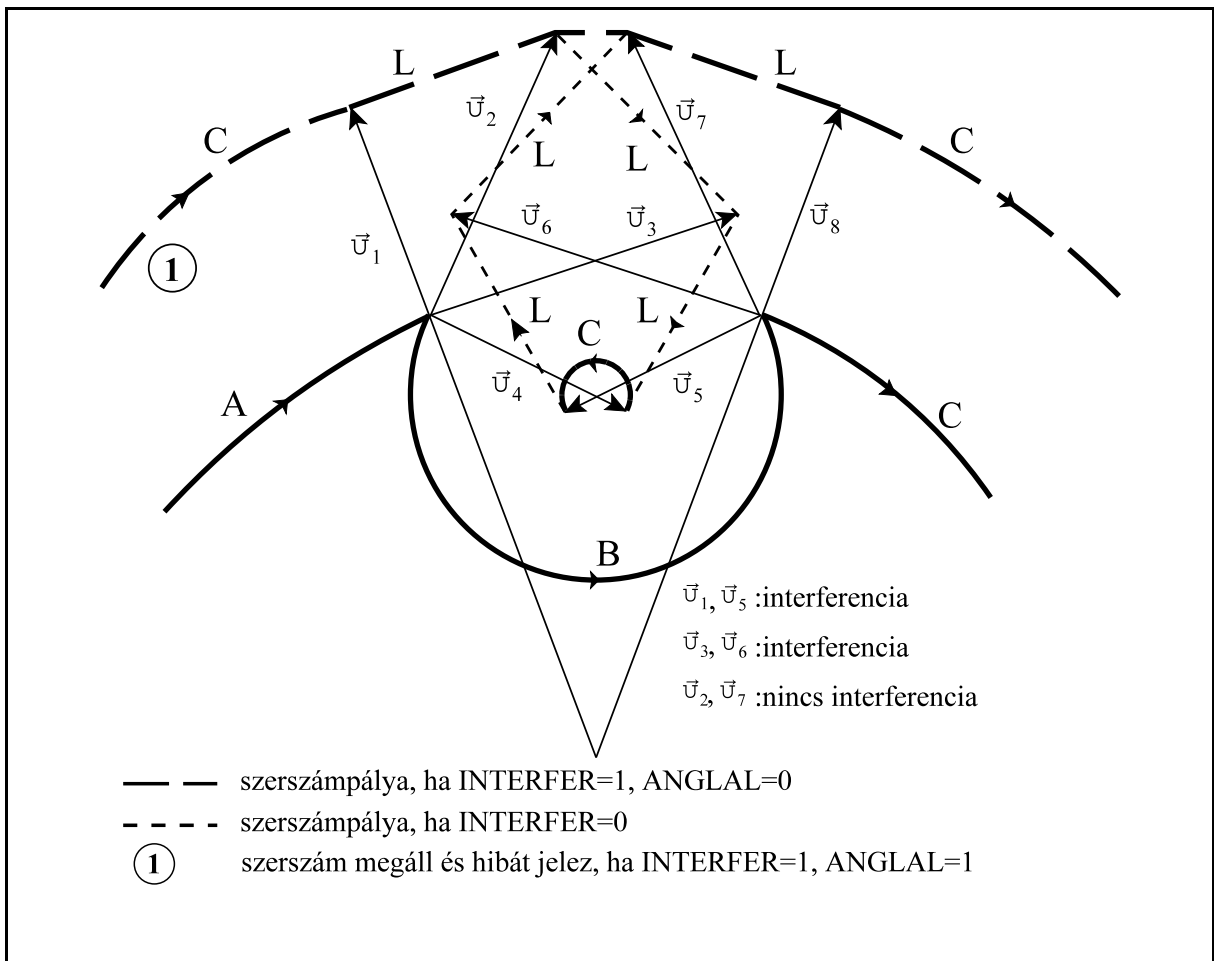
Az ilyen esetek elkerülése érdekében a vezérlés interferenciavizsgálatot végez, ha a paramétermező *INTERFER* paraméterét 1-be írjuk. Ebben az esetben a vezérlés azt vizsgálja, hogy a programozott elmozdulás és a sugárkorrekcióval korrigált elmozdulás közötti φ szögre teljesül-e a következő feltétel: $-90^\circ \leq \varphi \leq +90^\circ$.

Más szavakkal, a vezérlés azt vizsgálja, hogy a korrigált elmozdulásvektornak van-e a programozott elmozdulásvektorral ellentétes komponense.



14.4.8-2 ábra

Ha a paramétermező *ANGLAL* paramétere 1-be van írva a szögvizsgálat után a vezérlő 3048 *INTERFERENCIA HIBA* kóddal interferenciahibát jelez egy mondattal korábban, mint ahol a hiba fellép.



14.4.8-3 ábra

Abban az esetben, ha a paramétermező *ANGLAL* paramétere 0-ba van állítva, a vezérlő nem jelez hibát, hanem automatikusan korrigálni próbálja a kontúrt azzal a céllal, hogy a bevágásokat elkerülje. A korrekció menete a következő:

A kontúrkövetés be van kapcsolva az A, B, és C mondaton. A és B mondat között a kiszámított

korrekciós vektorok: $\vec{v}_1, \vec{v}_2, \vec{v}_3, \vec{v}_4$. B és C mondat között a korrekciós vektorok pedig: $\vec{v}_5, \vec{v}_6, \vec{v}_7, \vec{v}_8$.

Ha \vec{v}_4 és \vec{v}_5 között interferencia van \vec{v}_4 -et és \vec{v}_5 -öt elhagyja,

Ha \vec{v}_3 és \vec{v}_6 között interferencia van \vec{v}_3 -at és \vec{v}_6 -ot elhagyja,

Ha \vec{v}_2 és \vec{v}_7 között interferencia van \vec{v}_2 -t és \vec{v}_7 -et elhagyja,

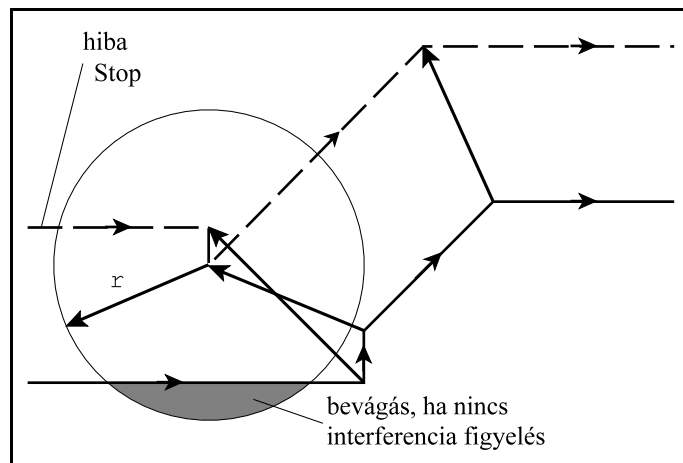
Ha \vec{v}_1 és \vec{v}_8 között interferencia van nem hagyható el, hibát jelez.

A fentiekből látható, hogy a korrekciós vektorokat a B mondat kezdőpontján és végpontján párba veszi, és párosával hagyja el azokat. Ha az egyik oldalon a korrekciós vektorok száma egy, vagy egyre csökken, akkor csak a másik oldaliakat hagyja el. Az elhagyás addig folytatódik, amíg az interferencia fennál. A B mondat kezdőpontján az első, és végpontján az utolsó korrekciós vektort nem hagyhatja el. Ha az elhagyások eredményeképpen megszűnik az interferencia nem jelez hibát, ha nem szűnik meg *3048 INTERFERENCIA HIBA* hibát jelez. Az elhagyások után a maradék korrekciós vektorokat mindig egyenessel köti össze, még abban az esetben is, ha a B mondat kör volt.

A fenti példából látható, hogy az A mondat végrehajtás csak akkor kezdődik el, ha a B mondatra elvégezte a vezérlő az interferenciavizsgálatot. Ehhez azonban a C mondatot is be kellett olvasni a pufferbe, és a B-C átmenetnél a korrekciós vektorokat kiszámítani.

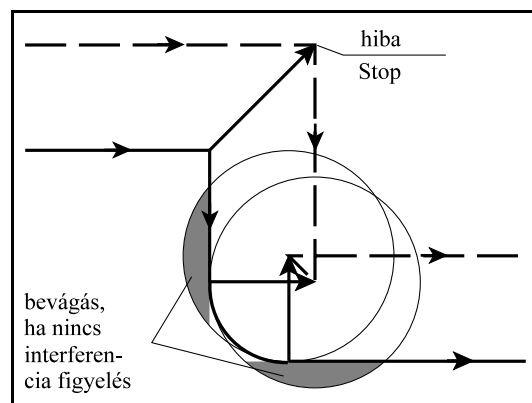
Az alábbiakban néhány tipikus esetet adunk interferenciára.

A szerszámsugárnál kisebb lépcső forgácsolása. A vezérlés *3048 INTERFERENCIA HIBA* hibát jelez mert belevágna az anyagba.



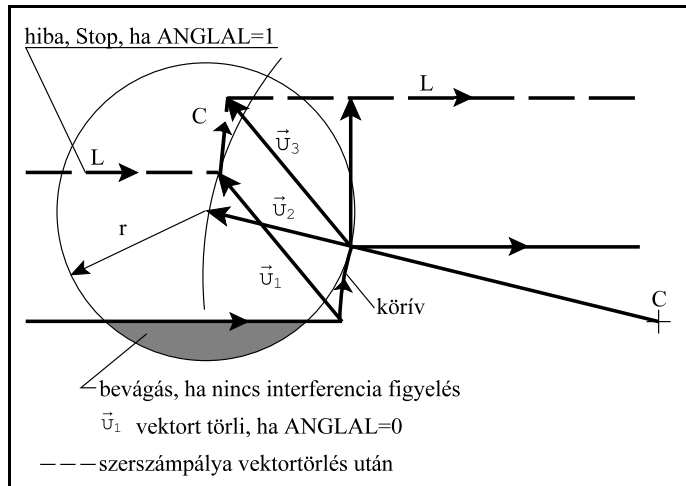
14.4.8-4 ábra

A szerszámsugárnál kisebb rádiuszú belső sarok megmunkálása. A vezérlés *3048 INTERFERENCIA HIBA* hibát jelez mert belevágna a munkadarabba.



14.4.8-5 ábra

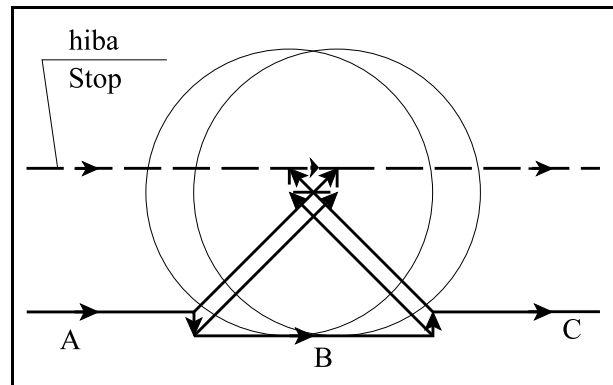
A szerszámsugárnál kisebb lépcső forgácsolása körív mentén. Ha az *ANGLAL* paraméter 0 a vezérlés a \vec{v}_2 vektort törli, és a \vec{v}_1 \vec{v}_3 vektort egy egyenessel köti össze, hogy elkerülje a bevágást. Ha az *ANGLAL* paraméter értéke 1, akkor 3048 *INTERFERENCIA HIBA* hibát jelez, és megáll az előző mondatban.



14.4.8-6 ábra

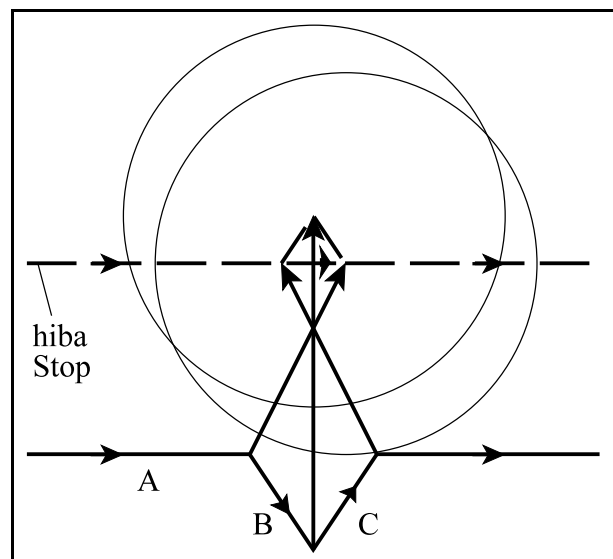
Vannak esetek, amikor valójában nem vágna bele a szerszám az anyagba, de az interferenciavizsgálat hibát jelez.

Ha a sugárkorrekciónál kisebb süllyesztést munkálunk meg, a valóságban esetleg belevágás nem történik, mint az ábrán látható, de a vezérlés 3048 *INTERFERENCIA HIBA* hibát jelez, mert a B mondatban a korrigált pályán az elmozdulás iránya ellentétes a programozottal.



14.4.8-7 ábra

Az alábbi példában szintén interferencia hibát jelez, mivel a B mondatban a korrigált pálya elmozdulása ellentétes a programozottal.



14.4.8-8 ábra

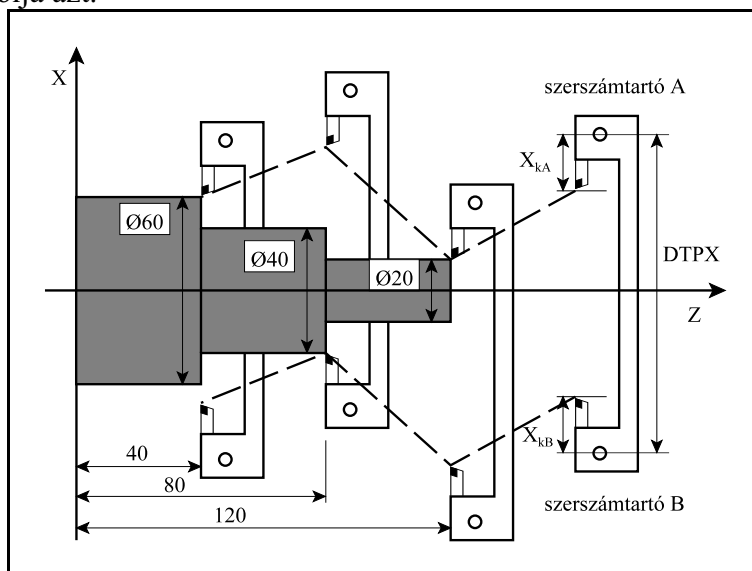
15 Különleges transzformációk

15.1 Tükrözés kettős szerszámtartó esetén (G68)

A

G68 utasítás bekapcsolja a kettős szerszámtartóra való tükrözést, a **G69** utasítás pedig kikapcsolja azt.

Ez a funkció két egymással szemben álló revolverfej programozására használható. Az első "A" szerszámtartó a pozitív a második "B" szerszámtartó pedig a negatív térnegyedben munkál meg. A tükrözés mindig az X tengelyre történik a G68 utasítás hatására. Az X tengelyre programozott koordináták ekkor előjelet váltanak, egyúttal egy koordináta transzformáció is történik az X tengely mentén. A koordinátaeltolás mértékét a két revolverfej közti távolság határozza meg, amit az 1001 DTPX paraméteren lehet megadni. Ezután a programot úgy kell megírni, mintha az eredeti "A"



15.1-1 ábra

revolverfejjel dolgoznánk, vagyis a programot a pozitív térnegyedre írjuk le. Például:

```
T101      (beváltjuk az "A" szerszámtartóban lévő 1-es szerszámot)
G0 X20 Z120 (az 1-es szerszámmal pozícionálunk)
G68      (bekapcsoljuk a tükrözést)
T202      (beváltjuk a "B" szerszámtartóban lévő 2-es szerszámot)
G0 X40 Z80 (2-es szerszámmal pozícionálunk)
G69      (kikapcsoljuk a tükrözést)
T101      (beváltjuk az "A" szerszámtartóban lévő 1-es szerszámot)
G0 X60 Z40 (az 1-es szerszámmal pozícionálunk)
```

Megjegyzés:

- A G68 és a G69 parancsoknak mindig önálló mondatban kell állnia, más parancsot abban a mondatban nem programozhatunk.
- A funkció használatához meg kell mérni a két szerszámtartó távolságát és az 1001 DTPX paraméterre be kell azt írni.
- G68 bekapcsolt állapotában a programozotthoz képest megfordul:
 az X koordináta előjele,
 G2 körirány G3-ra vált és fordítva,
 G41 irány G42-re vált és fordítva

15.2 Léptékezés (G50, G51)

A

G51 v P

paranccsal egy leprogramozott alakzat kicsinyíthető vagy nagyítható.

P1...P4: az alkatrészprogramban megadott pontok,

P1'...P4': léptékezés utáni pontok,

P0: léptékezés középpontja.

v koordinátákon adhatjuk be a léptékezés középpontjának koordinátáit. A felhasználható címek: X, Y, Z, U, V, W.

A léptékezés középpontjának v koordinátáit megadhatjuk abszolút és növekményes adatként is G90, G91, vagy I operátor használatával.

Ha valamelyik tengelycímnek, vagy egyiknek sem adunk értéket a léptékezés középpontjának a pillanatnyi tengelypozíciót veszi.

P címen a léptékezés arányszámát állíthatjuk be. Értékét 8 decimális számjegyen ábrázolhatjuk, a tizedespont helye tetszőleges.

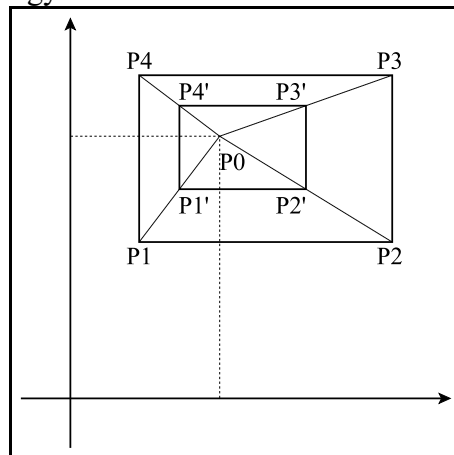
A

G50

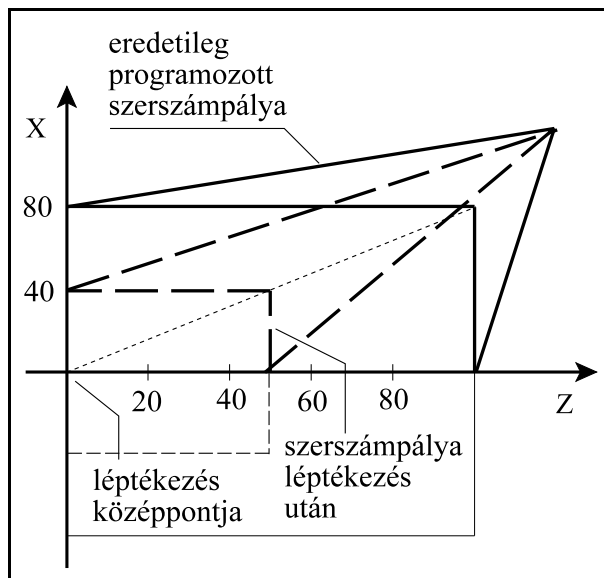
paranccsal a léptékezésszámítás kikapcsolható.

Mintapélda:

```
N1 G90 G0 X100 Z120
N2 G51 X0 Z0 P0.5
N3 G1 X0 Z100 F150
N4 X80
N5 Z0
N6 G50
N7 G0 X100 Z120
```



15.2-1 ábra



15.2-2 ábra

15.3 Tükrözés (G50.1, G51.1)

A

G51.1 v

paranccsal a v-ben kiválasztott koordináták mentén tükrözi a leprogramozott alakzatot úgy, hogy a tükrözés tengelyének, vagy tengelyeinek koordinátáit v-n adhatjuk meg. v lehet: X, Y, Z, U, V, W, A, B, C.

A tükrözés tengelyeinek v koordinátáit megadhatjuk abszolút és növekményes adatként is G90, G91, vagy I operátor használatával.

Ha valamelyik tengelycímnek, nem adunk értéket arra nem végez tükrözést.

A

G50.1 v

paranccsal a v-n megadott koordinátatengely(ek)en kikapcsolódik a tükrözés. v koordinátákra tetszőleges adat írható, hatása csak a kikapcsolás tényét rögzíti.

A parancs kiadásakor nem állhat fenn sem G68, sem léptékezési (G51) parancsállapot. Ellenkező esetben 3000 TÜKRÖZÉS G51, G68 ALATT hibajelzést ad.

Ha a kiválasztott sík egyik tengelyére tükrözünk:

- a körirány automatikusan megfordul (G02 G03 csere),
- a szerszámsugár korrekció (G41, G42) ellentétesen értelmeződik.

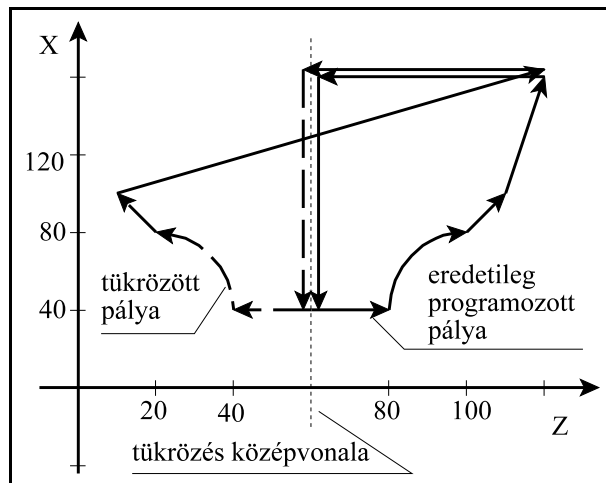
Mintapélda:

alprogram:

```
O0101
N1 G0 X40 F120
N2 G1 Z80
N3 G3 X80 Z100 R20
N4 G1 X100 Z110
N5 M99
```

főprogram:

```
O0100
N10 T101
N20 G0 X160 Z60 M3 S1000
N30 M98 P101 (alprogram hívás)
N40 G0 X160 Z120
N50 T202
N60 G0 X160 Z60 M3 S1000
N70 G51.1 Z60 (tükrözés az X-szel párhuzamos, Z=60 koordinátájú tengelyre)
N80 M98 P101 (alprogram hívás)
N90 G50.1 Z0 (tükrözés kikapcsolása az X-szel párhuzamos tengelyen)
N100 G0 X160 Z120
N110 M30
```



15.3-1 ábra

Tükrözést bekapcsolni csak G50 állapotban lehet, azaz ha nincs léptékezési parancsállapot.

A tükrözés bekapcsolt állapotában viszont a léptékezés bekapcsolható.

A tükrözésre is érvényes, hogy a léptékezési parancsokkal nem lapolódhat át, tehát először a a léptékezést kell kikapcsolni, és csak utána a tükrözést.

```
G51.1 ... (tükrözés bekapcsolása)
G51 ... (léptékezés bekapcsolása)
...
G50 ... (léptékezés kikapcsolása)
G50.1 ... (tükrözés kikapcsolása)
```


16 Automatikus geometriai számítások

16.1 Letörés és lekerekítés programozása

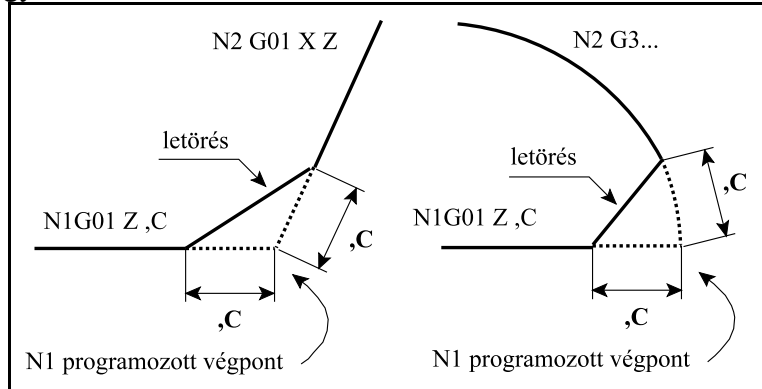
Két egyenes interpolációt (G01), vagy körinterpolációt (G02, G03) tartalmazó mondat közé a vezérlés automatikusan letörést, vagy lekerekítést tud beiktatni.

A

,C
(vessző és C) címen megadott értéknek megfelelő hosszúságú egyenlő szárú letörést iktat a ,C címet tartalmazó mondat végpontja és a következő mondat kezdőpontja közé. Pl.:

```
N1 G1 G91 Z30 ,C10
N2 X80 Z10
```

A ,C címen megadott érték azt mutatja meg, hogy a két egymást követő mondat feltételezett metszéspontjától mekkora távolságra kezdődik illetve fejeződik be a letörés. Letörést körök, vagy kör és egyenes közé is be lehet iktatni. Ekkor a ,C érték a metszésponttól húzott húr hossza.



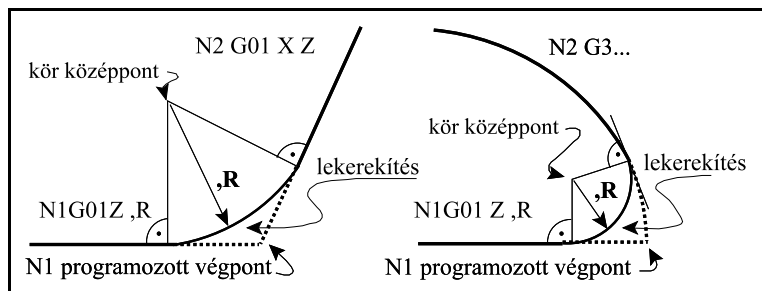
16.1-1 ábra

A

,R
(vessző és R) címen megadott értéknek megfelelő sugarú lekerekítést iktat a ,R címet tartalmazó mondat végpontja és a következő mondat kezdőpontja közé. Pl.:

```
N1 G91 G01 Z30 ,R8
N2 G03 X60 Z-30 R30
```

A ,R sugarú körívet úgy iktatja a két mondat közé, hogy a kör mindkét pályaelemhez érintőlegesen simuljon.



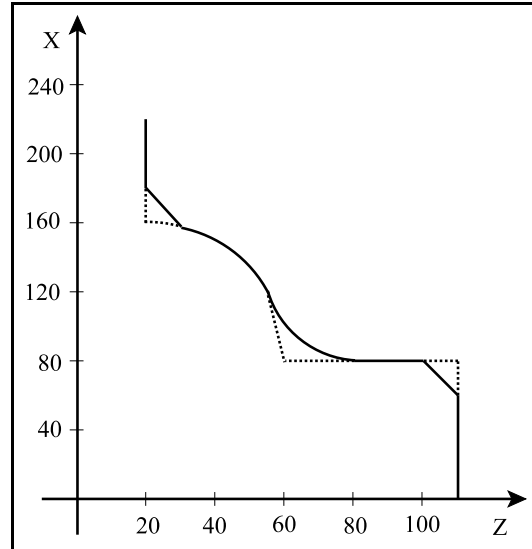
16.1-2 ábra

Egymást követő, több mondat végére is írható letörést, vagy lekerekítést tartalmazó utasítás, mint az alábbi példa mutatja:

```

...
G1 X80 ,C10
Z60 ,R22
G3 X160 Z20 R40 ,C10
G1 X220
...

```



16.1-3 ábra

☞ Megjegyzés:

- Letörés , vagy lekerekítés csak a kiválasztott síkban (G17, G18, G19) fekvő elemek között programozható, ellenkező esetben a vezérlő *3081 ,C ,R MEGADÁSI HIBA* üzenetet ad.
- Letörés , vagy lekerekítés csak G1, G2, vagy G3 mondatok között alkalmazható, ellenkező esetben a vezérlő *3081 ,C ,R MEGADÁSI HIBA* üzenetet ad.
- Ha a letörés szárhossza, vagy a lekerekítés sugara olyan nagy, hogy nem illeszthető a programozott mondatokhoz, a vezérlő *3084 ,C ,R TÚL NAGY* hibajelzést ad.
- Ha egy mondatba programozunk ,C-t és ,R-t a vezérlő *3017 ,C ÉS ,R EGY MONDATBAN* hibajelzést ad.
- Mondatonkénti üzemben a vezérlő a letörés, vagy a lekerekítés végrehajtása után áll meg, és vesz fel STOP állapotot.

16.2 Egyenes megadása irányszögével

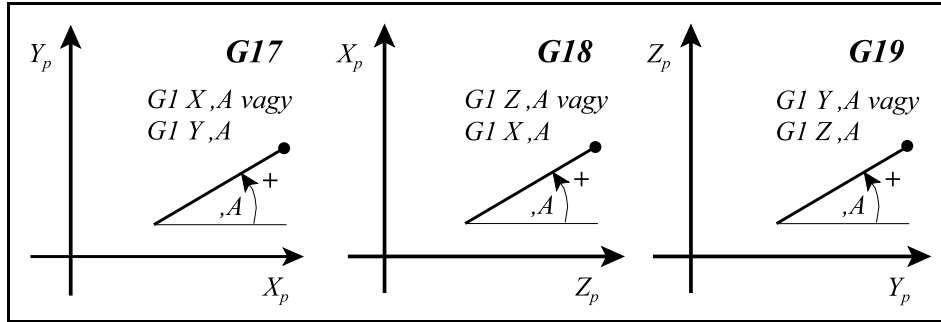
Egyenest a G17, G18, G19 utasítás által meghatározott síkban meg lehet adni a kiválasztott sík egyik koordinátájával és ,A címen megadva az egyenes irányszögével.

$$G17 \left\{ \begin{array}{l} G00 \\ G01 \end{array} \right\} \left\{ \begin{array}{l} X_p , A \\ Y_p , A \end{array} \right\} q F$$

$$G18 \left\{ \begin{array}{l} G00 \\ G01 \end{array} \right\} \left\{ \begin{array}{l} Z_p , A \\ X_p , A \end{array} \right\} q F$$

$$G19 \left\{ \begin{array}{l} G00 \\ G01 \end{array} \right\} \left\{ \begin{array}{l} Y_p , A \\ Z_p , A \end{array} \right\} q F$$

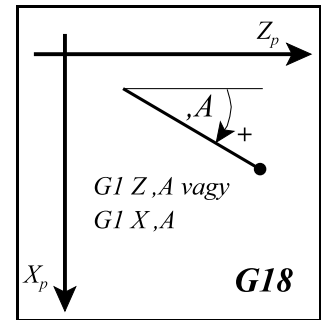
A fenti képletekben X_p, Y_p, Z_p az X, Y, Z, vagy a velük párhuzamos tengelyt jelöli, q tetszőleges egy, vagy több, a kiválasztott síkon kívül eső tengelyt jelöl. A ,A címen történő megadás G0 és G1 kód mellett is használható. ,A szög a kiválasztott sík első tengelyétől számítható, és a pozitív irány az óramutató járásával ellentétes. ,A értéke lehet pozitív és negatív is, valamint lehet 360° -nál nagyobb, illetve -360° -nál kisebb érték is.



16.2-1 ábra

☞ **Figyelem!**

– Ha az alkalmazott koordinátarendszer balsodrású az irányszög értelmezése a mellékelt ábra szerint (pozitív irány az óramutató járása szerinti) alakul.



16.2-2 ábra

Például:

(G18 G90) G1 X60 Z120 ...
 Z70 ,A150 (ez a megadás ekvivalens az
 X117.735 Z70 megadással)
 X180 ,A135 (ez a megadás ekvivalens az
 X180 Z38.868 megadással)

☞ **Megjegyzés:**

– Egy mondatban megadható egyenes irányszöggel és letörés, vagy lekerekítés is. Például:

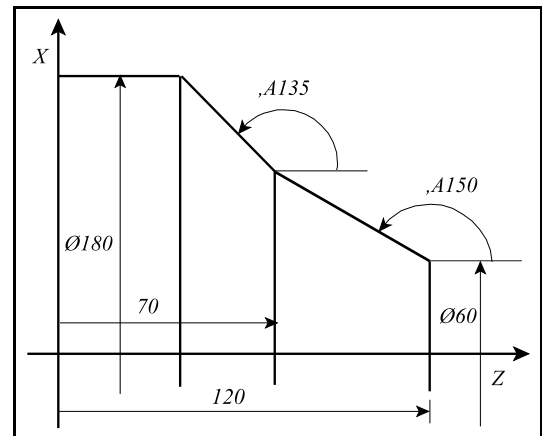
Z100 ,A30 ,C5
 X100 ,A120 ,R10
 Z-100 ,A210

– ,A címen történő irányszög megadása használható fűróciklusokban is. Ekkor a kiválasztott síkban történő pozicionálás végrehajtásakor veszi figyelembe a fent leírt módon. Például a

G81 G91 X100 ,A30 R-2 Z-25

mondat ekvivalens az alábbi mondattal:

G81 G91 X100 Y57.735 R-2 Z-25



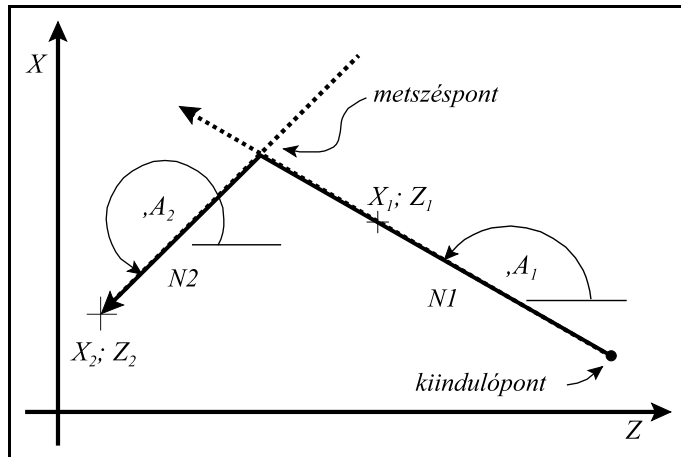
16.2-3 ábra

16.3 Síkbeli metszéspontszámítások

Az itt közölt metszéspontszámításokat a vezérlő csak a **szerszámsugár korrekció számítás bekapcsolt állapotában (G41, vagy G42)** végzi el. Ha esetleg a programban nem akarunk szerszámsugár korrekciót figyelembe venni, akkor is kapcsoljuk be azt és a megfelelő szerszám sugárkorrekciós értékét töröljük.

16.3.1 Két egyenes metszéspontja

Ha két, egymást követő, egyenes interpolációt kijelölő mondat közül a másodikat úgy adjuk meg, hogy definiálunk a kiválasztott síkban egy pontot *mindkét, a kijelölt síkba eső koordinátáját megadva*, amelyen az egyenes áthalad, és megadjuk az egyenes irányszögét is, a vezérlő kiszámítja az első mondatban kijelölt egyenes és a második mondatban megadott egyenes metszéspontját. A második mondatban így megadott egyenest túlhatározott egyenesnek nevezzük a továbbiakban. Az első mondat végpontja, illetve a második mondat kezdőpontja a kiszámított metszéspont lesz.



16.3.1-1 ábra

G17 G41 (G42)
N1 G1 ,A₁ vagy
X₁ Y₁
N2 G1G90 X₂ Y₂ ,A₂

G18 G41 (G42)
N1 G1 ,A₁ vagy
X₁ Z₁
N2 G1G90 X₂ Z₂ ,A₂

G19 G41 (G42)
N1 G1 ,A₁ vagy
Y₁ Z₁
N2 G1G90 Y₂ Z₂ ,A₂

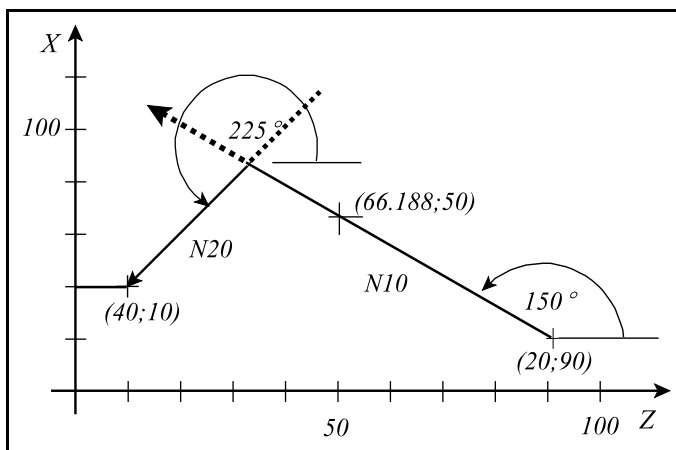
A metszéspontot mindig a G17, G18, G19 által kijelölt síkban számítja ki. Az első mondatot (N1) vagy csak irányszögével (A₁) adjuk meg, és ebben az esetben a kiindulópontból a megfelelő irányszögben húz egy egyenest a metszéspontig, vagy az egyenes egy tetszőleges, a kiindulóponttól különböző pontját adjuk meg (X₁, Y₁; X₁, Z₁; vagy Y₁, Z₁) és ekkor a két ponton áthaladó egyenessel számítja a metszéspontot. A *második mondatban* (N2) megadott *koordinátákat* mindig *abszolút* (G90) adatként értelmezi a vezérlő.

Például:

```
(G18) G90 G41 ...
G0 X20 Z90
N10 G1 ,A150
N20 X40 Z10 ,A225
G0 Z0
...
```

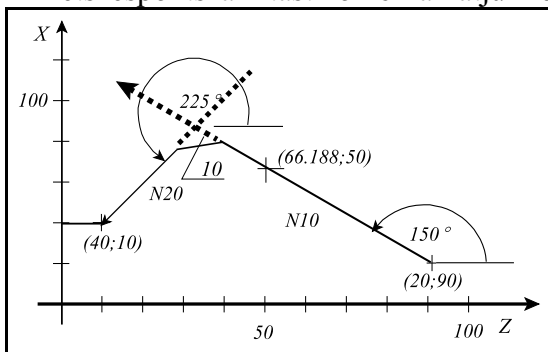
Az N10 mondatot megadhatjuk az egyenes egy pontjának koordinátaival is:

```
(G18) G90 G41 ...
G0 X20 Z90
N10 G1 X66.188 Z50
N20 X40 Z10 ,A225
G0 X0 Y20
...
```



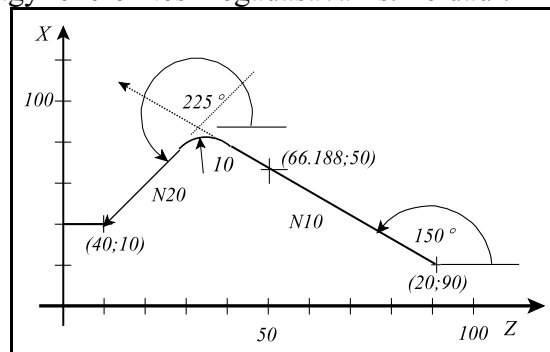
Figyeljük meg, hogy ebben az esetben az N10 mondatban megadott X, Z koordinátát (X66.18 Z50) nem végpontnak tekinti a vezérlő, hanem csak az egyenes kezdőpontját a megadott ponttal összekötő átmenő pontnak.

A metszéspontszámítást kombinálhatjuk letörés, vagy lekerekítés megadásával is. Például:



16.3.1-3 ábra

```
(G18) G90 G41 ...
G0 X20 Z90
N10 G1 X66.188 Z50 ,C10
N20 X40 Z10 ,A225
G0 X0 Y20
...
```



16.3.1-4 ábra

```
(G18) G90 G41 ...
G0 X20 Z90
N10 G1 X66.188 Z50 ,R10
N20 X40 Z10 ,A225
G0 X0 Y20
...
```

A fenti példákban a kiszámított metszésponttól méri vissza a letörés hosszát, illetve a kiszámított metszésponthoz igazítja a lekerekítést.

16.3.2 Egyenes és kör metszéspontja

Ha egyenes mondatot követően körmondatot úgy adunk meg, hogy a körnek megadjuk a végponti és középponti koordinátáját és a kör sugarát is, vagyis a kört túlhatározzuk, a vezérlés az egyenes és kör között metszéspontot számol. Az első mondat végpontja, illetve a második mondat kezdőpontja a kiszámított metszéspont lesz.

G17 G41 (G42)

N1 G1 ,A vagy

$X_1 Y_1$

N2 G2 (G3) G90 $X_2 Y_2 I J$

R Q

G18 G41 (G42)

N1 G₁,A vagy

$X_1 Z_1$

N2 G2 (G3) G90 $X_2 Z_2 I K$

R Q

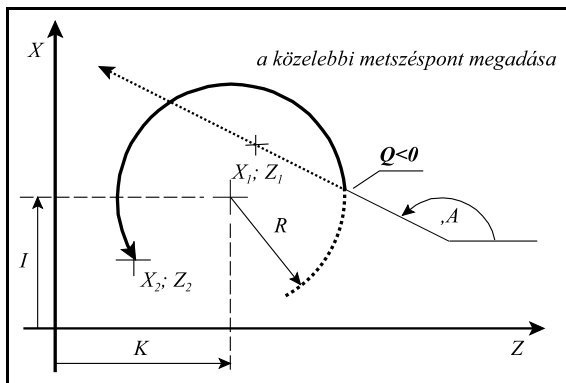
G19 G41 (G42)

N1 G1 ,A vagy

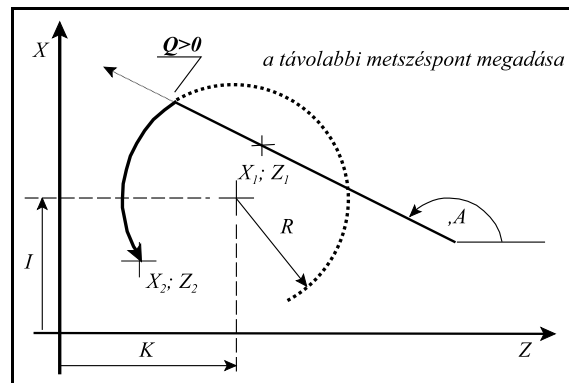
$Y_1 Z_1$

N2 G2 (G3) G90 $Y_2 Z_2 J K$

R Q



16.3.2-1 ábra

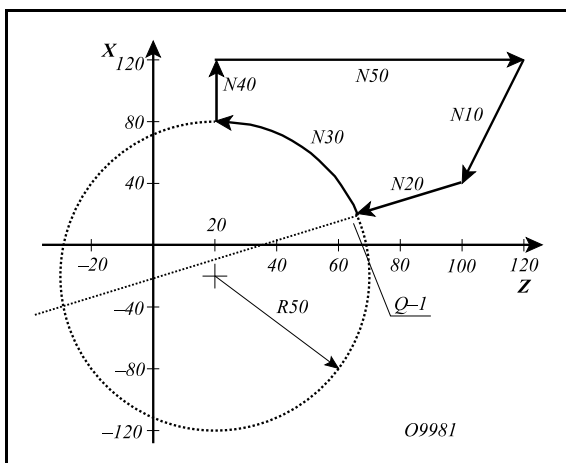


16.3.2-2 ábra

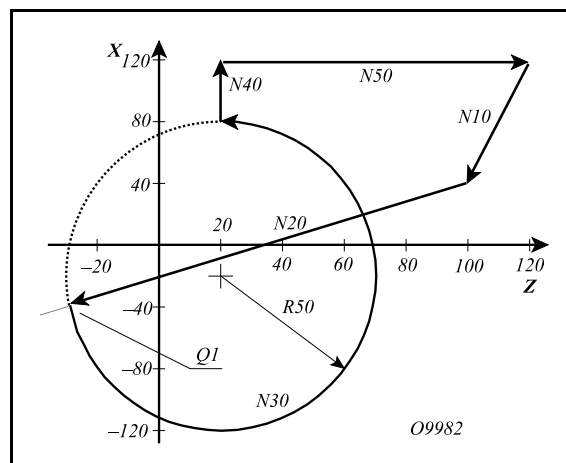
A metszéspontot mindig a G17, G18, G19 által kijelölt síkban számítja ki. Az első mondatot (N1) vagy csak irányszögével (,A) adjuk meg, és ebben az esetben a kiindulópontból a megfelelő irányszögben húz egy egyenest a metszéspontig, vagy az egyenes egy tetszőleges, a kiindulóponttól különböző pontját adjuk meg ($X_1, Y_1; X_1, Z_1;$ vagy Y_1, Z_1) és ekkor a két ponton áthaladó egyenessel számítja a metszéspontot. A második mondatban (N2) megadott koordinátákat, így a **kör középpontját** meghatározó **I, J, K koordinátákat** is mindig **abszolút** (G90) adatként értelmezi a vezérlő. Azt, hogy a kiadódó két metszéspont közül melyiket számolja ki a vezérlő a Q címen lehet megadni.

Ha a cím értéke kisebb, mint nulla ($Q < 0$) az egyenes irányába eső közelebbi, ha a cím értéke nagyobb, mint nulla ($Q > 0$) az egyenes irányába eső távolabbi metszéspontot számolja ki. Az egyenesen való haladás irányát az irányszög jelöli ki.

Nézzük a következő példát:



16.3.2-3 ábra



16.3.2-4 ábra

```
%O9981
N10 (G18) G42 G0 X40 Z100 S200 M3
N20 G1 X-40 Z-30
N30 G3 X80 Z20 I-10 K20 R50 Q-1
N40 G40 G0 X120
N50 Z120
N60 M30
%
```

```
%O9982
N10 (G18) G42 G0 X40 Z100 S200 M3
N20 G1 X-40 Z-30
N30 G3 X80 Z20 I-10 K20 R50 Q1
N40 G40 G0 X120
N50 Z120
N60 M30
%
```

Az N30 G3 körmondattól túlhatározott, mivel a középpont koordináták (I-10 K20 abszolút értékben), és a körsugár (R50) is meg van adva, a vezérlés az N20 mondatban megadott egyenes és az N30 mondatban megadott kör metszéspontját számolja. Az O9981 programban az egyenes irányába eső közelebbi metszéspontot számítja ki, mert az N30 körmondattal Q-1-et programoztunk. Az O9982 programban viszont az egyenes irányába eső távolabbi metszéspontot számítja ki, mert az N30 körmondattal Q1-et adtuk meg.

Az egyenes - kör metszéspontszámítást kombinálhatjuk letörés, vagy lekerekítés megadásával is. Például:

```
%O9983
N10 (G18) G42 G0 X40 Z100 S200 M3
N20 G1 X-40 Z-30 ,R15
N30 G3 X80 Z20 I-10 K20 R50 Q-1
N40 G40 G0 X120
N50 Z120
N60 M30
%
```

A vezérlő az N20 és N30 mondat metszéspontját kiszámolja és a metszésponthoz egy 15 mm sugarú lekerekítést illeszt az N20 mondatban megadott ,R15 hatására .

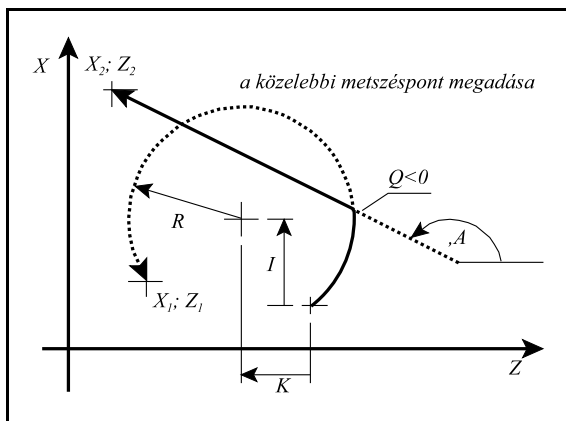
16.3.3 Kör és egyenes metszéspontja

Ha körmondatot követően egyenes mondatot úgy adunk meg, hogy az egyenest túlhatározzuk, azaz megadjuk az egyenes végponti koordinátáját és az irányszögét is, a vezérlés a kör és az egyenes között metszéspontot számol. Az első mondat végpontja, illetve a második mondat kezdőpontja a kiszámított metszéspont lesz.

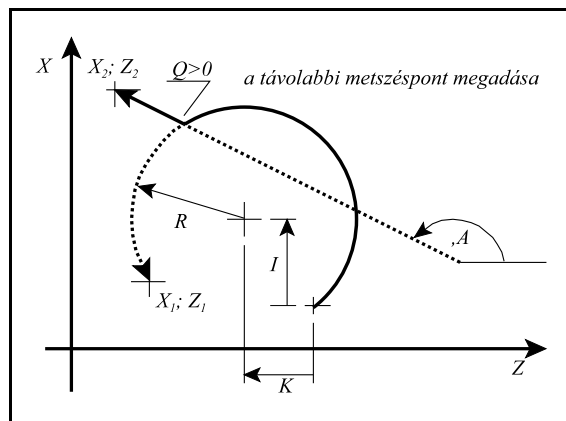
G17 G41 (G42)
N1 G2 (G3) X₁ Y₁ I J
vagy R
N2 G1 G90 X₂ Y₂ ,A Q

G18 G41 (G42)
N1 G2 (G3) X₁ Z₁ I K
vagy R
N2 G1 G90 X₂ Z₂ ,A Q

G19 G41 (G42)
N1 G2 (G3) Y₁ Z₁ J K
vagy R
N2 G1 G90 Y₂ Z₂ ,A Q



16.3.3-1 ábra

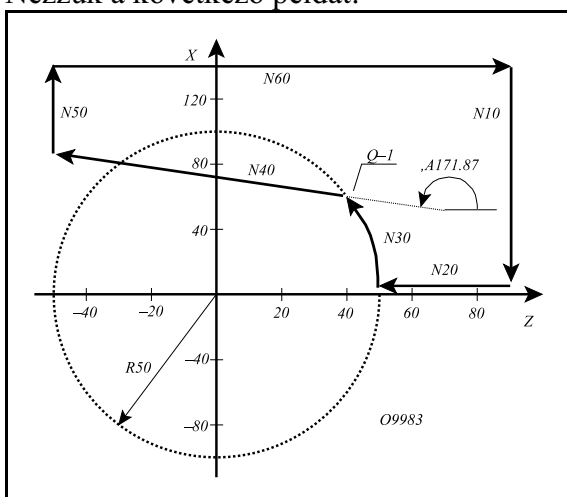


16.3.3-2 ábra

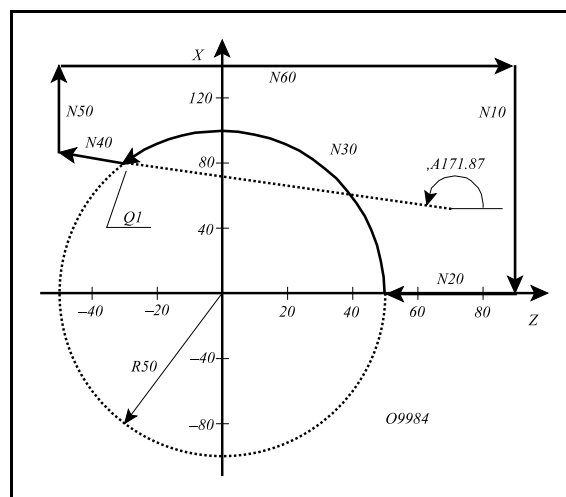
A metszéspontot mindig a G17, G18, G19 által kijelölt síkban számítja ki. Az első mondatot (N1), vagyis a kört egy tetszőleges pontjával (X₁, Y₁; X₁, Z₁; vagy Y₁, Z₁) és a középponti koordinátájával (I J; I K; vagy J K) adjuk meg, vagy a középponti koordináta helyett megadhatjuk a kör sugarát (R) is. A második mondatban (N2) az egyenest túlhatározzuk, vagyis megadjuk az egyenes végponti koordinátáit (X₂ Y₂; X₂ Z₂; vagy Y₂ Z₂) és az egyenes irányszögét (,A). Az egyenes végponti koordinátáit mindig **abszolút** (G90) adatként értelmezi a vezérlő. Mindig a kiadódó **metszéspontból** a megadott **végpontba** mutató egyenes **vektor irányszögét** kell megadni ,A címen, ellenkező esetben a programozói szándékkal ellentétes mozgások következnek be. Azt, hogy a kiadódó két metszéspont közül melyiket számolja ki a vezérlő a Q címen lehet megadni.

Ha a cím értéke kisebb, mint nulla (Q<0, pl: Q-1) az egyenes irányába eső közelebbi, ha a cím értéke nagyobb, mint nulla (Q>0, pl: Q1) az egyenes irányába eső távolabbi metszéspontot számolja ki. Az egyenesen való haladás irányát az irányszög jelöli ki.

Nézzük a következő példát:



16.3.3-3 ábra



16.3.3-4 ábra

```
%O9983
N10 (G18) G0 X90 X0 M3 S200
N20 G42 G1 Z50
N30 G3 X0 Z-50 R50
N40 G1 X85.714 Z-50 ,A171.87 Q-1
N50 G40 G0 X140
N60 Z90
N70 M30
%
```

```
%O9984
N10 (G18) G0 X90 X0 M3 S200
N20 G42 G1 Z50
N30 G3 X0 Z-50 R50
N40 G1 X85.714 Z-50 ,A171.87 Q1
N50 G40 G0 X140
N60 Z90
N70 M30
%
```

Az N40 egyenes mondat túlhatározott, mert az egyenes végponti koordinátái (X85.714 Z-50) is és irányszöge is (,A171.87) meg van adva. Ezért az előző, N30 mondatban programozott kör X0 Z-50 koordinátáit nem tekinti végponti értékeknek, hanem csak egy pontnak, amin a kör áthalad, és a végpont a kiszámított metszéspont lesz. Az O9983 számú programban a haladási irány szerinti közelebbi metszéspontot adtuk meg (Q-1), míg az O9984-ben a haladási irány szerinti távolabbat (Q1).

Kör és egyenes metszéspontjának megadását kombinálhatjuk letörés, vagy lekerekítés megadásával. Például:

```
%O9983
N10 (G18) G0 X90 X0 M3 S200
N20 G42 G1 Z50
N30 G3 X0 Z-50 R50 ,R15
N40 G1 X85.714 Z-50 ,A171.87 Q-1
N50 G40 G0 X140
N60 Z90
N70 M30
%
```

Példánkban az N30 mondatban megadtunk egy 15 mm-es lekerekítést (,R15). A vezérlő kiszámítja az N30 és N40 mondat közötti metszéspontot, és az így kiadódó kontúrhoz beilleszti a programozott lekerekítést.

16.3.4 Két kör metszéspontja

Ha kéz egymást követő körmondatot úgy adunk meg, hogy a második körnek megadjuk a végponti és középponti koordinátáját és a sugarát is, vagyis a második kört túlhatározzuk, a vezérlés a két kör között metszéspontot számol. Az első mondat végpontja, illetve a második mondat kezdőpontja a kiszámított metszéspont lesz.

G17 G41 (G42)

N1 G2 (G3) X₁ Y₁ I₁ J₁
 vagy X₁ Y₁ R₁

N2 G2 (G3) G90 X₂ Y₂ I₂
 J₂ R₂ Q

G18 G41 (G42)

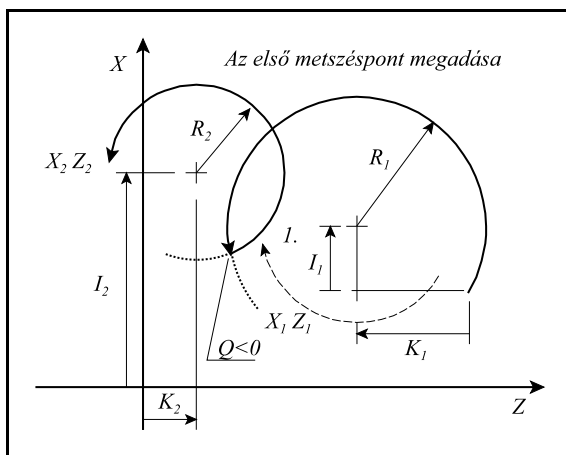
N1 G2 (G3) X₁ Z₁ I₁ K₁
 vagy X₁ Z₁ R₁

N2 G2 (G3) G90 X₂ Z₂ I₂
 K₂ R₂ Q

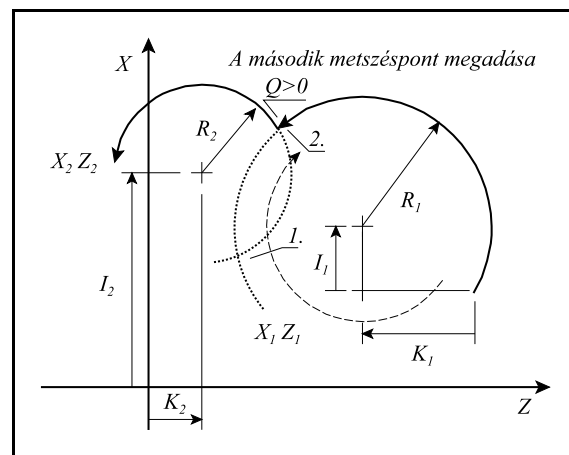
G19 G41 (G42)

N1 G2 (G3) Y₁ Z₁ J₁ K₁
 vagy Y₁ Z₁ R₁

N2 G2 (G3) G90 Y₂ Z₂ J₂
 K₂ R₂ Q



16.3.4-1 ábra

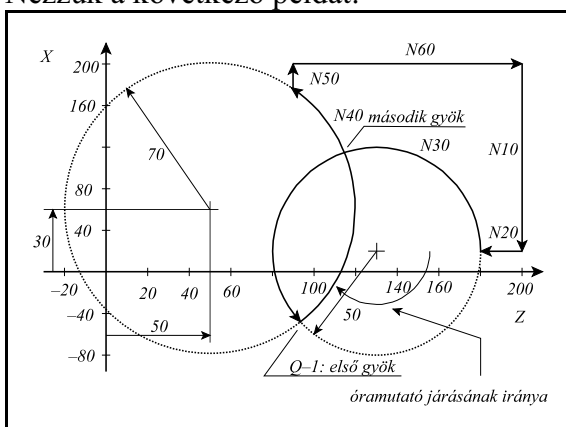


16.3.4-2 ábra

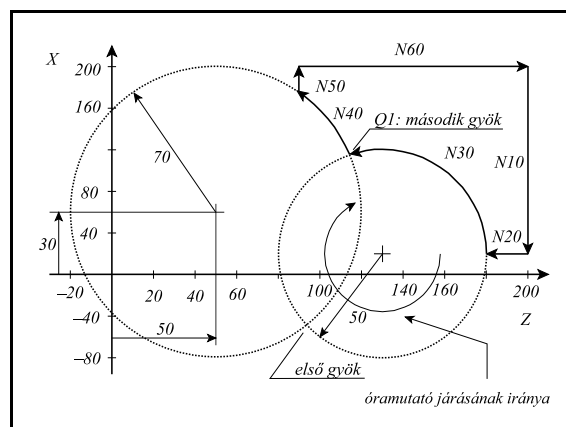
A metszéspontot mindig a G17, G18, G19 által kijelölt síkban számítja ki. Az első mondatot (N1) vagy a kör középponti koordinátájával (I₁ J₁; I₁ K₁; J₁ K₁), vagy a kör sugarával (R₁) adjuk meg. Ebben a mondatban a középponti koordináták értelmezése megegyezik a körmegadás alapértelmezésével, vagyis a kezdőponttól mért relatív távolság. A második mondatban (N2) megadott koordinátákat, így a **kör középpontját** meghatározó **I, J, K koordinátákat** is mindig **abszolút** (G90) adatként értelmezi a vezérlő. Azt, hogy a kiadódó két metszéspont közül melyiket számolja ki a Q címen lehet megadni. Ha a cím értéke kisebb, mint nulla (Q<0, pl: Q-1) az első, ha a cím értéke nagyobb, mint nulla (Q>0, pl: Q1) a második metszéspontot számolja ki.

Első az a metszéspont amelyiken az óramutató járásának irányában haladva (függetlenül a programozott G2, G3 iránytól) elsőnek haladunk át.

Nézzük a következő példát:



16.3.4-3 ábra



16.3.4-4 ábra

```
%O9985
N10 (G18) G0 X20 Z200 M3 S200
N20 G42 G1 Z180
N30 G3 X-80 Z130 R-50
N40 X174.892 Z90 I30 K50 R70 Q-1
N50 G40 G0 X200
N60 Z200
N70 M30
%
```

```
%O9986
N10 (G18) G0 X20 Z200 M3 S200
N20 G42 G1 Z180
N30 G3 X-80 Z130 R-50
N40 X174.892 Z90 I30 K50 R70 Q1
N50 G40 G0 X200
N60 Z200
N70 M30
%
```

Az N40 körmondattól túlhatározott, mert középpontjának koordinátái is (I30 K50 abszolút értékűként, valamint I sugárban értendő), és sugara is (R70) meg vannak adva. Ezért az előző, N30 mondatban programozott kör X-80 Z130 koordinátáit nem tekinti végponti értékeknek, hanem csak egy pontnak, amin a kör áthalad, és a végpont a kiszámított metszéspont lesz. Az O9985 számú programban az óramutató járási irány szerinti közelebbi metszéspontot adtuk meg (Q-1), míg az O9986-ban a távolabbit (Q1).

Két kör metszéspontjának megadását kombinálhatjuk letörés, vagy lekerekítés megadásával. Például:

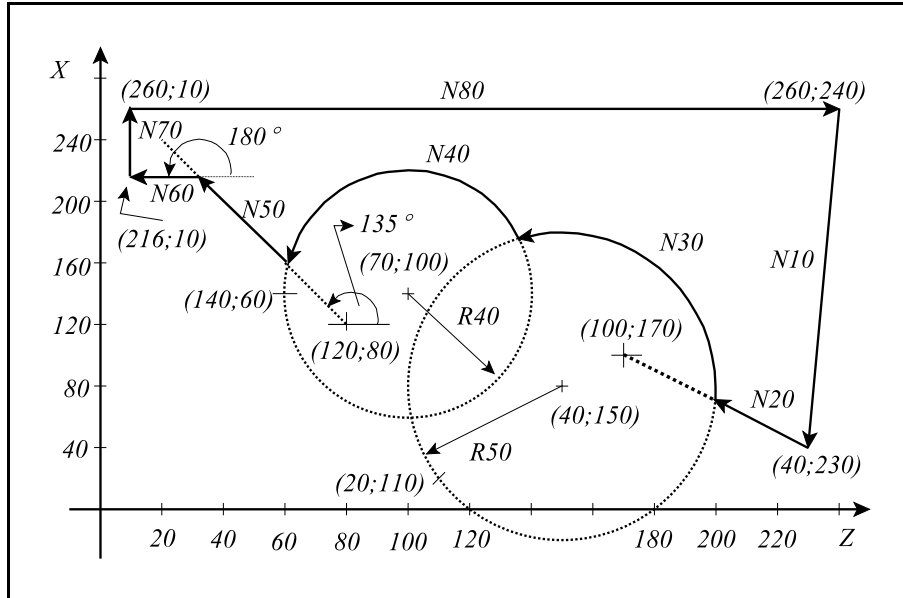
```
%O9986
N10 (G18) G0 X20 Z200 M3 S200
N20 G42 G1 Z180
N30 G3 X-80 Z130 R-50 ,R20
N40 X174.892 Z90 I30 K50 R70 Q1
N50 G40 G0 X200
N60 Z200
N70 M30
%
```

Példánkban az N30 mondatban megadtunk egy 20 mm-es lekerekítést (,R20). A vezérlő kiszámítja az N30 és N40 mondat közötti metszéspontot, és az így kiadódó kontúrhoz beilleszti a programozott lekerekítést.

16.3.5 A metszéspontszámítások láncolása

A **metszéspontszámító mondatokat lehet láncolni**, vagyis több, egymást követő mondatot is kijelölhetünk metszéspontszámításra. A vezérlő addig számít metszéspontot, amíg a programban túlhatározott egyeneseket, vagy köröket talál.

Tekintsük az alábbi példát:



16.3.5-1 ábra

```
%O9984
N10 (G18) G0 G42 X40 Z230 F300 S500 M3
N20 G1 X100 Z170
N30 G3 X20 Z110 I40 K150 R50 Q-1
N40 X140 Z60 I70 K100 R40 Q1
N50 G1 X120 Z80 ,A135 Q1
N60 X216 Z10 ,A180
N70 G40 G0 X260
N80 Z240
N90 M30
%
```

A fenti példában az N30, N40, N50, N60 mondat túlhatározott. Az N20 egyenest nem a programozott végpontjáig (X100 Z170) vezeti, mert az N30 körmondat túlhatározott, vagyis I K R címek mind ki vannak töltve, és Q címen megadtuk, hogy melyik metszéspontot keresse. Az N30 körmondatot sem a programozott végpontig (X20 Z110) vezeti, mert az N40 körmondat szintén túl van határozva. A programban az utolsó túlhatározott mondat az N60 egyenes. Mivel az utána következő N70 egyenes mondat nincs túlhatározva, ezért az N60 mondatban programozott X216 Z10 koordinátákat nem az egyenes egy átmenő pontjának, hanem az N60 mondat végponti koordinátáinak tekinti.

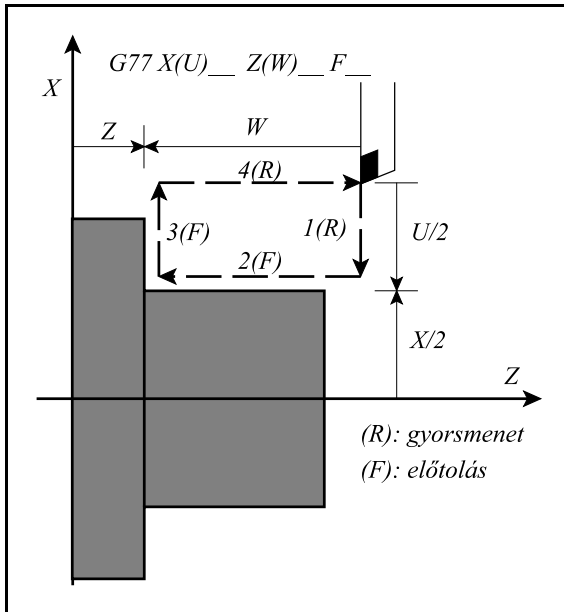
Általában elmondhatjuk, hogy a túlhatározott egyenes és kör mondatok kijelölt síkba eső koordinátapontjait csak akkor tekinti a vezérlő végponti koordinátának, ha utána már nem következik túlhatározott mondat.

17 Esztergáló ciklusok

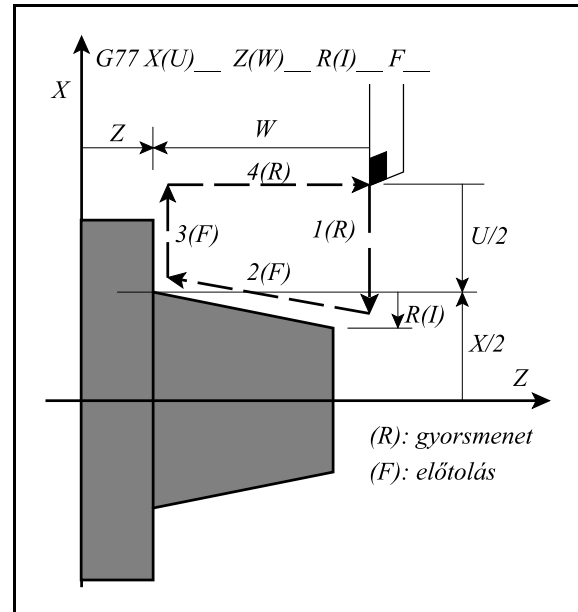
17.1 Egyszerű ciklusok

Az egyszerű ciklusok a G77 hosszsztergáló, a G78 egyszerű menetvágó és a G79 oldalazó ciklus.

17.1.1 A hosszsztergáló ciklus (G77)



17.1.1-1 ábra



17.1.1-2 ábra

Hengeres hosszsztergáló ciklust a következőképp adhatunk meg:

G77 X(U) Z(W) F

Inkrementális adatmegadás az I operátorral, illetve G91 programozásával is lehetséges.

Inkrementális adatmegadás esetén az adat előjele az 1-es és a 2-es pálya irányát határozza meg. Ábránkon úgy az U, mind a W cím előjele negatív.

A mondatban F címen programozott, vagy megörökölt előtolással mozog a 2-es és a 3-as pályán, gyorsmenettel az 1-esen és a 4-esen.

Kúpos hosszsztergáló ciklust a következőképp adhatunk meg:

G77 X(U) Z(W) R(I) F

A kúposágot vagy R, vagy I címen adhatjuk meg. Mindkét esetben az adat értelmezése ugyanaz.

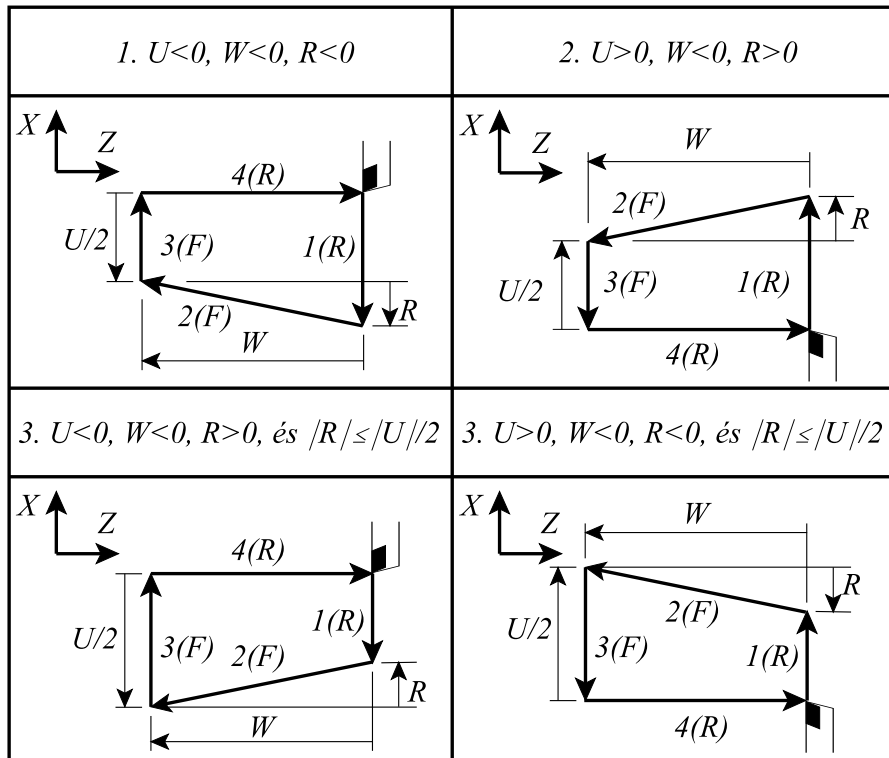
Az **R(I)** címen megadott adat **mindig inkrementális** adatként kerül értelmezésre, és az X(U) címen megadott pozíciótól értendő. Az R(I) cím előjele határozza meg a kúp lejtési irányát.

A többi cím értelmezése megegyezik a hengeres hosszsztergáló ciklusnál elmondottakkal.

A G77 kód, és a G77 mondatban programozott adatok öröklődőek.

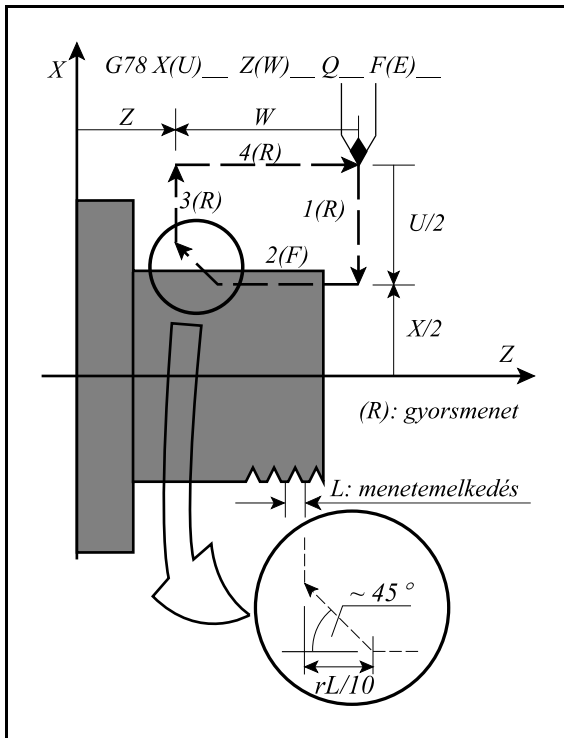
Mondatonkénti üzemben mind a négy művelet (1, 2, 3, 4) végén megáll.

Inkrementális programozás esetén U, W, és R(I) címek előjele az alábbiak szerint befolyásolják a mozgások irányát:

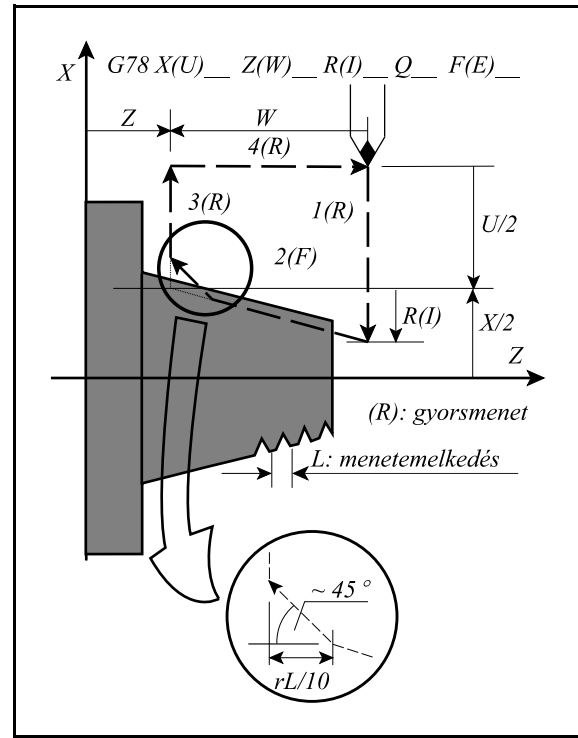


17.1.1-3 ábra

17.1.2 Az egyszerű menetvágó ciklus (G78)



17.1.2-1 ábra



17.1.2-2 ábra

Egyszerű menetvágó ciklust a következőképp adhatunk meg:

G78 X(U) Z(W) Q F(E)

Inkrementális adatmegadás az I operátorral, illetve G91 programozásával is lehetséges.

Inkrementális adatmegadás esetén az adat előjele az 1-es és a 2-es pálya irányát határozza meg.

Ábránkon úgy az U, mind a W cím előjele negatív.

A mondatban F címen programozzuk a menet emelkedését, vagy E címen az inchenkénti menetek számát, valamint Q címen a menet kezdetének a jeladó nullimpulzusától számított szögértét °-ban megadva, a G33 mondatban leírtak szerint.

Az 1, 3, 4 mozgás gyorsmenettel történik.

A 2-es pálya végén, ahol a menetvágás történik, egy kb 45°-os letörést végez.

A letörési szakasz hosszát az 1334 THRDCHMFR paraméter határozza meg, amit az ábrán r-rel jelöltünk. A szakasz hossza

$$r \cdot L / 10$$

ahol: r: a THRDCHMFR paraméter értéke

L: a programozott menetemelkedés

A THRDCHMFR paraméter értéke 1–255-ig terjedhet, vagyis a letörés mértéke 0.1L-től 25.5L-ig. Ha a paraméter értéke pl. 4 és a programozott menetemelkedés F2, akkor a letörés hossza:

$$2 * (4 / 10) = 0.8 \text{ mm}$$

Kúpos menetvágó ciklust a következőképp adhatunk meg:

G78 X(U)___ Z(W)___ R(I)___ Q___ F(E)___

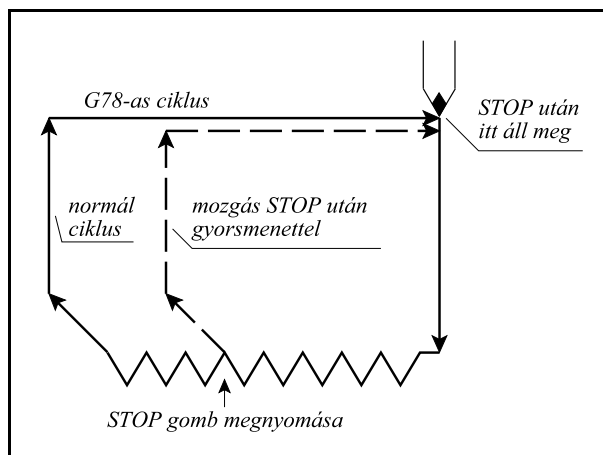
A kúposágot vagy R, vagy I címen adhatjuk meg. Mindkét esetben az adat értelmezése ugyanaz. Az **R(I)** címen megadott adat **mindig inkrementális** adatként kerül értelmezésre, és az X(U) címen megadott pozíciótól értendő. Az R(I) cím előjele határozza meg a kúp lejtési irányát. A többi cím értelmezése megegyezik a hengeres ciklusnál elmondottakkal. A letörés szöge ebben az esetben is 45° -os és a letörés hosszát r-t a tengelyekkel párhuzamos egyenes mentén mérjük.

A G78 kód, és a G78 mondatban programozott adatok öröklődőek. Mondatonkénti üzemben mind a négy művelet (1, 2, 3, 4) végén megáll.

A STOP gomb megnyomásának hatása a ciklus 2-es műveletében

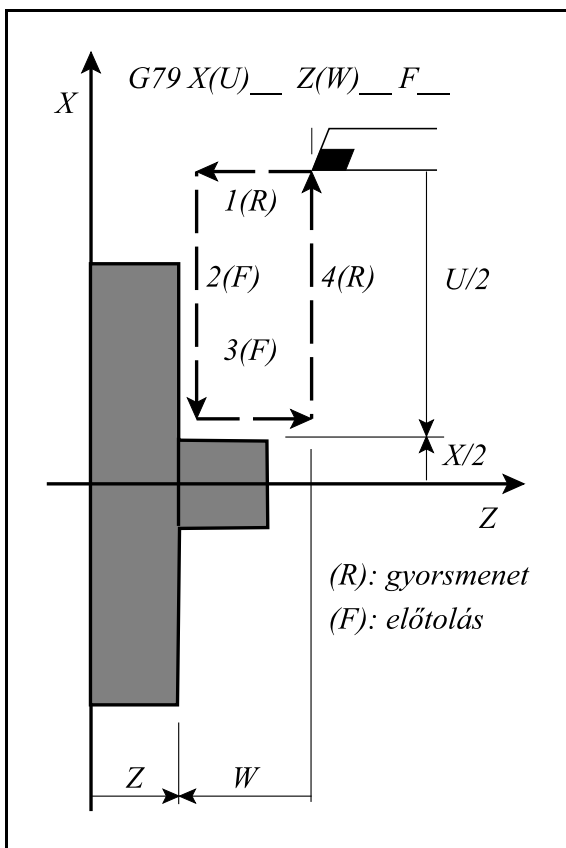
A ciklus 1-es, 3-as és 4-es műveletét a STOP gomb használatával bármikor meg lehet állítani és a szárok úgy állnak meg, mint normál G0 interpoláció esetén.

A 2-es menetvágó részben is hatásos a STOP gomb megnyomása, ám ekkor a vezérlés először megcsinálja ugyanazt a letörést, mint a 2-es művelet végén, azután először gyorsmenettel kiemel az X tengely mentén, majd Z-ben a kiindulási pontra áll. A menekülési útvonalon már nem hatásos a STOP gomb.

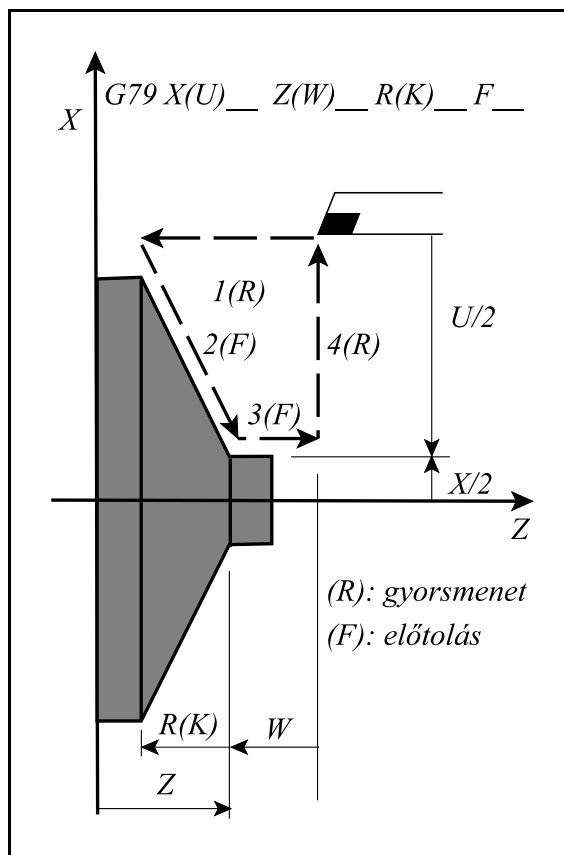


17.1.2 -3 ábra

17.1.3 Az oldalazó ciklus (G79)



17.1.3 -1 ábra



17.1.3 -2 ábra

Oldalazó ciklust a következőképp adhatunk meg:

G79 X(U) Z(W) F

Inkrementális adatmegadás az I operátorral, illetve G91 programozásával is lehetséges. Inkrementális adatmegadás esetén az adat előjele az 1-es és a 2-es pálya irányát határozza meg. Ábránkon úgy az U, mind a W cím előjele negatív.

A mondatban F címen programozott, vagy megörökölt előtolással mozog a 2-es és a 3-as pályán, gyorsmenettel az 1-esen és a 4-esen.

Kúpos oldalazó ciklust a következőképp adhatunk meg:

G79 X(U) Z(W) R(K) F

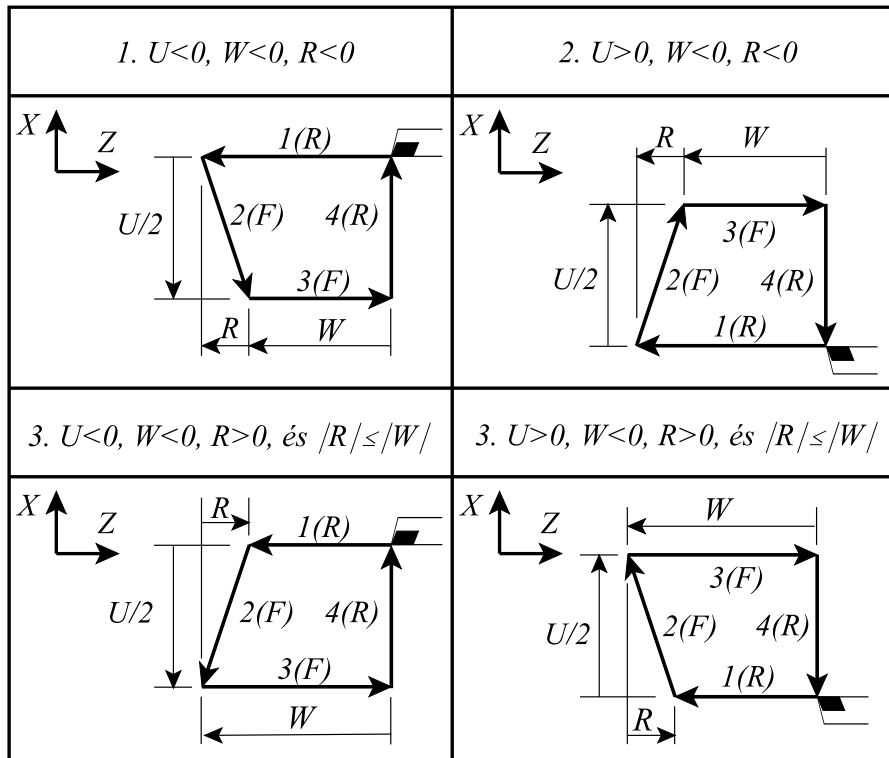
A kúposágot vagy R, vagy K címen adhatjuk meg. Mindkét esetben az adat értelmezése ugyanaz. Az **R(K)** címen megadott adat **mindig inkrementális** adatként kerül értelmezésre, és az X(U) címen megadott pozíciótól értendő. Az R(K) cím előjele határozza meg a kúp lejtési irányát.

A többi cím értelmezése megegyezik a hengeres hosszsztergáló ciklusnál elmondottakkal.

A G79 kód, és a G79 mondatban programozott adatok öröklődőek.

Mondatonkénti üzemben mind a négy művelet (1, 2, 3, 4) végén megáll.

Inkrementális programozás esetén U, W, és R(K) címek előjele az alábbiak szerint befolyásolják a mozgások irányát:



17.1.3 -3 ábra

17.1.4 Az egyszerű ciklusok használata

Úgy a ciklusok G kódja, mint a bemenő paraméterei öröklődnek. Ez azt jelenti, hogy, ha a ciklus változóit, X(U), Z(W), vagy R(I, vagy K), már megadtuk egyszer és értékük változatlan, nem kell újra beírni a programba azokat. Például:

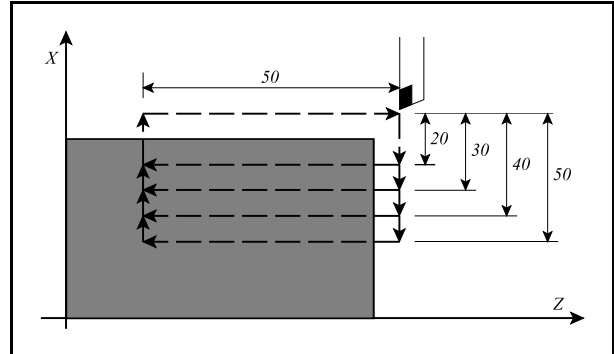
```
G91...
G77 X-20 Z-50 F0.5
X-30
X-40
X-50
...
```

A fenti példában csak a fogásvétel értéke (X) változik ezért csak ezt a címet kell újra kitölteni, a többi értéke változatlan marad.

Ciklust csak akkor hajt végre a ciklus bekapcsolt állapotában, ha valamelyik mozgásra utaló változó, X(U), Z(W), vagy R(I, vagy K), is ki van töltve. Ha például ciklusállapotban

önálló mondatban funkciót hajtunk végre a ciklusállapot bekapcsolva marad, de a ciklust nem ismétli meg:

```
...
G77 U-20 W-50 F0.5      (ciklust bekapcsolja és végrehajtja)
T202                    (ciklus bekapcsolva de nem hajtja végre)
U-30                    (ciklust végrehajtja)
...
```



17.1.4 -1 ábra

A ciklust és az öröklődő változókat az 1-es csoportba tartozó interpolációs G kódok törlik, illetve a G4 várakozás kivételével az összes egylövetű G kód.

Az egyszerű ciklusokat tartalmazó mondatokba írható M, S, T funkció is. A funkciók mindig a ciklus 1-es műveletében hajtódnak végre, vagy a mozgással párhuzamosan, vagy a mozgás végén. Ha ez bizonyos esetekben kellemetlen a funkciót írjuk külön mondatba.

17.2 Összetett ciklusok

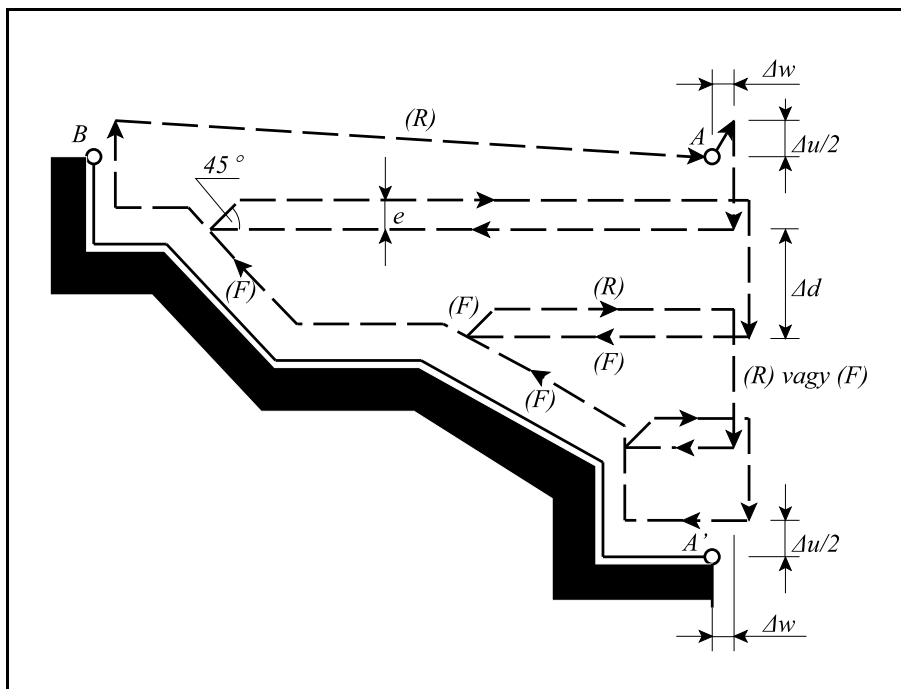
Az összetett ciklusok leegyszerűsítik a munkadarab program írását. Például a munkadarab kész méretének kontúrját a símitáshoz le kell írni. Ez a kontúr egyben meghatározza az alapját a munkadarab nagyolását végző ciklusoknak (G71, G72, G73). A nagyoló ciklusokon kívül rendelkezésre áll egy símitó (G70), egy menetvágó (G76) és két beszűrő ciklus (G74, G75) is.

17.2.1 Nagyoló ciklus (G71)

Két nagyoló ciklus van az 1-es és a 2-es típusú.

Az 1-es típusú nagyoló ciklus

Ha adott egy munkadarab kész méretének kontúrja, amit az ábrán az A–A'–B pontokkal jelöltünk, a G71-es ciklus Δd nagyságú fogásokkal kinagyolja a nyers darabot $\Delta u/2$ és Δw nagyságú símitási ráhagyással.



17.2.1 -1 ábra

1. Megadási módszer:

G71 U(Δd) R(e)

G71 P(n_s) Q(n_f) U(Δu) W(Δw) F(f) S(s) T(t)

N(n_s) X(U) ...

...

F ___

S ___

T ___

N(n_f) ...

utasítással történik, ahol:

- Δd :** fogásmélység. Mindig **sugárban** értendő **pozitív** szám. A fogásmélység mértékét az 1339 DPTH CUT paraméteren is meg lehet adni, illetve ez a paraméter kerül átírásra a program utasítás hatására. Ez egyben azt is jelenti, hogy, ha a fogásmélység értékét nem adjuk meg, a vezérlő azt erről a paraméterről veszi.
- e:** a kiemelés mértéke. Mindig **sugárban** értendő **pozitív** szám. A kiemelés mértékét az 1340 ESCAPE paraméteren is meg lehet adni, illetve ez a paraméter kerül átírásra a program utasítás hatására. Ez egyben azt is jelenti, hogy, ha a kiemelés értékét nem adjuk meg, a vezérlő azt erről a paraméterről veszi.
- n_s :** a program símitást leíró részének (A–A'–B szakasz) kezdő mondatszám.
- n_f :** a program símitást leíró részének (A–A'–B szakasz) befejező mondatszám.
- Δu :** a símitási ráhagyás nagysága és iránya az X tengely mentén. Az X koordináta értelmezésének függvényében **átmérőben**, vagy **sugárban** értendő **előjeles** szám.
- Δw :** a símitási ráhagyás nagysága és iránya az Z tengely mentén. **Előjeles** szám.
- f, s, t:** a ciklus folyamán a program símitást leíró n_s -től n_f -ig tartó részében (A–A'–B szakasz) programozott F, S, T funkciókat nem hajtja végre, hanem a G71 mondatban megadott f, s, t értékeket érvényesíti.

Azt, hogy az U címen megadott érték jelentése Δd , vagy Δu az dönti el, hogy az adott mondatban programoztuk-e P-t és Q-t. Ha nem az U cím jelentése Δd , ha igen az U cím jelentése Δu .

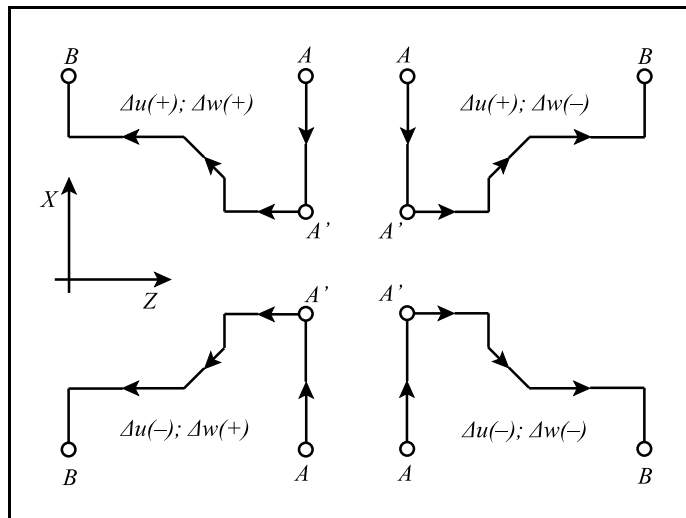
A nagyoló ciklust az a mondat hajtja végre, amelyikben P-t és Q-t adtuk meg.

Az A–A' pontok közötti mozgást a P címen megadott n_s számú mondatban kell megadni **kötelezően G00**, vagy **G01** programozásával. Az itt megadott kód dönti el, hogy a nagyolás során a **fogásvétel** (az A–A' irányú mozgás) gyorsmenettel (**G00** programozása esetén), vagy elótolással (**G01** programozása esetén) történjék. Ebben a mondatban P(n_s) Z irányú mozgást **nem szabad** megadni.

Az A'–B szakasz a tulajdonképpeni, egyenesekből és körívekből álló kontúr. A kontúrnak úgy X, mind Z irányban monoton növekvőnek, vagy csökkenőnek kell lennie, ami azt jelenti, hogy visszafordulás egyik irányban sem lehetséges. A ciklus mind a négy síknegyedben használható. A mellékelt ábra a símitási ráhagyás előjelét is feltünteti. Az n_s és n_f mondatok közötti programrészben programozott F, S, T funkciókat figyelmen kívül hagyja és azokat érvényesíti, amelyeket a G71 mondatban (f, s, t), vagy előbb programoztuk. Ugyanez vonatkozik az n_s és n_f mondatok között programozott konstans vágósebességre is, vagyis a G71 mondat előtti G96, vagy G97 állapotot, és konstans vágósebesség értéket érvényesíti.

Az n_s -től n_f -ig terjedő mondatok nem tartalmazhatnak alprogramhívást.

A szerszám sugár korrekció számítás (G41, G42) bekapcsolható a ciklus végrehajtása alatt azzal



17.2.1 -2 ábra

a megkötéssel, hogy n_s -től n_f -ig terjedő mondatok között kell bekapcsolni (G41, vagy G42) és kikapcsolni (G40) a sugár korrekció számítását:

HELYYES	HELYTELEN
$ \begin{array}{l} N(n_s) \quad X(U) \quad G41 \quad \dots \\ \quad \quad (G41) \quad \dots \\ \quad \quad \dots \\ \quad \quad (G40) \\ N(n_f) \quad G40 \quad \dots \end{array} $	$ \begin{array}{l} \quad \quad \quad \quad \quad G41 \\ N(n_s) \quad X(U) \quad \dots \\ \quad \quad \quad \quad \quad \dots \\ \quad \quad \quad \quad \quad \dots \\ \quad \quad \quad \quad \quad G40 \\ N(n_f) \quad \dots \end{array} $
vagy	vagy
$ \begin{array}{l} \quad \quad \quad \quad \quad G41 \\ N(n_s) \quad X(U) \quad \dots \\ \quad \quad \quad \quad \quad \dots \\ \quad \quad \quad \quad \quad \dots \\ N(n_f) \quad \dots \\ \quad \quad \quad \quad \quad G40 \end{array} $	$ \begin{array}{l} N(n_s) \quad G41 \quad X(U) \quad \dots \\ \quad \quad \quad \quad \quad \dots \\ \quad \quad \quad \quad \quad \dots \\ N(n_f) \quad \dots \\ \quad \quad \quad \quad \quad G40 \end{array} $

Ha a ciklust futás közben megszakítjuk, szerkesztés üzemmódba lépünk, és az 1339 DPTHCUT paramétert átírjuk, majd automatában START-tal elindítjuk a programot, a következő fogásvételt már az új fogásmélységgel veszi. Ugyanez vonatkozik az 1340 ESCAPE paraméterre, vagyis a kiemelés nagyságára.

2. Megadási módszer:

G71 P(n_s) Q(n_f) U(Δu) W(Δw) D(Δd) F(f) S(s) T(t)
 $N(n_s) \quad X(U) \quad \dots$
 \dots
 $\quad \quad \quad \quad \quad \underline{F}$
 $\quad \quad \quad \quad \quad \underline{S}$
 $\quad \quad \quad \quad \quad \underline{T}$
 $N(n_f) \quad \dots$

A második megadási módszer bemenő paraméterei megegyeznek az elsőével.

A 2-es típusú nagyológiklus

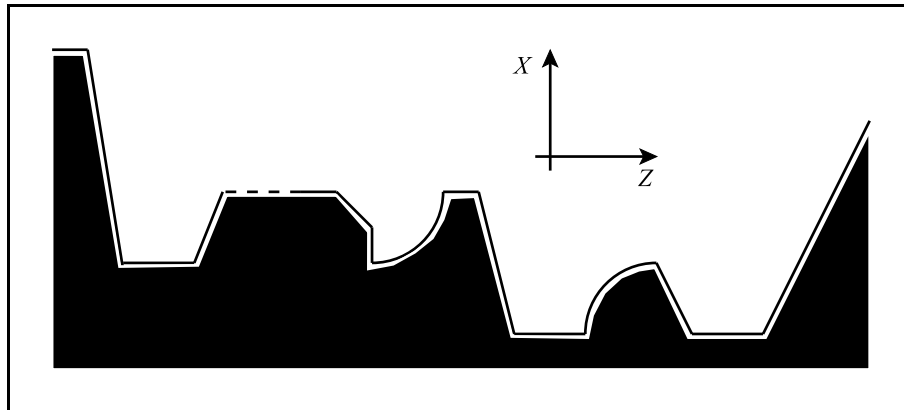
A 2-es típusú nagyoló ciklust ugyanúgy kell megadni, mint az 1-es típusút, a kódja G71 és a bemenő paraméterei is ugyanazok, mint az 1-es típusúnak. A különbség a kontúr kezdő mondatának (n_s számú mondat) megadásában van. Amíg az 1-es típus hívása esetén ebben a mondatban Z címre nem lehet hivatkozni, vagyis az A–A' szakasz mozgásának merőlegesnek kell lennie a Z tengelyre, addig a 2-es típus hívása esetén ebben a mondatban kötelezően hivatkozni kell Z címre. Az A–A' szakasznak tehát nem kell merőlegesnek lennie a Z tengelyre.

1-es típus megadása	2-es típus megadása
$ \begin{array}{l} G71 \quad U8 \quad R1 \\ G71 \quad P100 \quad Q200 \quad U0.5 \quad W0.2 \\ N100 \quad \mathbf{X(U)} \quad \underline{\quad} \\ \dots \\ \dots \\ \dots \\ N200 \end{array} $	$ \begin{array}{l} G71 \quad U8 \quad R1 \\ G71 \quad P100 \quad Q200 \quad U0.5 \quad W0.2 \\ N100 \quad \mathbf{X(U)} \quad \underline{\quad} \quad \mathbf{Z(W)} \quad \underline{\quad} \\ \dots \\ \dots \\ \dots \\ N200 \end{array} $

Abban az esetben, ha 2-es típusú ciklust kell használnunk, ám a kontúrt bevezető mondatban csak X irányban kell mozognunk, vagyis merőlegesen a Z tengelyre, programozzunk a Z tengely mentén inkrementális 0 elmozdulást, azaz ZI0-t, vagy W0-t.

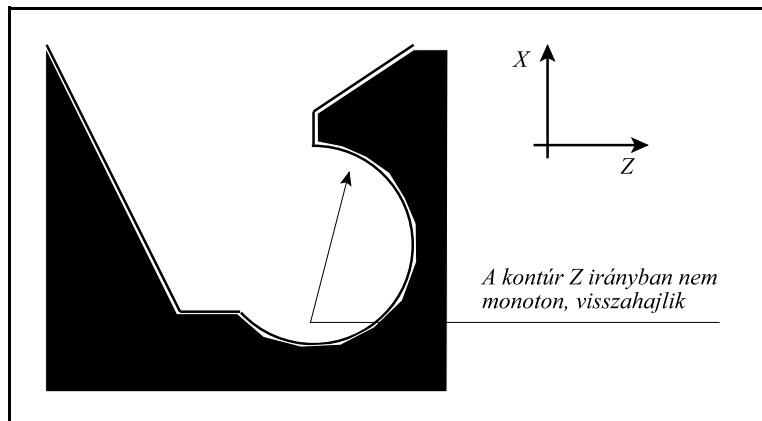
A 2-es típusú nagyológiklus abban különbözik az 1-estől, hogy a kontúrnak X irányban nem kell

monoton növekvőnek, vagy csökkenőnek lennie, vagyis a kontúr lehet visszahajló. A ciklus maximum 10 db visszahajló zsebet tud kezelni.



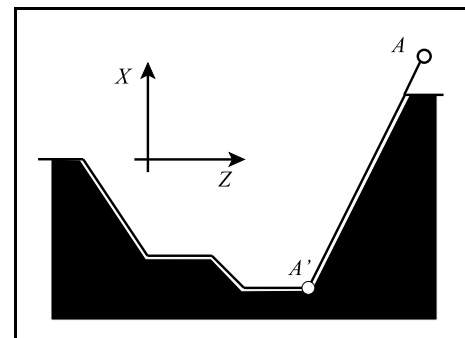
17.2.1 -3 ábra

Z irányban viszont a kontúrnak továbbra is **monoton**nak kell lennie, nem lehet visszahajlás benne.



17.2.1 -4 ábra

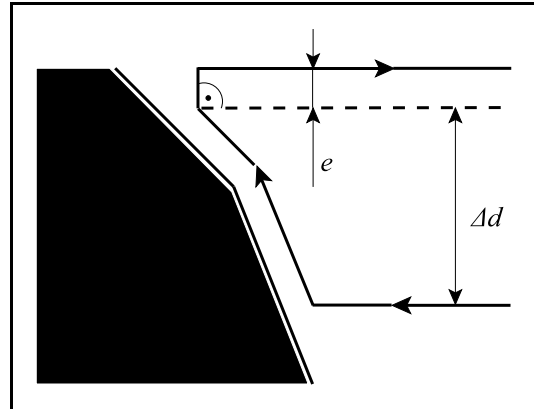
A kontúr első mondata (n_s) tartalmazhat Z irányú elmozdulást is (sőt Z címre kötelezően hivatkozni kell), vagyis az első vágásnak nem kell merőlegesnek lennie a Z tengelyre.



17.2.1 -5 ábra

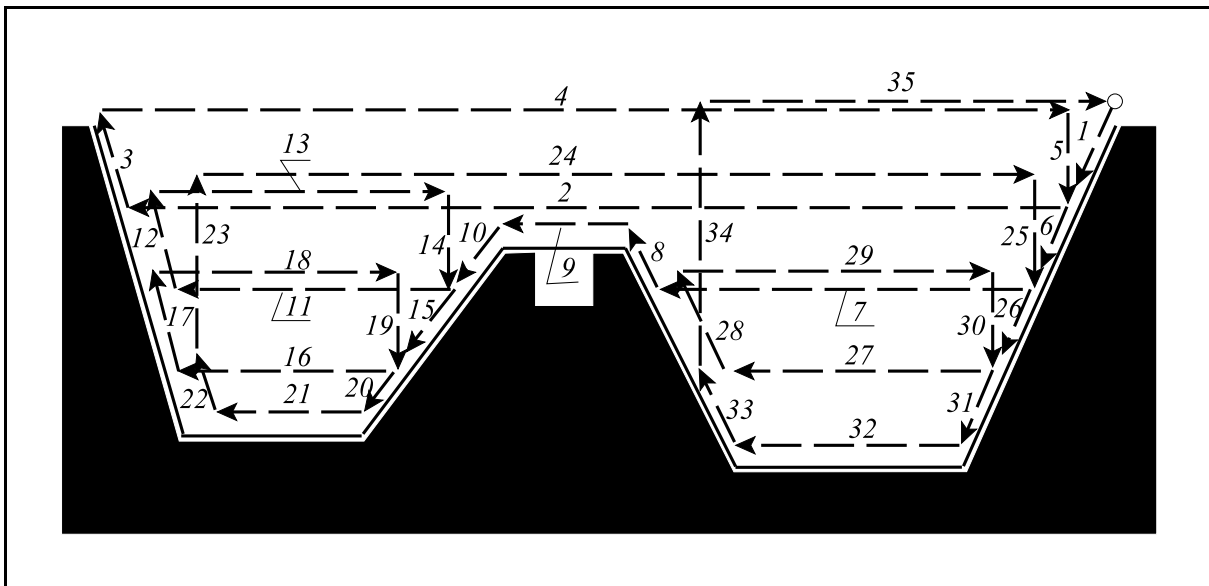
17.2.1 Nagyoló ciklus (G71)

A 2-es típusú nagyolóciklusnál a kiemelés a Z tengelyre merőlegesen történik, az érvényes kiemelési "e" értékkel.



17.2.1 -6 ábra

Az alábbi ábrán közlünk egy példát arra, hogyan bontja ki a ciklus a nyers darabot:



17.2.1 -7 ábra

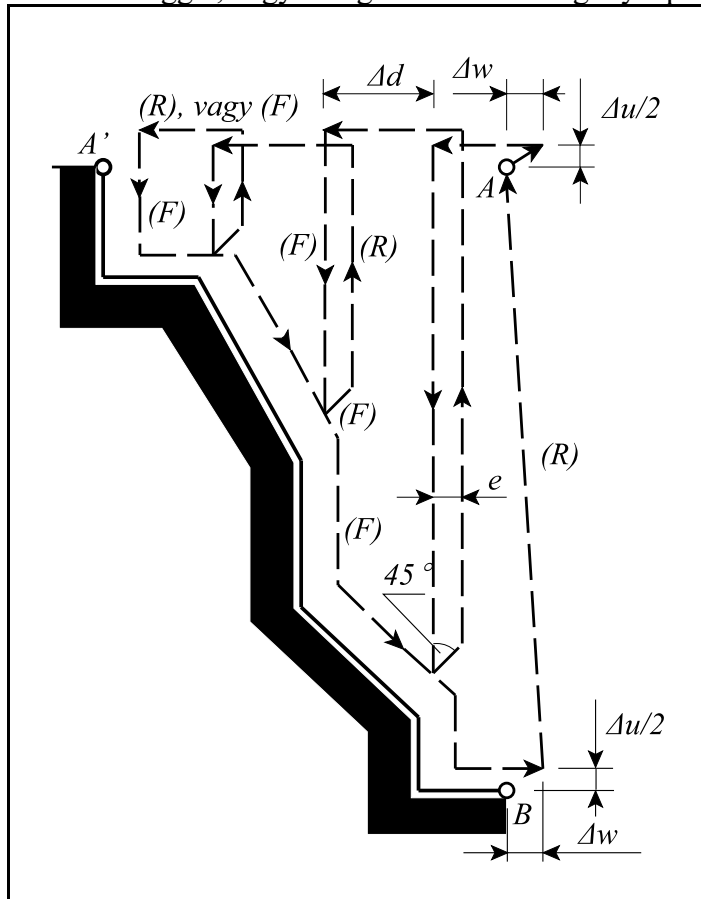
A símitási ráhagyás programozásánál a fenti esetben Z irányú símitási ráhagyást (Δw) nem szabad programozni (W0 kell legyen), mert ellenkező esetben a szerszám belevághat az egyik oldali falba.

17.2.2 Homlok nagyoló ciklus (G72)

Két nagyoló ciklus van az 1-es és a 2-es típusú.

Az 1-es típusú homlok nagyoló ciklus

A homlok nagyoló ciklus (G72), amint az az alábbi ábrán is látható ugyanaz, mint a G71-es nagyoló ciklus, azzal a különbséggel, hogy a forgácsolás az X tengellyel párhuzamosan történik.



17.2.2 -1 ábra

1. Megadási módszer:

G72 **W**(Δd) **R**(e)

G72 **P**(n_s) **Q**(n_f) **U**(Δu) **W**(Δw) **F**(f) **S**(s) **T**(t)

N(n_s) **Z**(W) ...

...

F ___

S ___

T ___

N(n_f) ...

A bemenő paraméterek jelentése teljes egészében megegyezik a G71 ciklusnál elmondottakkal.

2. Megadási módszer:

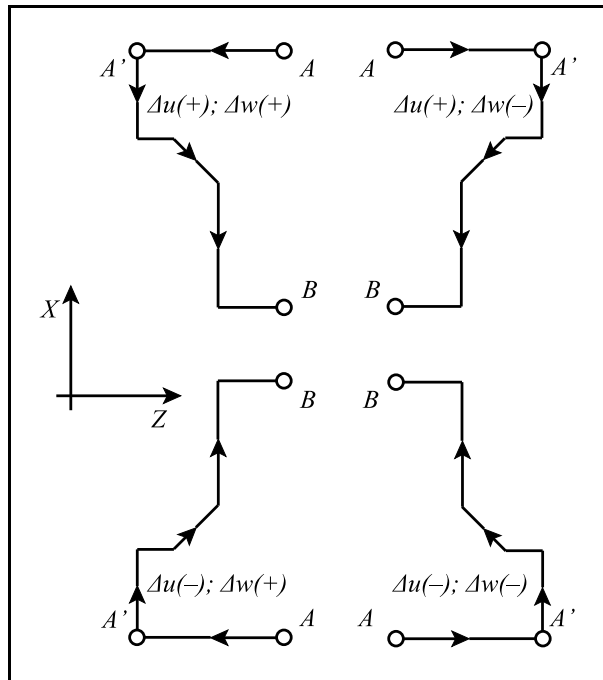
G72 P (n_s) Q (n_f) U(Δu) W(Δw) D(Δd) F(f) S(s) T(t)
 N(n_s) Z(W) ...
 ...
 F ___
 S ___
 T ___
 N(n_f) ...

A ciklus mind a négy síknegyedben használható. Az ábra a simítási ráhagyás előjelét is megadja mind a négy esetre.

Az A–A' pontok közötti mozgást leíró n_s számú mondatban X tengelyre való hivatkozás nem lehet, a mozgás mindig a Z tengellyel párhuzamos kell legyen. Az n_s számú mondatban meghatározott interpolációs kód (G00, vagy G01) határozza meg, hogy a ciklus során a fogásvétel előtolással, vagy gyorsmenettel történik.

A programozott kontúrnak mindkét tengely mentén monotonnak, vagyis folytonosan növekvőnek, vagy csökkenőnek kell lenni.

A szerszámsugár korrekció használatára a G71 ciklusban elmondottak érvényesek.



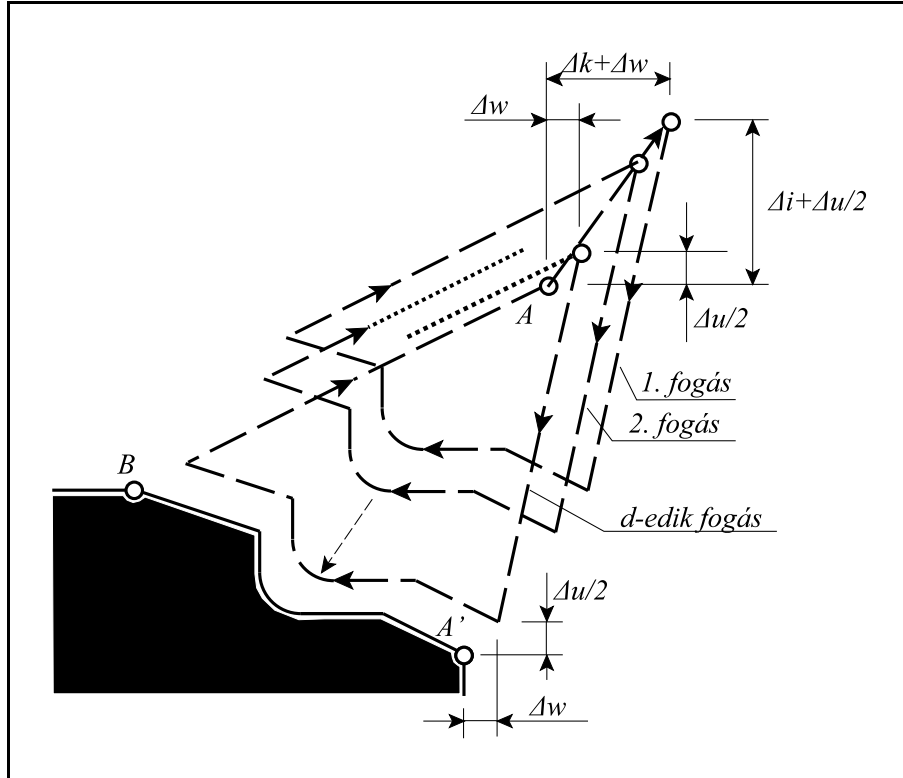
17.2.2 -2 ábra

A 2-es típusú homlok nagyoló ciklus

A 2-es típusú homlok nagyoló ciklust ugyanúgy kell megadni, mint az 1-es típusút, a kódja G72 és a bemenő paraméterei is ugyanazok, mint az 1-es típusúnak. A különbség a kontúr kezdő mondatának (n_s számú mondat) megadásában van. Amíg az 1-es típus hívása esetén ebben a mondatban X címre nem lehet hivatkozni, vagyis az A–A' szakasz mozgásának merőlegesnek kell lennie az X tengelyre, addig a 2-es típus hívása esetén ebben a mondatban kötelezően hivatkozni kell X címre. Az A–A' szakasznak tehát nem kell merőlegesnek lennie az X tengelyre. A továbbiakban a ciklusra ugyanazok a megkötések érvényesek, mint a G71-es 2-es típusú ciklusra. A kontúr lehet visszahajló is, azonban X irányban monotonnak kell lennie.

17.2.3 Minta ismétlődő ciklus (G73)

Ez a ciklus előkovácsolt, öntött, vagy előnagyolt darabok nagyolásánál alkalmazható, ahol a végleges forma körvonalai már rendelkezésre állnak. A ciklus fogásról fogásra ismét egy, a programban leírt kontúrt.



17.2.3 -1 ábra

1. Megadási módszer:

G73 U(Δi) W(Δk) R(d)

G73 P(n_s) Q(n_f) U(Δu) W(Δw) F(f) S(s) T(t)

N(n_s) ...

...

F ___

S ___

T ___

N(n_f) ...

utasítással történik, ahol:

Δi : a nagyolási ráhagyás nagysága és iránya az X tengely mentén. Mindig **sugárban** értendő **előjeles** szám. A ráhagyás mértékét az 1341 RELIEFX paraméteren is meg lehet adni, illetve ez a paraméter kerül átírásra a program utasítás hatására.

Δk : a nagyolási ráhagyás nagysága és iránya a Z tengely mentén. Mindig **előjeles** szám. A ráhagyás mértékét az 1342 RELIEFZ paraméteren is meg lehet adni, illetve ez a paraméter kerül átírásra a program utasítás hatására.

d: a fogások száma. A fogások számát az 1343 NUMDIV paraméteren is meg lehet adni, illetve ez a paraméter kerül átírásra a program utasítás hatására. A nagyolási ráhagyáson

megadott értéket (RELIEFX, RELIEFZ paraméter) ezzel a számmal osztja, és a nagyolás során a fogásokat az így kiadódó értékkel veszi.

n_s : a program símitást leíró részének (A–A'–B szakasz) kezdő mondatszám.

n_f : a program símitást leíró részének (A–A'–B szakasz) befejező mondatszám.

Δu : a símitási ráhagyás nagysága és iránya az X tengely mentén. Az X koordináta értelmezésének függvényében **átmérőben**, vagy **sugárban** értendő **előjeles** szám.

Δw : a símitási ráhagyás nagysága és iránya az Z tengely mentén. **Előjeles** szám.

Az, hogy a G73 mondatban megadott U és W cím jelentése Δi és Δk , vagy Δu és Δw , attól függ, hogy az adott mondatban programoztunk-e P-t és Q-t. Vagyis, ha programoztunk P-t és Q-t, akkor U és W jelentése Δu és Δw , ha nem programoztunk, akkor Δi és Δk .

A ciklus a P-t és Q-t tartalmazó mondatban játszódik le. Az n_s -től az n_f -ig tartó mondatoknak tartalmazniuk kell a rápozicionálást (A–A' szakasz) illetve a kontúr leírását (A'–B szakasz). A ciklus mind a négy síknegyedben végrehajtható a Δi , Δk , Δu , Δw értékek előjelének függvényében. A ciklus végén a szerszám az "A" pontra tér vissza. A megmunkálás a következő mondattól folytatódik.

Az n_s és n_f mondatok közötti programrészben programozott F, S, T funkciókat figyelmen kívül hagyja és azokat érvényesíti, amelyeket a G73 mondatban (f, s, t), vagy előbb programoztunk. Ugyanez vonatkozik az n_s és n_f mondatok között programozott konstans vágósebességre is, vagyis a G73 mondat előtti G96, vagy G97 állapotot, és konstans vágósebesség értéket érvényesíti.

Az n_s -től n_f -ig terjedő mondatok nem tartalmazhatnak alprogramhívást.

A szerszám sugár korrekció számítás megadása lehetséges a ciklust leíró mondatokban a G71 funkció tárgyalásakor említett megszorításokkal.

2. Megadási módszer:

G73 P (n_s) **Q** (n_f) **U**(Δu) **W**(Δw) **I**(Δi) **K**(Δk) **D**(d) **F**(f) **S**(s) **T**(t)
 $N(n_s)$...
 ...
 F____
 S____
 T____
 $N(n_f)$...

A második megadási módszer bemenő paraméterei megegyeznek az elsőével.

17.2.4 Símitó ciklus (G70)

A G71, G72, vagy a G73-mal történő nagyolás után a G70 utasítással símitást adhatunk meg. A símitás megadása a következő utasítással történik:

G70 P (n_s) Q (n_f) U(Δu) W(Δw)

n_s: a program símitást leíró részének kezdő mondatszám.

n_f: a program símitást leíró részének befejező mondatszám.

Δu: a símitási ráhagyás nagysága és iránya az X tengely mentén. Az X koordináta értelmezésének függvényében **átmérőben**, vagy **sugárban** értendő **előjeles** szám.

Δw: a símitási ráhagyás nagysága és iránya az Z tengely mentén. **Előjeles** szám.

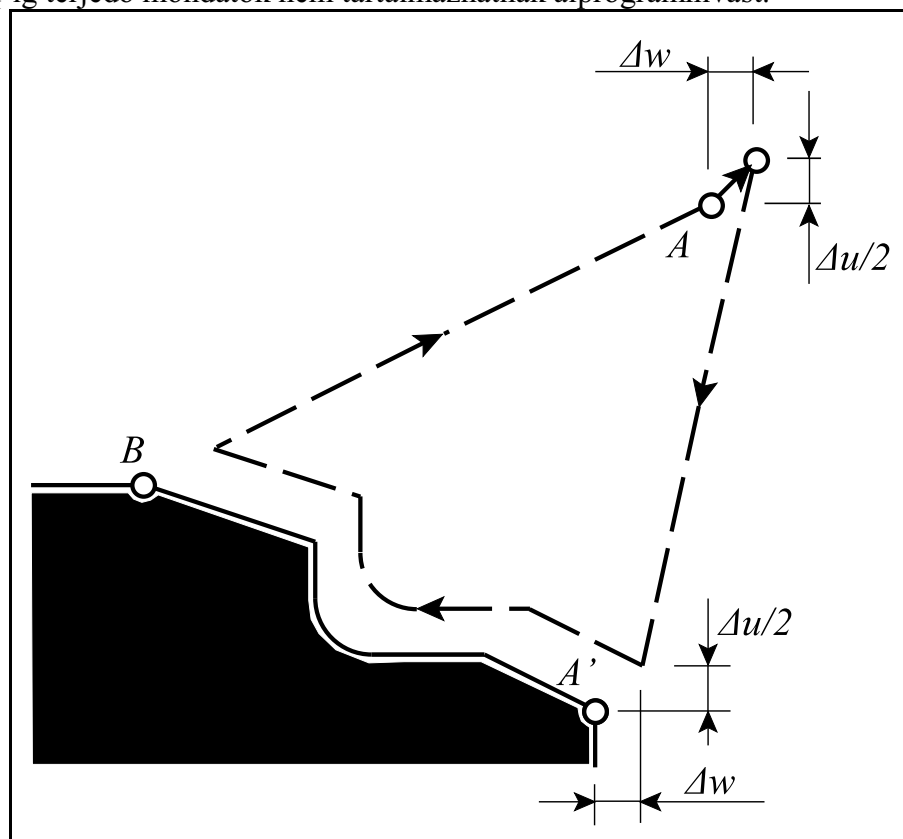
A ciklus folyamán a program símitást leíró n_s-től n_f-ig tartó részében programozott **F**, **S**, **T** funkciókat **végrehajtja**, ellentétben a G71, G72, G73 ciklussal.

A símitó ciklus végén a szerszám visszatér a kiindulási pontra és a következő mondat kerül beolvasásra.

A símitó ciklus folyamán a szerszám **sugár korrekció** számítása **működik**.

U és W címen símitási ráhagyás adható meg abban az esetben, ha a símitási ráhagyást több lépésben akarjuk eltávolítani.

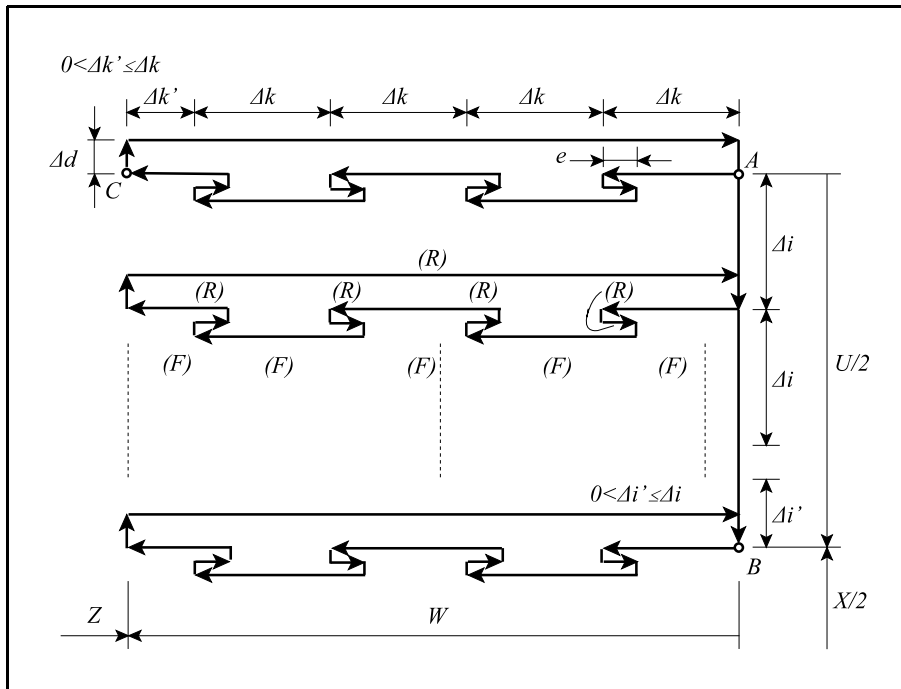
Az n_s-től n_f-ig terjedő mondatok nem tartalmazhatnak alprogramhívást.



17.2.4-1 ábra

17.2.5 Homlok beszúró ciklus (G74)

A mellékelt ábra egy G74 típusú homlok beszúró ciklus mozgását mutatja. A beszúrás Z irányban történik.



17.2.5 -1 ábra

1. Megadási módszer:

G74 R (e)

G74 X(U) Z(W) P (Δi) Q (Δk) R (Δd) F

utasítással történik, ahol:

e: a visszahúzás mértéke.

Öröklődő érték, addig nem változik, amíg át nem írjuk. A visszahúzás mértékét az 1344 RETG74G75 paraméteren is meg lehet adni, illetve ez a paraméter kerül átírásra a program utasítás hatására.

X: a "B" pont X irányú abszolút mérete

U: az AB pont távolsága inkrementálisan

Z: a "C" pont Z irányú abszolút mérete

W: az AC pont távolsága inkrementálisan

Δi: a fogásvétel nagysága X irányban. Mindig **sugárban** értendő, pozitív szám.

Δk: a fogásvétel nagysága Z irányban. Mindig pozitív szám.

Δd: a szerszám elhúzásának mértéke alul. Δd előjele mindig pozitív (a mozgás iránya mindig az AB vektor előjelével ellentétes).

Ha azonban az X(U) cím kitöltését és P(Δi) cím kitöltését is elhagyjuk, az R(Δd) előjele értelmezésre kerül, és a mozgás irányát alul Δd előjele határozza meg.

F: előtolás

Az ábrán (F)-fel jelöltük azokat a szakaszokat, amelyeket előtolással és (R)-rel azokat, amelyeket gyorsmenettel tesz meg.

Azt, hogy a G74-es mondatban az R cím kitöltése e-t, vagy Δd -t határoz meg, a Z(W) cím kitöltése dönti el. Ha Z(W) cím ki van töltve R cím jelentése Δd .

Ha az X(U) cím kitöltését és P(Δi) cím kitöltését is elhagyjuk, akkor mozgás csak a Z tengely mentén történik, vagyis egy fűróciklus játszódik le.

2. Megadási módszer:

G74 X(U) Z(W) I (Δi) K (Δk) D (Δd) F

A második megadási módszer bemenő paramétereinek értelmezése megegyezik az elsőével.

17.2.6 Beszúró ciklus (G75)

A mellékelt ábra egy G75 típusú beszúró ciklus mozgását mutatja.

1. Megadási módszer:

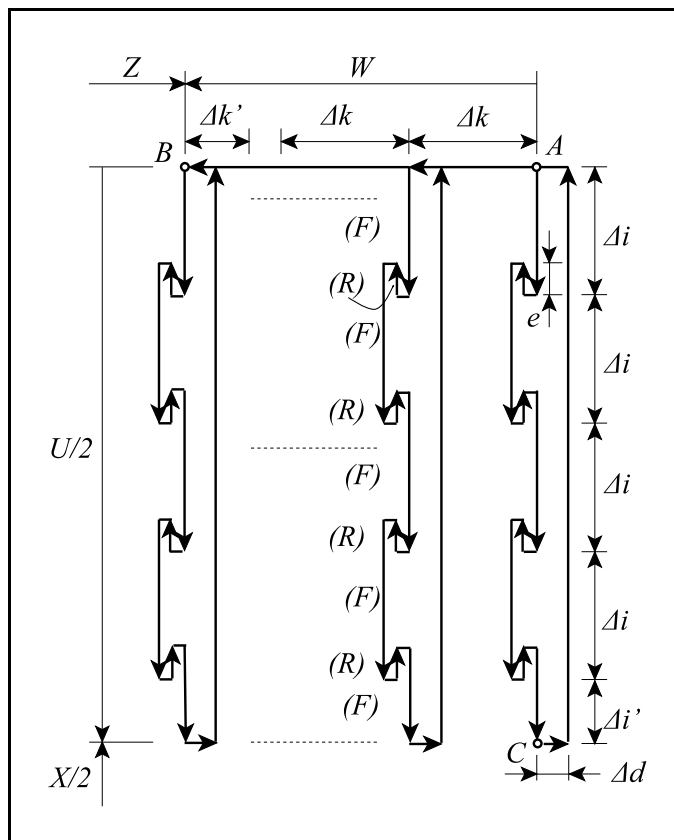
G75 R (e)

G75 X(U) Z(W) P (Δi) Q (Δk) R (Δd) F

A ciklus változóinak értelmezése megegyezik a G74 cikluséval, azzal a különbséggel, hogy a beszúrás X irányban történik, ezért X(U) és Z(W) címek értelmezése felcserélődik.

2. Megadási módszer:

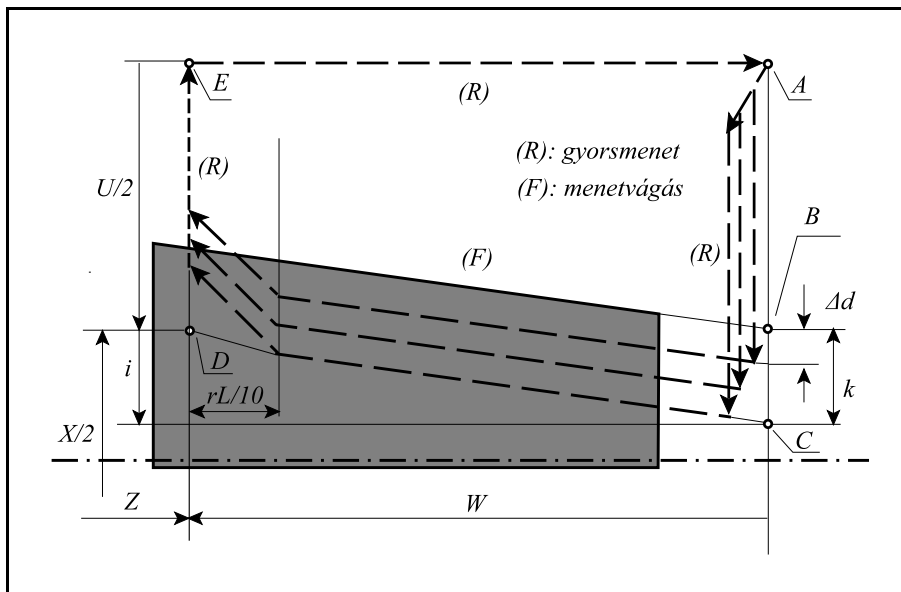
G75 X(U) Z(W) I (Δi) K (Δk) D (Δd) F



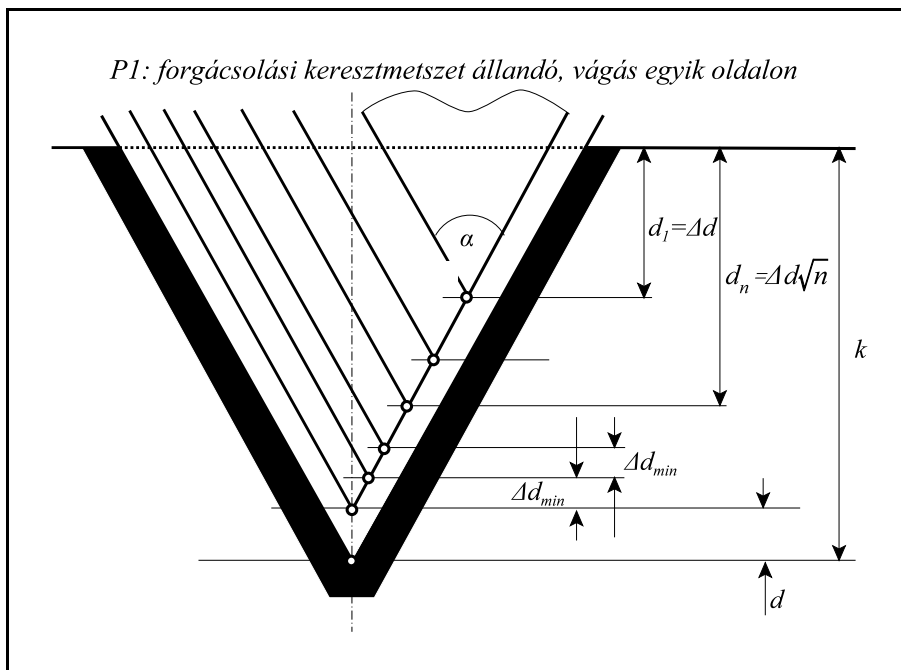
17.2.6 -1 ábra

A menetvágó ciklus (G76)

A mellékelt ábra egy G76 típusú menetvágó ciklus mozgását mutatja.



17.2.6 -1 ábra



17.2.6 -2 ábra

1. Megadási módszer:

G76 P (n) (r) (α) Q (Δd_{min}) R (d)

G76 X(U) Z(W) P (k) Q (Δd) R (i) F(E)(L)

utasítással történik, ahol:

n: a símitó ciklusok száma ($n=01\dots99$)

Ez az érték öröklődő, és addig nem változik, amíg át nem írjuk. A símitó ciklusok számát az 1335 COUNTFIN paraméteren is meg lehet adni, illetve ez a paraméter kerül átírásra a program utasítás hatására.

r: a letörés mértéke ($r=01\dots99$)

A menetből való kifutáskor a vezérlő kb. 45° -ban emeli ki a szerszámot. A kiemelési letörés hosszát adjuk meg r segítségével. A szakasz hossza

$$r \cdot L / 10$$

ahol: L : a programozott menetemelkedés

Ez az érték öröklődő, és addig nem változik, amíg át nem írjuk. A letörési szakasz hosszát az 1334 THRDCHMFR paraméteren is meg lehet adni, illetve ez a paraméter kerül átírásra a program utasítás hatására.

 α : a menetvágó kés élszöge fokban ($\alpha=01\dots99$)

Ez az érték öröklődő, és addig nem változik, amíg át nem írjuk. Az élszög értékét az 1336 TIPANGL paraméteren is meg lehet adni, illetve ez a paraméter kerül átírásra a program utasítás hatására.

n , r és α értékét a P címen adhatjuk meg egyszerre. Mivel mindegyik értéket egy kétjegyű szám fejez ki, P címre egy hatjegyű számot kell írni. Például, ha a símitó ciklusok száma $n=2$, a letörés értéke $1.5L$ ($r=15$), és 60° -os késsel dolgozunk, akkor P értéke: **P021560**.

 Δd_{\min} : a minimális fogás értéke (mindig **sugárban** értendő, pozitív szám)

Ha a menetvágás során az n -edik ciklusban a fogásvétel értéke $d_n - d_{n-1} < \Delta d_{\min}$, akkor a fogásvétel értékét mindig Δd_{\min} értékre korlátozza be. Ez az érték öröklődő, és addig nem változik, amíg át nem írjuk. A minimális fogás értékét az 1337 MINTHRDP paraméteren is meg lehet adni, illetve ez a paraméter kerül átírásra a program utasítás hatására.

d: símitási ráhagyás (mindig **sugárban** értendő, pozitív szám)

Ez az érték öröklődő, és addig nem változik, amíg át nem írjuk. A símitási ráhagyás értékét az 1338 FINALLW paraméteren is meg lehet adni, illetve ez a paraméter kerül átírásra a program utasítás hatására.

Az itt felsorolt paraméterek az első **G76 P** (n) (r) (α) **Q** (Δd_{\min}) **R** (d) mondat bemenő adatai. A vezérlő a G76 kódra akkor végzi el a fentebb felsorolt paraméterek átvételét, ha a G76 mondatban nincs sem $X(U)$, sem $Z(W)$ cím kitöltve.

i: a kúposág mértéke (mindig **sugárban** értendő)

Ha $i=0$, vagy az R címet nem töltjük ki, hengeres menetet vág.

k: a menet mélysége (mindig **sugárban** értendő, pozitív szám) **Δd : az első fogás mélysége** (mindig **sugárban** értendő, pozitív szám)**L: a menet emelkedése**

Programozása megegyezik a G33-nál elmondottakkal. Az F címre írt érték menetemelkedést, az E címre írt érték inchenkénti menetszámot jelöl.

A fenti paraméterek a második **G76 X(U) Z(W) R(i) P(k) Q(Δd) F(E)(L)** mondat bemenő adatai. Menetvágást csak a fenti módon kitöltött mondatra végez, ami azt jelenti, hogy az $X(U)$, $Z(W)$ címek valamelyikének kitöltve kell lenni. Ha egyik koordinátacím sincs kitöltve, akkor a mondatot paraméterbeállító mondatnak értelmezi.

A ciklus végrehajtása során a **fogásvételt**, vagyis az A és a C pontok közötti mozgást, **gyorsmenettel** hajtja végre, ha a ciklus alatt **G00** kód van érvényben, illetve a megörökölt **előtolással**, ha a ciklus alatt **G01** kód van érvényben.

A C és D pontok között előtoló mozgást végez, az F(E) címen megadott L menetemelkedésnek megfelelően. A többi szakaszt gyorsmenettel hajtja végre.

A menetet mindig az ábra szerinti módszerrel, az egyik oldalon vágja a $d_n = \Delta d \sqrt{n}$ összefüggés szerint úgy, hogy a forgácsolási keresztmetszet állandó legyen.

A menet végén mindig letörést vág a beállított paramétereknek megfelelően.

X(U), Z(W) címek előjele határozza meg a fogásvétel, illetve a menetvágás irányát,
R(i) cím előjele a kúp lejtési irányát.

STOP hatására a G78 kódnál leírtak szerint a programozott letörésnek megfelelően kiemeli a szerszámot, majd a kezdőpontra (A) áll. START gomb hatására a megszakított fogásvételt előlről kezdi.

2. megadási módszer:

G76 X(U) Z(W) I(i) K(k) D(Δd) A(α) F(E)(L) Q P

Az i, k, Δd, L adatok értelmezése megegyezik az 1. módszernél leírtakkal.

“A” címen (α) az 1. módszerhez hasonlóan a menetvágó kés élszögét lehet megadni. A különbség az, hogy amíg az 1. módszernél csak 1°-os felbontással tudjuk megadni, addig a 2. módszernél 0.001°-ossal. Ha “A” cím nincs kitöltve α értékét paraméterről veszi. Viszont az “A” cím megadása nem változtatja meg az 1336 TIPANGL paraméter értékét.

Az n, r, α, Δd_{min} és d bemenő paramétereket a vezérlő éppúgy figyelembe veszi a 2. módszerrel történő menetmegadásnál, mint az 1-nél. A mondatot megelőző **G76 P** (n) (r) (α) **Q** (Δd_{min}) **R** (d) mondattal ebben az esetben is megadhatunk bemenő paramétereket.

A vezérlő akkor tekinti a mondatmegadást a 2. módszer szerintinek, ha a mondatban K cím ki van töltve.

Q: a menet kezdetének a jeladó nullimpulzusától számított szögértéke °-ban megadva. A cím értelmezése megegyezik a G33-nál elmondottakkal.

P: a menetvágás módszere.

Ötféle menetvágási módszer között lehet választani az alábbi ábrák alapján

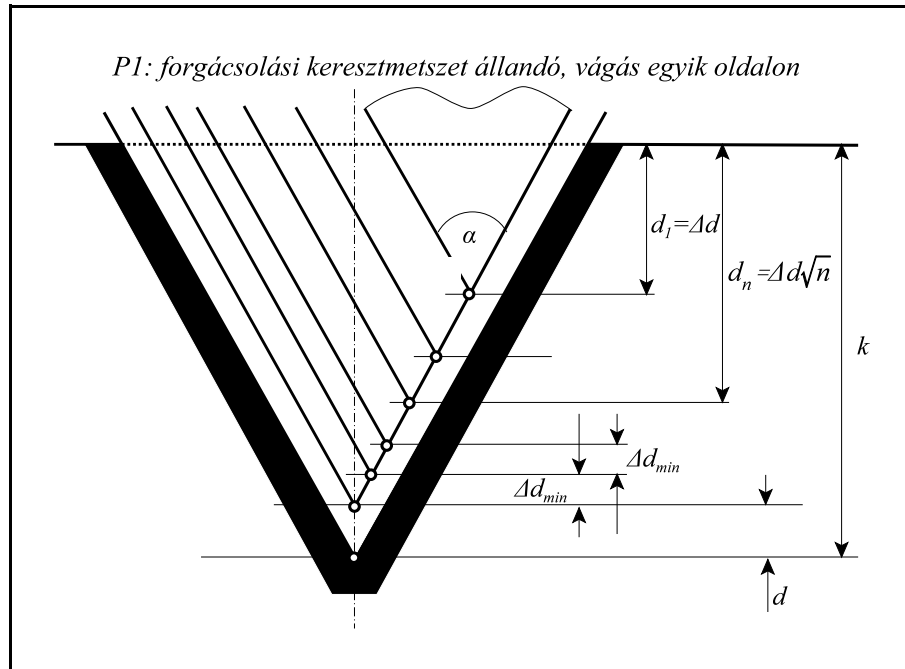
P1: forgácsolási keresztmetszet állandó, vágás egyik oldalon

P2: vágás mindkét oldalon

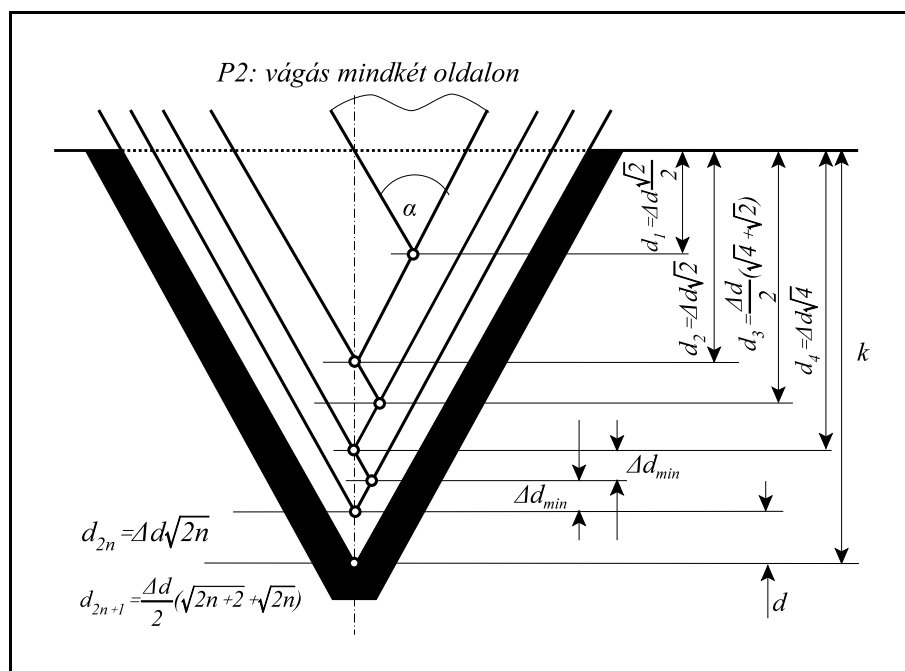
P3: fogásvétel állandó vágás egyik oldalon

P4: fogásvétel állandó vágás mindkét oldalon

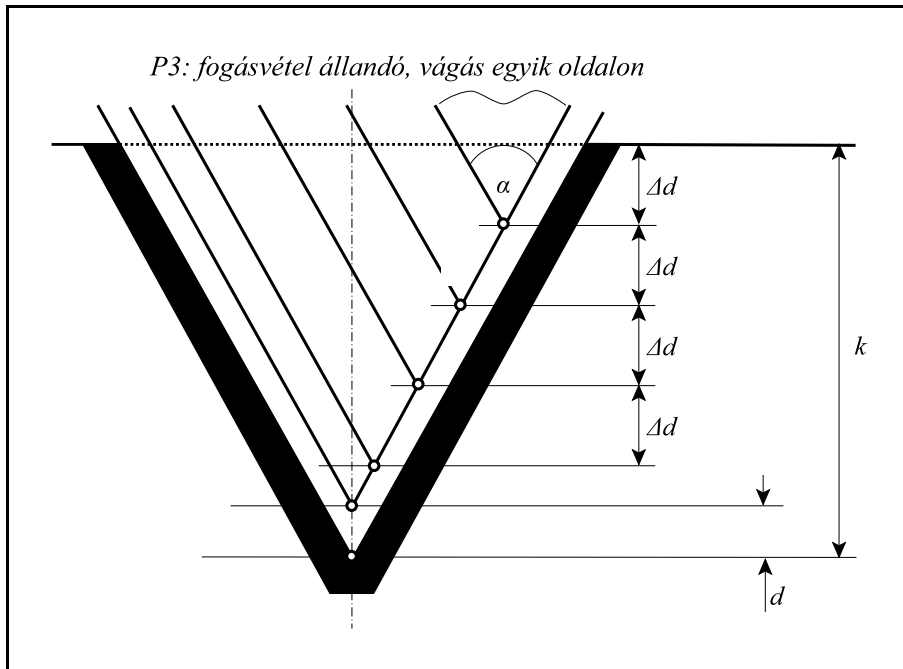
P5: forgácsolási keresztmetszet állandó, vágás mindkét oldalon



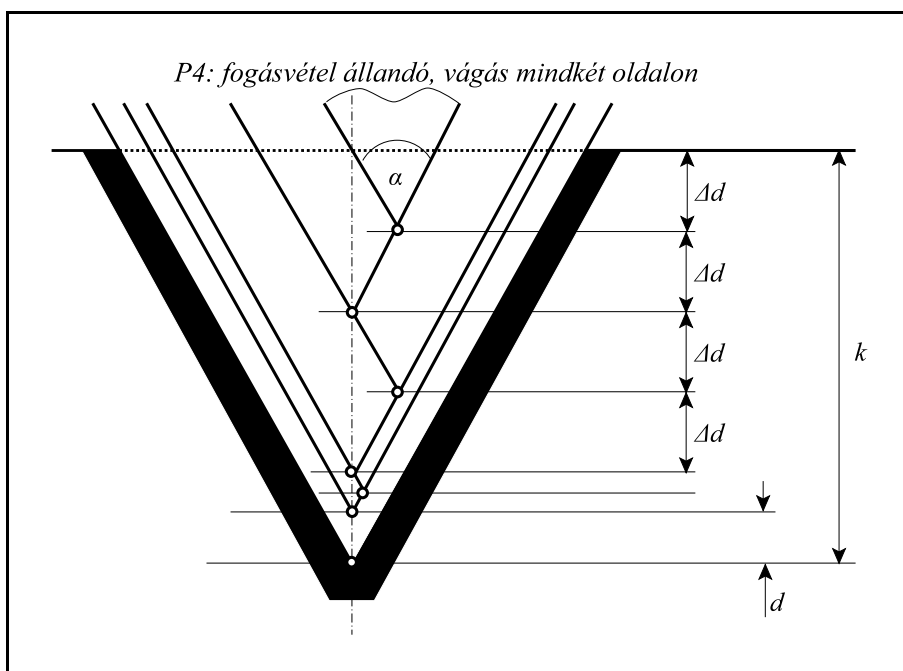
17.2.6 -3 ábra



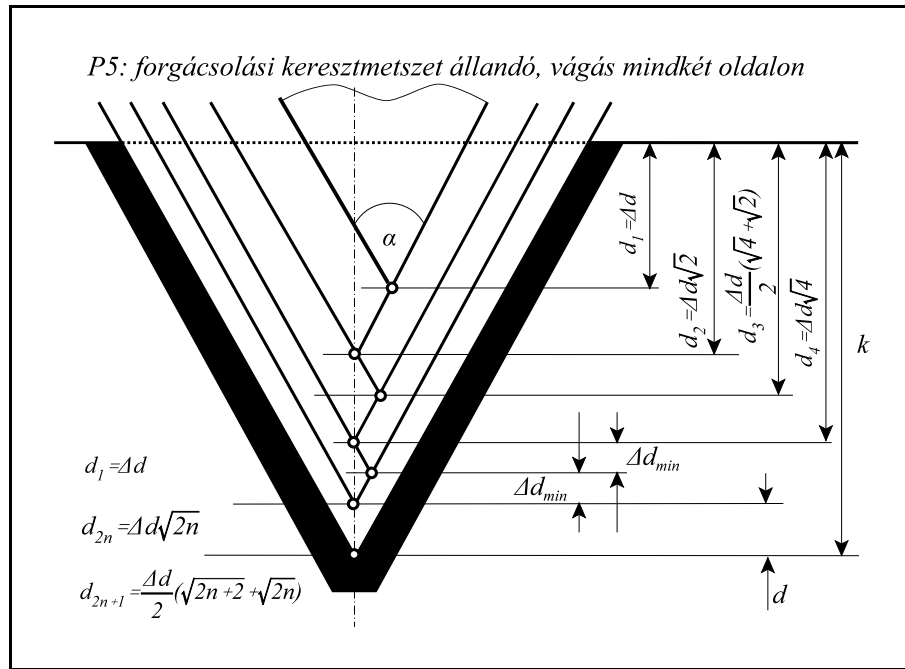
17.2.6 -4 ábra



17.2.6 -5 ábra



17.2.6 -6 ábra



17.2.6 -7 ábra

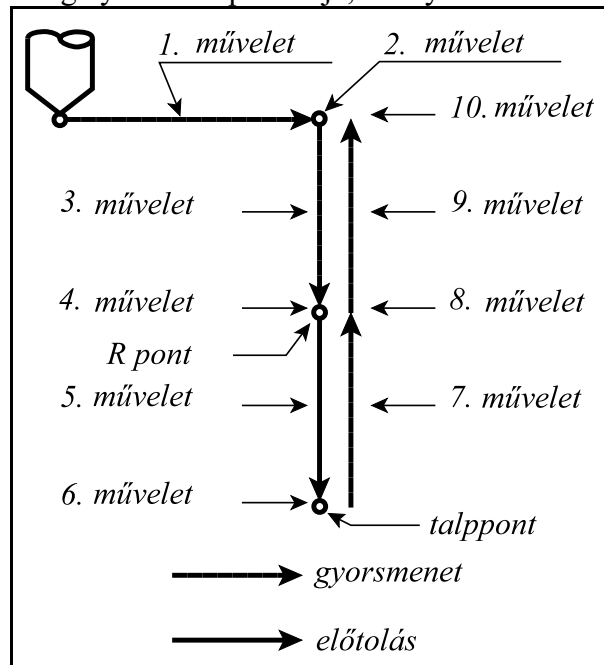
18 Fúróciklusok

A fúróciklusokat a következő műveletekre lehet bontani:

1. művelet: pozicionálás a kiválasztott síkban
2. művelet: tevékenység pozicionálás után
3. művelet: gyorsmeneti mozgás az R pontig (megközelítési pont)
4. művelet: tevékenység az R ponton
5. művelet: fúrás a talppontig
6. művelet: tevékenység a talpponton
7. művelet: visszahúzás az R pontig
8. művelet: tevékenység az R ponton
9. művelet: gyorsmeneti visszahúzás a kiindulási pontig
10. művelet: tevékenység a kiindulási ponton

R pont, megközelítési pont: A szerszám a munkadarabot eddig a pontig közelíti meg gyorsmeneti mozgással.

Kiindulási pont: A fúrótengelynek az a pozíciója, amelyet a ciklus indítása előtt felvesz.



18-1 ábra

A fenti műveletek a fúróciklusok általános leírását adják, a konkrét esetekben műveletek elmaradhatnak.

A fúróciklusoknak van **pozicionálási síkja**, és **fúrótengelye**. A fúrótengelyt a G17, G18, G19 síkválasztó utasítások jelölik ki. Az összes többi tengelyt a pozicionálási síkban mozgatja.

G kód	fúrótengely
G17	Z _p
G18	Y _p
G19	X _p

ahol: X_p: X, vagy azzal párhuzamos tengely
 Y_p: Y, vagy azzal párhuzamos tengely
 Z_p: Z, vagy azzal párhuzamos tengely

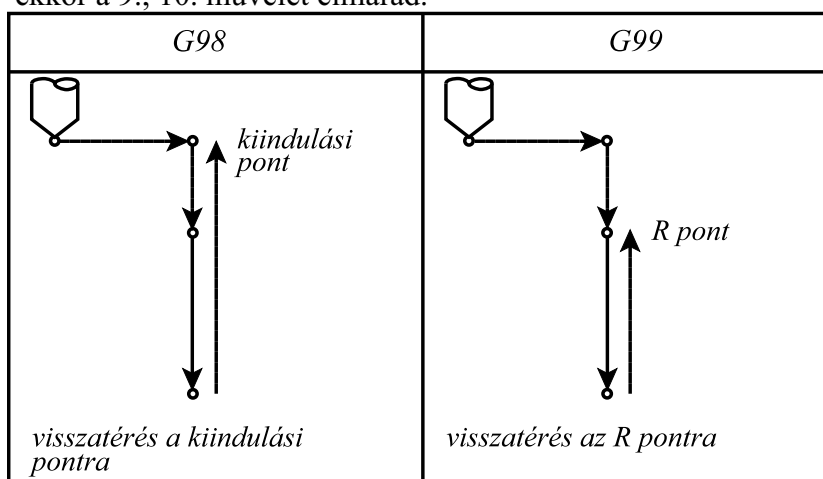
U, V, W tengelyeket akkor tekinti párhuzamos tengelyeknek, ha a paramétermezőben annak vannak definiálva.

Ha homlokfúrást akarunk programozni, ahol a fúrótengely a Z, válasszuk a G17-es síkot, ha oldalfúrást, ahol a fúrótengely az X, válasszuk a G19-es síkot.

A fúróciklusok konfigurálása a G98 és G99 utasításokkal lehetséges:

G98: a szerszám a fúróciklus során a kiindulási pontig kerül visszahúzásra. Alaphelyzet, amelyet a vezérlő bekapcsolás, üzemmódváltás, vagy a ciklus üzemmód törlése után vesz fel.

G99: a szerszám a fúróciklus során az R pontig kerül visszahúzásra, következésképp ekkor a 9., 10. művelet elmarad.



18-2 ábra

A fúróciklusok kódjai: G83.1, G84.1, G86.1, G81, ..., G89

Ezek a kódok bekapcsolják a ciklus üzemmódot, amely lehetővé teszi a ciklusváltozók öröklődését.

A G80 kód kikapcsolja a ciklus üzemmódot és törli az eltárolt ciklusváltozókat.

A fúróciklusokban használt címek és értelmezésük:

G17	G_	X_p_	Y_p_	C_	I_	J_	Z_p_	R_	Q_	E_	P_	F_	S_	L_	
G18	G_	Z_p_	X_p_	C_	K_	I_	Y_p_	R_	Q_	E_	P_	F_	S_	L_	
G19	G_	Y_p_	Z_p_	C_	J_	K_	X_p_	R_	Q_	E_	P_	F_	S_	L_	

ismétlési szám
fúrési adatok
elmozdulás főorsó orientálás után
a furat pozíciója
a fúrás kódja

A fúrás kódja:

Az egyes kódok értelmezését lásd később.

A kódok öröklődnek mindaddig amíg G80 utasítást, vagy az 1-es G kód csoportba (interpolációs csoport: G01, G02, G03, G33) tartozó kódot nem programozunk, illetve törlődik üzemmódváltásra.

Amíg a ciklusállapot be van kapcsolva, a G83.1, G84.1, G86.1, G81, ..., G89 utasításokkal, addig az öröklődő ciklusváltozók a különböző típusú fúróciklusok között is átöröklődnek.

A kezdőpont vagy kiindulási pont:

A kezdőpont a fúrásra kijelölt tengely pozíciója, amely bejegyzésre kerül:

– ciklusüzemmód bekapcsolásakor. Például:

```
N1 G17 G90 G0 Z200
N2 G81 X0 C0 Z50 R150
N3 X100 C30 Z80
```

esetén a kezdőpont pozíciója Z=200 az N2 és N3 mondatban is.

– vagy új fúrótengely kijelölésénél. Például:

```
N1 G17 G90 G0 Z200 W50
N2 G81 X0 C0 Z50 R150
N3 X100 C30 W20 R25
```

N2 mondatban a kezdőpont pozíciója Z=200

N3 mondatban a kezdőpont pozíciója W=50

Abban az esetben, ha változik a fúrótengely kijelölése R programozása kötelező, ellenkező esetben *3053 NINCS TALPPONT VAGY R PONT* üzenet képződik.

A furat pozíciója: X_p, Y_p, Z_p, C

A beírt koordinátaértékek közül a fúrótengelyt leszámítva a többi koordinátaadatot veszi a furat pozíciójának.

A beírt értékek lehetnek inkrementális, vagy abszolút, dimenziójuk pedig metrikus vagy inches.

A beírt koordinátaértékekre érvényesek a tükrözési, és léptékezési parancsok.

A furat pozíciójára a vezérlés gyorsmeneti pozicionálással áll rá, függetlenül attól, hogy melyik kód volt érvényben az 1-es csoportból.

Elmozdulás főorsó orientálás után: I, J, K

Ha az adott gépen lehetőség van a főorsó orientálására, a G76, és G87 kiesztergáló ciklusokban a szerszámot a felülettől eltávolítva lehet visszahúzni, hogy a szerszám hegye ne karcolja azt. Ekkor I, J és K címen lehet megadni hogy a szerszámot milyen irányban távolítsa el a felülettől a vezérlés. A kiválasztott síknak megfelelően értelmezi a vezérlés a címeket:

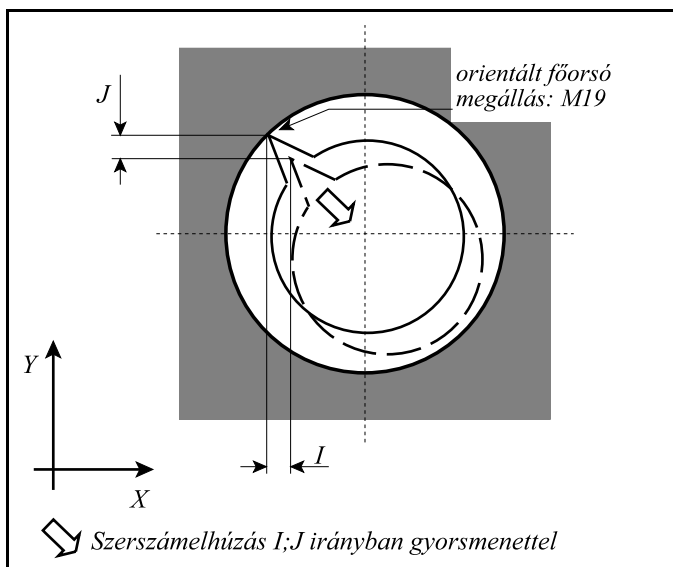
G17: I, J

G18: K, I

G19: J, K

A címek mindig inkrementális, derékszögű adatként kerülnek értelmezésre. A cím lehet metrikus és inch-es.

I, J, K adatokra nem érvényesek a tükrözési, vagy léptékezési parancsok. I, J és K öröklődő értékek. G80, vagy az interpolációs csoport kódjai értékeit törlik. Az elhúzás gyorsmenettel történik.

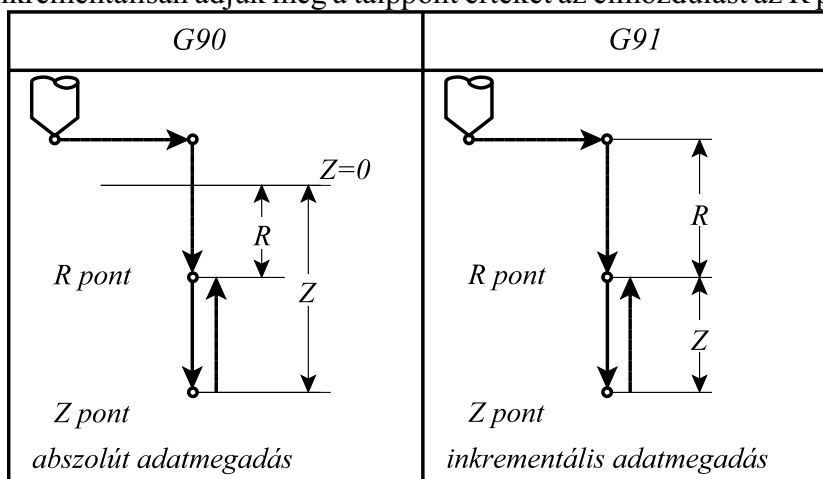


18-3 ábra

Fúrési adatok:

A furat talppontja: X_{p2}, Y_{p2}, Z_p

A furat talppontját a fúrotengely címén kell megadni. A furat talppontjának koordinátája mindig derékszögű adatként kerül értelmezésre. Lehet inch-es, vagy metrikus, abszolút, vagy inkrementális. Ha inkrementálisan adjuk meg a talppont értékét az elmozdulást az R ponttól számítja.



18-4 ábra

A talppont adataira érvényesek a tükrözési és léptékezési parancsok. A talppont adata öröklődő érték. G80, vagy az interpolációs csoport kódjai értékét törlik. A talppontot mindig az érvényes előtolással közelíti meg a vezérlő.

A megközelítési pont, R pont: R

A megközelítési pontot R címen adjuk meg. Az R cím mindig derékszögű adat amely lehet inkrementális és abszolút, metrikus, vagy inch-es. Ha R adat inkrementális, értékét a kiindulási ponttól számítjuk. Az R pont adataira érvényesek a tükrözési és léptékezési adatok. Az R pont adata öröklődik. G80, vagy az interpolációs csoport kódjai értékét törlik. Az R pontot mindig gyorsmeneti mozgással közelíti meg a vezérlő.

A fogásmélység értéke: Q

G83.1-es és G83-as ciklusokban a fogásmélység értéke. Mindig inkrementális, derékszögű, pozitív adat. A fogásmélység értéke öröklődő adat. G80, vagy az interpolációs csoport kódjai értékét törlik. A fogásmélységre nem érvényes a léptékezési parancs.

Segédadat: E

G83.1-es ciklusban a visszahúzás mértéke, illetve a G83-asban pedig a fogásvétel előtt ekkora értékig megy gyorsmenettel. Mindig inkrementális, derékszögű, pozitív adat. A segédadatra nem érvényes a léptékezési parancs. A segédadat értéke öröklődő. G80, vagy az interpolációs csoport kódjai értékét törlik. Ha nem programozták, akkor a *RETG73*, illetve *CLEG83* paraméterről veszi a vezérlés a szükséges értéket.

Várakozás: P

A várakozási időt adja meg a furat alján. Megadására a G04-nél elmondott szabályok érvényesek. A várakozás értéke öröklődő. G80, vagy az interpolációs csoport kódjai értékét törlik.

Előtolás: F

Az előtolást határozza meg. Értéke öröklődő. Csak egy másik F adat programozása írja felül, G80 vagy más kód nem törli.

Főorsó fordulatszám: S

Értéke öröklődő. Csak egy másik S adat programozása írja felül, G80 vagy más kód nem törli.

Ismétlési szám: L

A ciklus ismétlésének számát határozza meg. Értékhatára: 1–9999. Ha L nincs kitöltve L=1 értéket vesz figyelembe. L=0 esetén a ciklus adatai eltárolódnak, de nem hajtódnak végre. L értéke csak abban a mondatban érvényes, ahol megadtuk.

Példa a fúrási kódok és a ciklusváltozók öröklődésére:

```
N1 G17 G0 Z_ M3
N2 G81 X_ C_ Z_ R_ F_
```

A ciklus üzemmód kezdetén a fúrási adatok (Z, R) meghatározása kötelező)

```
N3 X_
```

Mivel az N2 mondatban a fúrási adatok meg lettek határozva és az N3 mondatban ugyanazok szükségesek, kitöltésük felesleges, azaz G81, Z_, R_, F_ elhagyható. A furat pozíciója csak X irányban változik, a fúró ebben az irányban mozog, majd ugyanazt a furatot fúrja, mint az N2 mondatban.

```
N4 G82 C_ Z_ P_
```

A furat pozíciója C irányban mozog. A fúrás módszere G82-nek megfelelően alakul, a talppont Z új értéket vesz fel, a megközelítési pont és előtolás (R, F) N2 mondatból öröklődnek.

```
N5 G80 M5
```

Törli a ciklusüzemmódot és az öröklődő ciklusváltozókat, F kivételével.

```
N6 G85 C_ Z_ R_ P_ M3
```

Mivel az N5 mondatban törölődtek a fúrási adatok G80 utasítás hatására Z, R, és P értékeket újra meg kell adni.

```
N7 G0 X_ C_
```

Törli a ciklusüzemmódot és az öröklődő ciklusváltozókat, F kivételével.

Példák a ciklus ismétlésének használatára:

Ha ugyanolyan furatokat kell egyenlő távolságra készíteni ugyanolyan paraméterekkel az ismétlési számot az L címen adhatjuk meg. L csak abban a mondatban érvényes, amelyben megadtuk.

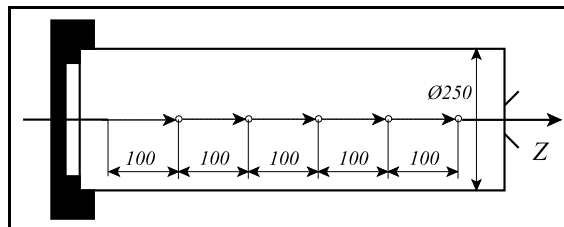
```
N1 G90 G19 G0 X300 Z40 C0 M3
N2 G91 G81 X-40 Z100 R-20 F50 L5
```

A fenti utasítások hatására a vezérlő az Z tengely mentén 100 mm távolságra egymástól 5 db egyforma furatot fúr. Az első furat pozíciója Z=140, C=0. Mivel oldalfúrásról van szó (X tengellyel fúrunk) a G19 síkot választottuk.

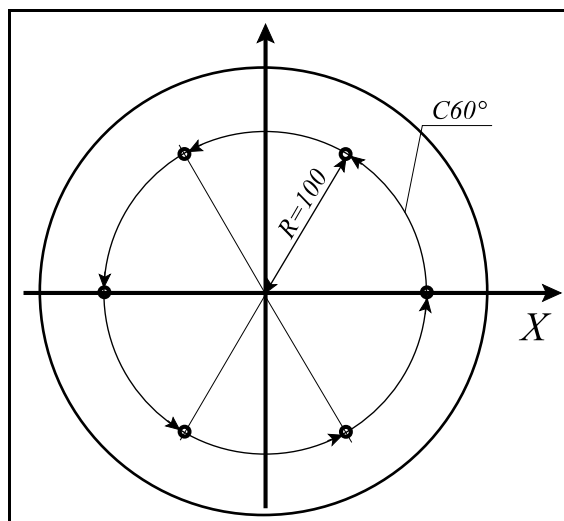
A furat pozíciója G91 hatására inkrementálisan lett megadva.

```
N1 G90 G17 G0 X200 C-60 Z50
N2 G81 CI60 Z-40 R3 F50 L6
```

A fenti utasítások hatására a vezérlő egy 100 mm-es lyukkörön 60 fokként 6 db furatot fúr. Az első furat pozíciója az X=200 C=0 koordinátájú pontra esik. Mivel homlokfúrásról van szó (Z tengellyel fúrunk) a G17 síkot választottuk.



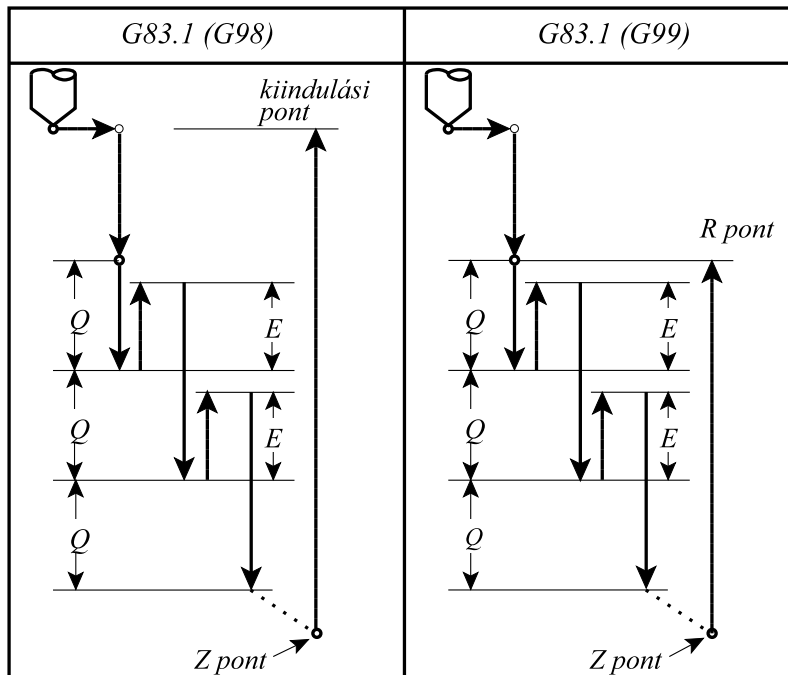
18-5 ábra



18-6 ábra

18.1 A fúróciklusok részletes leírása.

18.1.1 Nagysebességű mélyfúróciklus (G83.1)



18.1.1-1 ábra

A ciklusban felhasznált változók:

G17 **G83.1** X_p Y_p C_— Z_p R_— Q_— E_— F_— L_—

G18 **G83.1** Z_p X_p C_— Y_p R_— Q_— E_— F_— L_—

G19 **G83.1** Y_p Z_p C_— X_p R_— Q_— E_— F_— L_—

A ciklus műveletei:

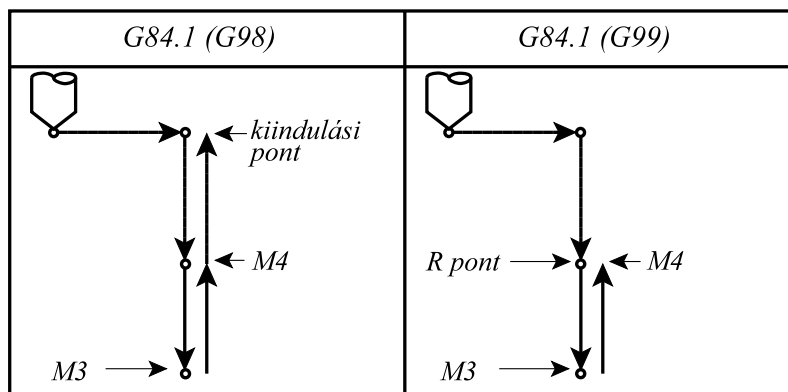
1. művelet: pozicionálás a kiválasztott síkban gyorsmenettel
2. művelet: –
3. művelet: gyorsmeneti mozgás az R pontig (megközelítési pont)
4. művelet: –
5. művelet: fúrás a talppontig F előtolással
6. művelet: –
7. művelet: G99 esetén: visszahúzás az R pontig gyorsmenettel
8. művelet: –
9. művelet: G98 esetén: visszahúzás a kiindulási pontig gyorsmenettel
10. művelet: –

Az 5. fúrási művelet leírása:

- a Q címen megadott fogásmélységet előtolással belefúrja az anyagba,
- az E címen, vagy a *RETG73* paraméteren megadott értékkel gyorsmenettel visszahúzó,
- az előző befúrás talppontjától számítva Q mélységet ismételtelen befúr,
- az E címen megadott értékkel gyorsmenettel visszahúzó.

Az eljárás a Z címen megadott talppontig folytatódik.

18.1.2 Balmenetfűrő ciklus (G84.1)



18.1.2-1 ábra

A ciklus csak kiegyenlítőbetéttel ellátott menetfűrővel alkalmazható.

A ciklusban felhasznált változók:

G17 **G84.1** X_p Y_p C Z_p R (P) F L

G18 **G84.1** Z_p X_p C Y_p R (P) F L

G19 **G84.1** Y_p Z_p C X_p R (P) F L

A ciklus indítása előtt M4 (óramutató járásával ellentétes) főorsó forgásirányt kell bekapcsolni illetve programozni.

Az előtolás értékét a fűrő menetemelkedésének függvényében kell megadni:

– G94 percenkénti előtolás állapotban:

$$F = P \cdot S$$

ahol: P: a menetemelkedés mm/ford, vagy inch/ford dimenzióban

S: a főorsó fordulat ford/perc dimenzióban

– G95 fordulatonkénti előtolás állapotban:

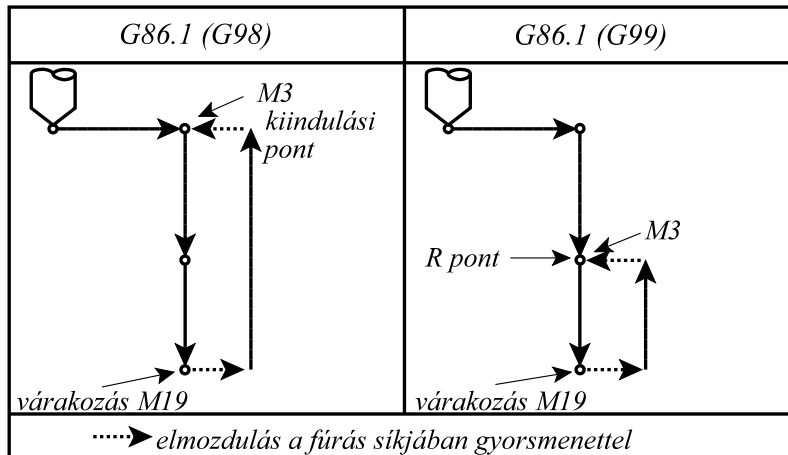
$$F = P$$

ahol: P: a menetemelkedés mm/ford, vagy inch/ford dimenzióban

A ciklus műveletei:

1. művelet: pozicionálás a kiválasztott síkban gyorsmenettel
2. művelet: –
3. művelet: gyorsmeneti mozgás az R pontig (megközelítési pont)
4. művelet: –
5. művelet: fűrás a talppontig F előtolással, override és stop tiltva
6. művelet: – várakozás P címen megadott értékkel, ha a *TAPDWELL* paraméter engedélyezve van (=1)
– főorsó forgásirányváltás: M3
7. művelet: visszahúzás az R pontig F előtolással, override és stop tiltva
8. művelet: főorsó forgásirányváltás: M4
9. művelet: G98 esetén: gyorsmeneti visszahúzás a kiindulási pontig
10. művelet: –

18.1.3 Kiesztergálás automatikus szerszámelhúzással (G86.1)



18.1.3-1 ábra

A G86.1 ciklust csak akkor lehet használni, ha a főorsó orientálás be van építve a szerszámgépbe. A vezérlő számára ezt a tényt az *ORIENT1* paraméterbit 1 állapota jelzi. Ellenkező esetben 3052 G76, G87 HIBA jelzést ad.

Mivel a ciklus a kiesztergálás után főorsó orientálást végez és a szerszámot elhúzza a felülettől I, J és K-n megadott értékkel, a szerszám kihúzásakor nem karcolódik a felület.

A ciklusban felhasznált változók:

G17 **G86.1** X_p__ Y_p__ C__ I__ J__ Z_p__ R__ P__ F__ L__
 G18 **G86.1** Z_p__ X_p__ C__ K__ I__ Y_p__ R__ P__ F__ L__
 G19 **G86.1** Y_p__ Z_p__ C__ J__ K__ X_p__ R__ P__ F__ L__

A ciklus indítása előtt M3 parancsot kell kiadni.

A ciklus műveletei:

1. művelet: pozicionálás a kiválasztott síkban gyorsmenettel
2. művelet: –
3. művelet: gyorsmeneti mozgás az R pontig (megközelítési pont)
4. művelet: –
5. művelet: kiesztergálás a talppontig F előtolással
6. művelet:
 - várakozás P címen megadott értékkel
 - főorsó orientálás: M19
 - szerszámelhúzás a kiválasztott síkban I, J, K értékkel gyorsmenettel
7. művelet: G99 esetén: visszahúzás az R pontig, gyorsmenettel
8. művelet: G99 esetén
 - szerszámvisszahúzás a kiválasztott síkban I, J, K-n megadott értékkel ellentétesen, gyorsmenettel,
 - főorsó újraindítása M3 irányban
9. művelet: G98 esetén: gyorsmeneti visszahúzás a kiindulási pontig
10. művelet: G98 esetén
 - szerszámvisszahúzás a kiválasztott síkban I, J, K-n megadott értékkel ellentétesen, gyorsmenettel,
 - főorsó újraindítása M3 irányban

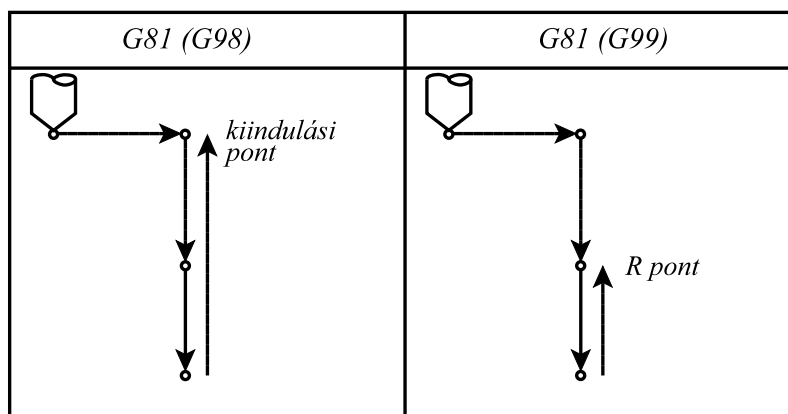
18.1.4 A ciklusállapot kikapcsolása (G80)

A kód hatására a ciklusállapot kikapcsolódik, a ciklusváltozók törlődnek.

Z és R inkrementális 0 értéket vesz fel, a többi változó 0-t.

Ha a **G80** mondatba koordinátákat programozunk, és más utasítást nem adunk, akkor a ciklus bekapcsolása előtt érvényes interpolációs kód (az 1-es G kód csoport, vagy interpolációs csoport) alapján hajtódik végre a mozgás.

18.1.5 Fúróciklus, kiemelés gyorsmenettel (G81)



18.1.5-1 ábra

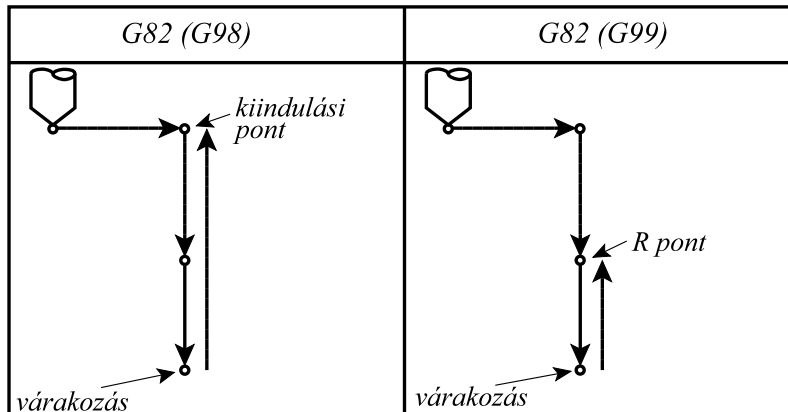
A ciklusban felhasznált változók:

G17 **G81** X_p__ Y_p__ C__ Z_p__ R__ F__ L__
 G18 **G81** Z_p__ X_p__ C__ Y_p__ R__ F__ L__
 G19 **G81** Y_p__ Z_p__ C__ X_p__ R__ F__ L__

A ciklus műveletei:

1. művelet: pozícionálás a kiválasztott síkban gyorsmenettel
2. művelet: –
3. művelet: gyorsmeneti mozgás az R pontig (megközelítési pont)
4. művelet: –
5. művelet: fúrás a talppontig F előtolással
6. művelet: –
7. művelet: G99 esetén: visszahúzás az R pontig, gyorsmenettel
8. művelet: –
9. művelet: G98 esetén: gyorsmeneti visszahúzás a kiindulási pontig
10. művelet: –

18.1.6 Fúróciklus várakozással, kiemelés gyorsmenettel (G82)



18.1.6-1 ábra

A ciklusban felhasznált változók:

G17 **G82** X_p__ Y_p__ C__ Z_p__ R__ P__ F__ L__

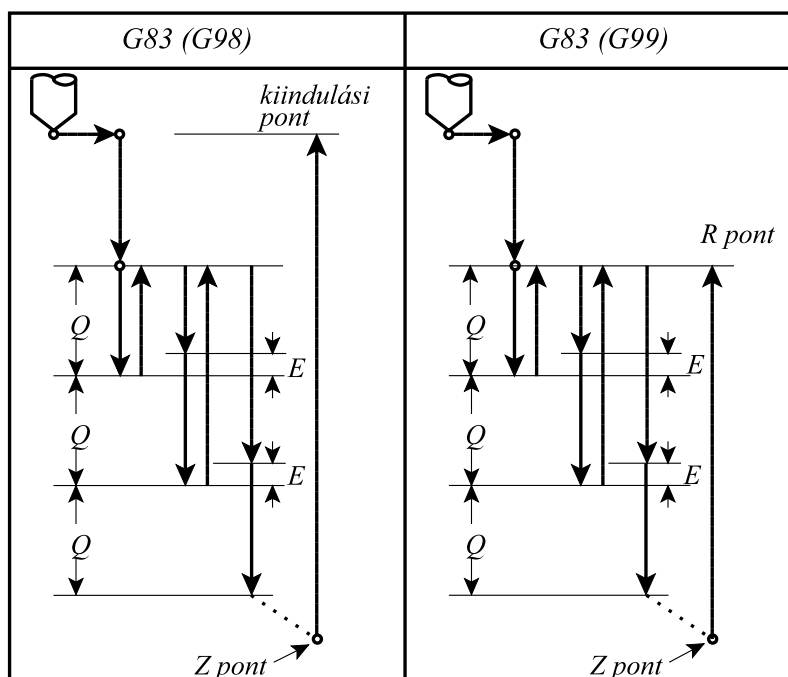
G18 **G82** Z_p__ X_p__ C__ Y_p__ R__ P__ F__ L__

G19 **G82** Y_p__ Z_p__ C__ X_p__ R__ P__ F__ L__

A ciklus műveletei:

1. művelet: pozicionálás a kiválasztott síkban gyorsmenettel
2. művelet: –
3. művelet: gyorsmeneti mozgás az R pontig (megközelítési pont)
4. művelet: –
5. művelet: fúrás a talppontig F előtolással
6. művelet: várakozás P címen megadott ideig
7. művelet: G99: esetén: visszahúzás az R pontig, gyorsmenettel
8. művelet: –
9. művelet: G98 esetén: gyorsmeneti visszahúzás a kiindulási pontig
10. művelet: –

18.1.7 Mélyfúróciklus (G83)



18.1.7-1 ábra

A ciklusban felhasznált változók:

G17 **G83** X_p__ Y_p__ C__ Z_p__ R__ Q__ E__ F__ L__

G18 **G83** Z_p__ X_p__ C__ Y_p__ R__ Q__ E__ F__ L__

G19 **G83** Y_p__ Z_p__ C__ X_p__ R__ Q__ E__ F__ L__

A ciklus műveletei:

1. művelet: pozicionálás a kiválasztott síkban gyorsmenettel
2. művelet: –
3. művelet: gyorsmeneti mozgás az R pontig (megközelítési pont)
4. művelet: –
5. művelet: fúrás a talppontig F előtolással
6. művelet: –
7. művelet: G99 esetén: visszahúzás az R pontig gyorsmenettel
8. művelet: –
9. művelet: G98 esetén: visszahúzás a kiindulási pontig gyorsmenettel
10. művelet: –

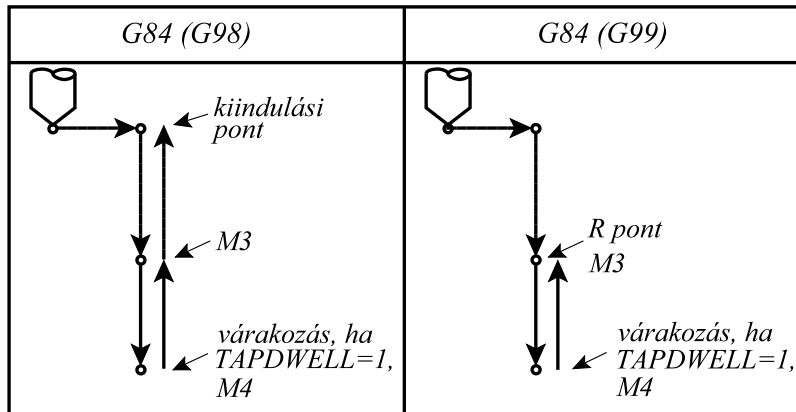
Az 5. fúrási művelet leírása:

- a Q címen megadott fogásmélységet előtolással belefúrja az anyagba,
- gyorsmenettel visszahúz az R pontig,
- gyorsmenettel megközelíti az előző mélységet E távolságig,
- az előző befúrás talppontjától számítva Q mélységet ismételtelen befúr, F előtolással (elmozdulás E+Q)
- gyorsmenettel visszahúz, az R pontig

Az eljárás a Z címen megadott talppontig folytatódik.

E távolságot vagy a programból E címről, vagy a *CLEG83* paraméterről veszi.

18.1.8 Menetfúró ciklus (G84)



18.1.8-1 ábra

A ciklus csak kiegyenlítőbetéttel ellátott menetfúróval alkalmazható.

A ciklusban felhasznált változók:

G17 **G84** X_p Y_p C Z_p R (P) F L

G18 **G84** Z_p X_p C Y_p R (P) F L

G19 **G84** Y_p Z_p C X_p R (P) F L

A ciklus indítása előtt M3 (óramutató járásával megegyező) főorsó forgásirányt kell bekapcsolni.

Az előtolás értékét a fúró menetemelkedésének függvényében kell megadni:

– G94 percenkénti előtolás állapotban:

$$F = P \cdot S$$

ahol: P: a menetemelkedés mm/ford, vagy inch/ford dimenzióban

S: a főorsó fordulat ford/perc dimenzióban

– G95 fordulatonkénti előtolás állapotban:

$$F = P$$

ahol: P: a menetemelkedés mm/ford, vagy inch/ford dimenzióban

A ciklus műveletei:

1. művelet: pozicionálás a kiválasztott síkban gyorsmenettel
2. művelet: –
3. művelet: gyorsmeneti mozgás az R pontig (megközelítési pont)
4. művelet: –
5. művelet: fúrás a talppontig F előtolással, override és stop tiltva
6. művelet:
 - várakozás P címen megadott értékkel, ha a TAPDWELL paraméter engedélyezve van (=1)
 - főorsó forgásirányváltás: M4
7. művelet: visszahúzás az R pontig F előtolással, override és stop tiltva
8. művelet: főorsó forgásirányváltás: M3
9. művelet: G98 esetén: gyorsmeneti visszahúzás a kiindulási pontig
10. művelet: –

18.1.9 Menetfűrő ciklus kiegyenlítő betét nélkül (G84.2, G84.3)

Menetfűrés esetén a fűrőtengely előtolása és a főorsó fordulat hányadosának egyenlőnek kell lennie a menetfűrő menetemelkedésével. Másképpen fogalmazva menetfűrésnél ideális esetben az alábbi hányadosnak pillanatról pillanatra állandónak kell lenni: $P = \frac{F}{S}$

ahol: P: a menetemelkedés (mm/ford, vagy inch/ford),
F: előtolás (mm/min, vagy inch/min),
S: főorsó fordulatszám (ford/min).

A G84.1-es balmenet, és a G84-es jobbmenet fűrőciklusban a főorsó fordulatszáma és a fűrőtengely előtolása egymástól teljesen függetlenül vezérlődik. A fenti feltétel ennek megfelelően nem teljesülhet pontosan. Különösen igaz ez a furat alján, ahol a fűrőtengely előtolásának és a főorsó fordulatszámának egymással szinkronban kellene lelassulnia és megállnia, majd az ellenkező irányban felgyorsulnia. Ez a feltétel egyáltalán nem tartható vezérléstechnikailag a fenti esetben. A fenti problémát úgy lehet kikerülni, hogy a menetfűrőt egy rugós kiegyenlítőbetéttel tesszük be a főorsóba, amely kiegyenlíti az $\frac{F}{S}$ hányados értékében

bekövetkező ingadozást.

Más a vezérlés elve a kiegyenlítőbetét kiküszöbölését lehetővé tevő G84.2, G84.3 fűrőciklusoknál. Ezeknél a vezérlő folyamatosan gondoskodik, hogy az $\frac{F}{S}$ hányados pillanatról pillanatra

állandó legyen.

Vezérléstechnikailag az előző esetben a vezérlés csak a főorsó fordulatszámát szabályozza, míg az utóbbiban annak pozícióját is. A G84.2, G84.3 ciklusokban a fűrőtengely és a főorsó mozgását lineáris interpolációval kapcsolja össze. Ezzel a módszerrel a gyorsítási és lassítási szakaszokban is biztosítható az $\frac{F}{S}$ hányados állandósága.

- G84.2: jobbmenet fűrésa kiegyenlítő betét nélkül
- G84.3: balmenet fűrésa kiegyenlítő betét nélkül

A fenti ciklusok csak olyan gépeken alkalmazhatóak, ahol a főorsóra pozíciójeladó van szerelve, és a főhajtás visszacsatolható pozíció szabályzásra. (*INDEX1* paraméter =1.) Ellenkező esetben a vezérlés a kód lehívásakor *3052 G76, G87 HIBA* hibaüzenetet ad.

A ciklusban felhasznált változók:

G17 **G84.** X_p Y_p C Z_p R F S L
G18 **G84.** Z_p X_p C Y_p R F S L
G19 **G84.** Y_p Z_p C X_p R F S L

A ciklus végén a főorsó álló állapotba kerül, szükség esetén újraindításáról a programozónak kell gondoskodni.

Az előtolás és főorsó fordulatszám értékét a fűrő menetemelkedésének függvényében kell megadni:

- G94 percenkénti előtolás állapotban: $F = P \cdot S$

ahol: P: a menetemelkedés mm/ford, vagy inch/ford dimenzióban
S: a főorsó fordulat ford/perc dimenzióban

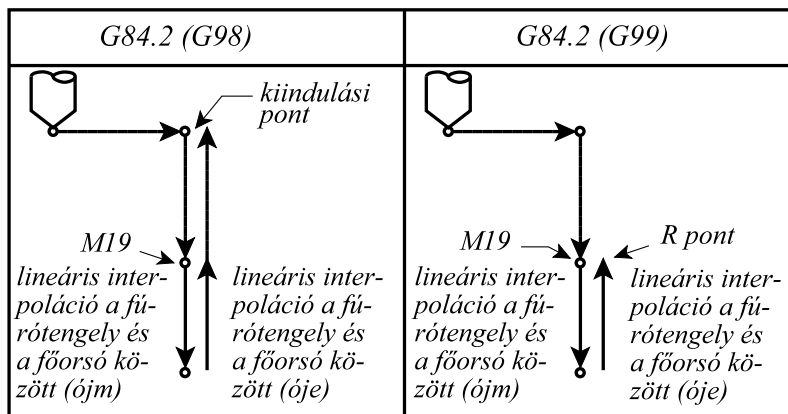
Ebben az esetben az elmozdulás és az előtolás a fűrőtengely és a főorsó mentén a következő (Z-t tételezve fel fűrőtengelynek):

	elmozdulás	előtolás
Z	$z=R$ pont és talppont távolsága	$F'_z = F' \left(\frac{mm}{min} \text{ vagy } \frac{inch}{min} \right)$
S	$S = \frac{z \cdot 360}{F'} \text{ (fok)}$	$F'_s = S \cdot 360 \left(\frac{fok}{min} \right)$

– G95 fordulatonkénti előtolás állapotban: $F=P$

ahol: P: a menetemelkedés mm/ford, vagy inch/ford dimenzióban Ebből látható, hogy fordulatonkénti előtolás állapotban (G95) a menetemelkedés közvetlenül programozható, de az előtolások megállapítására S programozása is szükséges. Ebben az esetben az elmozdulás és az előtolás a fűrőtengely és a főorsó mentén a következő (a Z tengelyt feltételezve fűrőtengelynek):

	elmozdulás	előtolás
Z	$z=R$ pont és talppont távolsága	$F'_z = F' \cdot S \left(\frac{mm}{min} \text{ vagy } \frac{inch}{min} \right)$
S	$S = \frac{z \cdot 360}{F'} \text{ (fok)}$	$F'_s = S \cdot 360 \left(\frac{fok}{min} \right)$

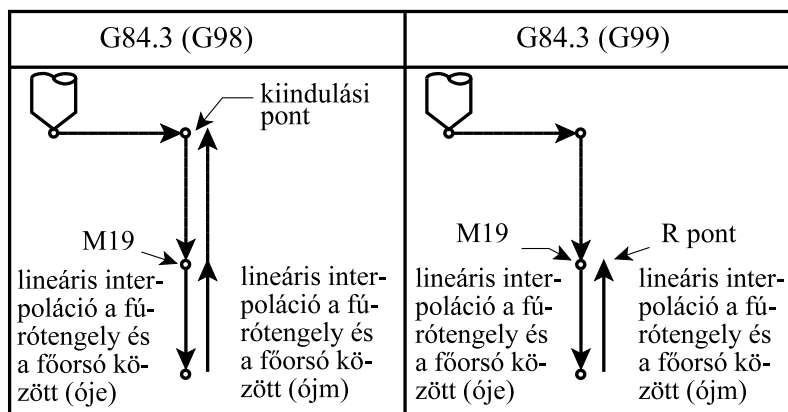


18.1.9-1 ábra

A ciklus műveletei G84.2 estén:

1. művelet: pozicionálás a kiválasztott síkban gyorsmenettel
2. művelet: -
3. művelet: gyorsmeneti mozgás az R pontig (megközelítési pont)
4. művelet: főorsó orientáció: M19
5. művelet: lineáris interpoláció a fűrőtengely és a főorsó között óramutató járásával megegyező (+) főorsó forgásirányban, override és stop tiltva
6. művelet: -

7. művelet: lineáris interpoláció a fűrótengely és a főorsó között óramutató járásával ellentétes (-) főorsó forgásirányban, override és stop tiltva
8. művelet: -
9. művelet: G98 esetén: gyorsmeneti visszahúzás a kiindulási pontig
10. művelet: -

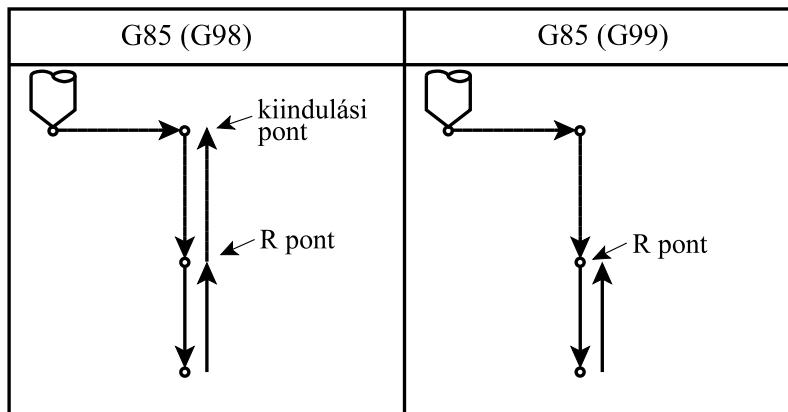


18.1.9-2 ábra

A ciklus műveletei G84.3 estén:

1. művelet: pozicionálás a kiválasztott síkban gyorsmenettel
2. művelet: -
3. művelet: gyorsmeneti mozgás az R pontig (megközelítési pont)
4. művelet: főorsó orientáció: M19
5. művelet: lineáris interpoláció a fűrótengely és a főorsó között óramutató járásával ellentétes (-) főorsó forgásirányban, override és stop tiltva
6. művelet: -
7. művelet: lineáris interpoláció a fűrótengely és a főorsó között óramutató járásával megegyező (+) főorsó forgásirányban, override és stop tiltva
8. művelet: -
9. művelet: G98 esetén: gyorsmeneti visszahúzás a kiindulási pontig
10. művelet: -

18.1.10 Fúróciklus, kiemelés előtolással (G85)



18.1.10-1 ábra

A ciklusban felhasznált változók:

G17 **G85** X_p__ Y_p__ C__ Z_p__ R__ F__ L__

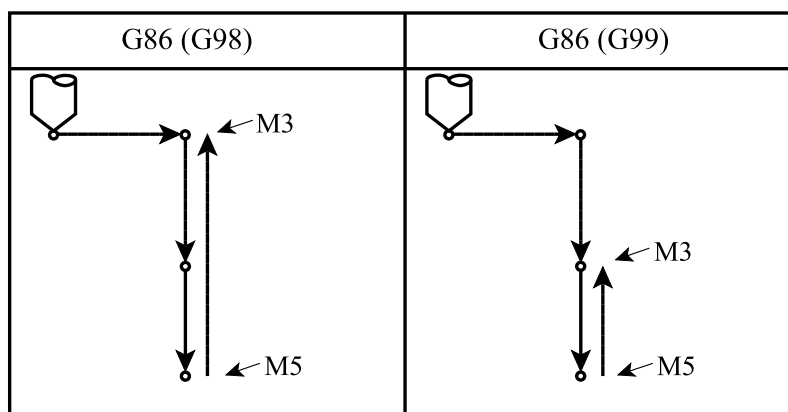
G18 **G85** Z_p__ X_p__ C__ Y_p__ R__ F__ L__

G19 **G85** Y_p__ Z_p__ C__ X_p__ R__ F__ L__

A ciklus műveletei:

1. művelet: pozicionálás a kiválasztott síkban gyorsmenettel
2. művelet: –
3. művelet: gyorsmeneti mozgás az R pontig (megközelítési pont)
4. művelet: –
5. művelet: fúrás a talppontig F előtolással
6. művelet: –
7. művelet: visszahúzás az R pontig, F előtolással
8. művelet: –
9. művelet: G98 esetén: gyorsmeneti visszahúzás a kiindulási pontig
10. művelet: –

18.1.11 Fúróciklus, gyorsmeneti kiemelés álló főorsóval (G86)



18.1.11-1 ábra

A ciklusban felhasznált változók:

G17 **G86** X_p__ Y_p__ C__ Z_p__ R__ F__ L__

G18 **G86** Z_p__ X_p__ C__ Y_p__ R__ F__ L__

G19 **G86** Y_p__ Z_p__ C__ X_p__ R__ F__ L__

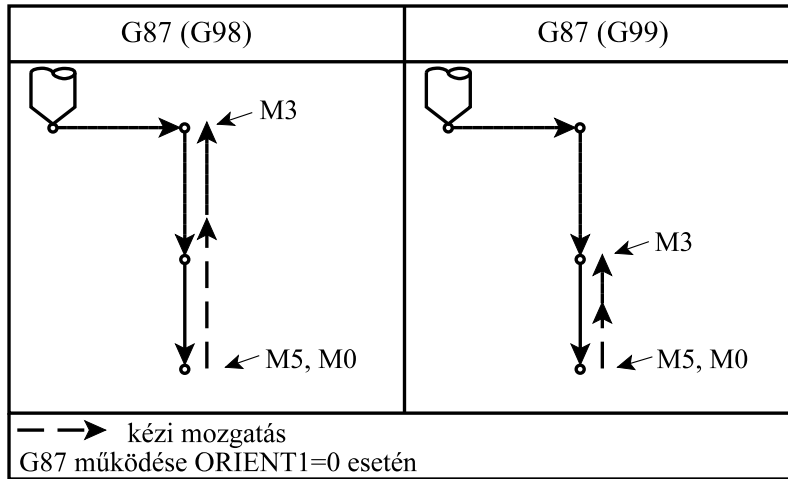
A ciklus indításakor M3-as forgásirányt kell adni a főorsónak.

A ciklus műveletei:

1. művelet: pozicionálás a kiválasztott síkban gyorsmenettel
2. művelet: –
3. művelet: gyorsmeneti mozgás az R pontig (megközelítési pont)
4. művelet: –
5. művelet: fúrás a talppontig F előtolással
6. művelet: főorsó leállítás: M5
7. művelet: G99 esetén: visszahúzás az R pontig, gyorsmenettel
8. művelet: G99 esetén: főorsó újraindítás: M3
9. művelet: G98 esetén gyorsmeneti visszahúzás a kiindulási pontig
10. művelet: G98 esetén: főorsó újraindítás: M3

18.1.12 Fúróciklus, kézi működtetés a talpponton/ Kiesztergálás visszafelé, automatikus szerszámelhúzással (G87)

A ciklust a vezérlő kétféleképp hajtja végre:



18.1.12-1 ábra

A. Fúróciklus, kézi működtetés a talpponton

Abban az esetben, ha a gépen nincs kiépítve a főorsó orientálás lehetősége (*ORIENT1* paraméter=0) a vezérlő az A eset szerint jár el.

A ciklusban felhasznált változók:

G17 **G87** X_p__ Y_p__ C__ Z_p__ R__ F__ L__

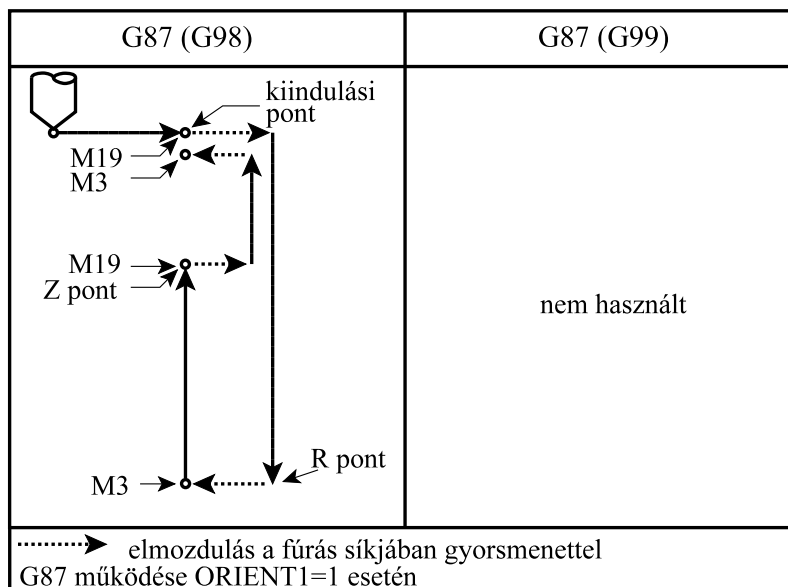
G18 **G87** Z_p__ X_p__ C__ Y_p__ R__ F__ L__

G19 **G87** Y_p__ Z_p__ C__ X_p__ R__ F__ L__

A ciklus indításakor M3-as forgásirányt kell adni a főorsónak.

A ciklus műveletei:

1. művelet: pozicionálás a kiválasztott síkban gyorsmenettel
2. művelet: –
3. művelet: gyorsmeneti mozgás az R pontig (megközelítési pont)
4. művelet: –
5. művelet: fúrás a talppontig F előtolással
6. művelet: – főorsó leállítás: M5
– a vezérlő STOP állapotot vesz fel M0, ahonnan a kezelővalamelyik kézi működtető üzembe (MOZGATÁS, LÉPTETÉS, KÉZIKERÉK) kilépve kézzel működtetheti a gépet, vagyis elhúzhatja a szerszám hegyét a furat felületétől, és kihúzhatja a szerszámot a furatból. Ezután visszalépve AUTOMATA üzembe startra továbbmegy a megmunkálás.
7. művelet: G99 esetén: START után visszahúzás az R pontig, gyorsmenettel
8. művelet: G99 esetén: főorsó újraindítás: M3
9. művelet: 98 esetén: START után gyorsmeneti visszahúzás a kiindulási pontig
10. művelet: G98 esetén: főorsó újraindítás: M3



18.1.12-2 ábra

B. Kiesztergálás visszafelé, automatikus szerszámelhúzással

Abban az esetben, ha a gépen ki van építve a főorsó orientálás lehetősége (*ORIENT1* paraméter=1) a vezérlő a B eset szerint jár el.

A ciklusban felhasznált változók:

G17 **G87** X_p__ Y_p__ C__ I__ J__ Z_p__ R__ F__ L__
 G18 **G87** Z_p__ X_p__ C__ K__ I__ Y_p__ R__ F__ L__
 G19 **G87** Y_p__ Z_p__ C__ J__ K__ X_p__ R__ F__ L__

A ciklus indításakor M3-as forgásirányt kell adni a főorsónak.

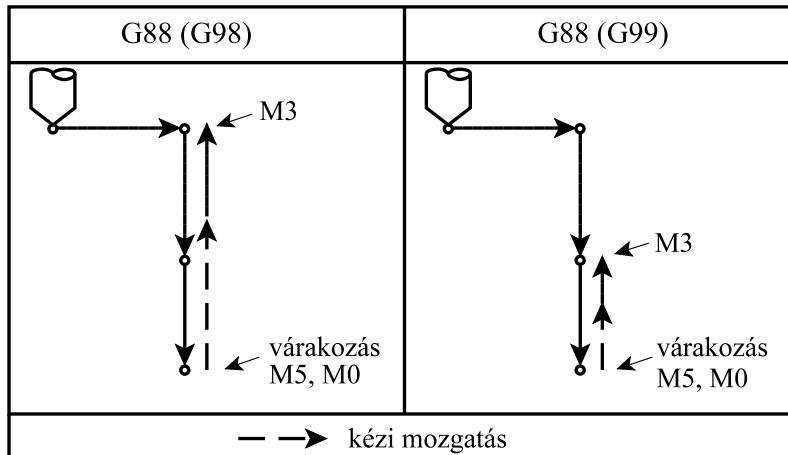
A ciklus műveletei:

1. művelet: pozicionálás a kiválasztott síkban gyorsmenettel
2. művelet: – főorsó orientálás
– szerszámelhúzás a kiválasztott síkban I, J, K értékkel gyorsmenettel
3. művelet: gyorsmeneti mozgás az R pontig (megközelítési pont)
4. művelet: – szerszámvisszahúzás a kiválasztott síkban I, J illetve K-n megadott értékkel ellentétesen, gyorsmenettel,
– főorsó újraindítása M3 irányban
5. művelet: kiesztergálás a talppontig F előtolással
6. művelet: – főorsó orientálás: M19
– szerszámelhúzás a kiválasztott síkban I, J, K értékkel gyorsmenettel
7. művelet: –
8. művelet: –
9. művelet: gyorsmeneti visszahúzás a kiindulási pontig
10. művelet: – szerszámvisszahúzás a kiválasztott síkban I, J illetve K-n megadott értékkel ellentétesen, gyorsmenettel,
– főorsó újraindítása M3 irányba

A ciklus természetéből következően az eddigiekkel ellentétben a megközelítési pont, azaz az R pont mélyebben fekszik, mint a talppont. Ezt a fúrótengely és R címek programozásánál figyelembe kell venni.

Mivel a ciklus a kiesztergálás előtt főorsó orientálást végez és a szerszámot elhúzza a felülettől az I, J illetve K-n megadott értékkel, a behatolásakor elkerülhető a szerszám törése.

18.1.13 Fúróciklus, várakozás után kézi működtetés a talpponton (G88)



18.1.13-1 ábra

A ciklusban felhasznált változók:

G17 **G88** X_p__ Y_p__ C__ Z_p__ R__ P__ F__ L__

G18 **G88** Z_p__ X_p__ C__ Y_p__ R__ P__ F__ L__

G19 **G88** Y_p__ Z_p__ C__ X_p__ R__ P__ F__ L__

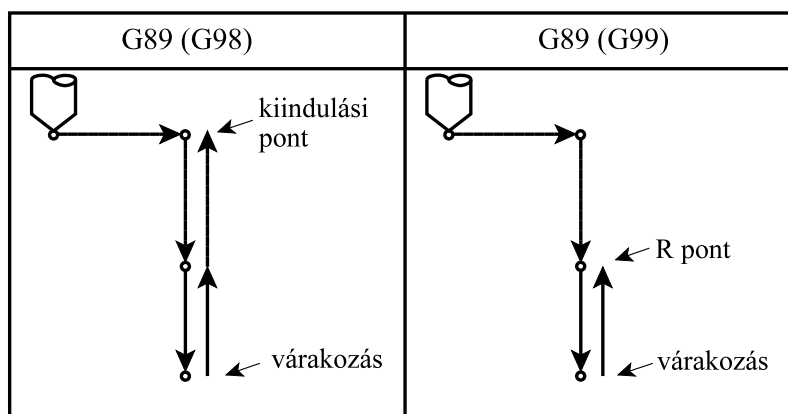
A ciklus indításakor M3-as forgásirányt kell adni a főorsónak.

A ciklus műveletei:

1. művelet: pozicionálás a kiválasztott síkban gyorsmenettel
2. művelet: –
3. művelet: gyorsmeneti mozgás az R pontig (megközelítési pont)
4. művelet: –
5. művelet: fúrás a talppontig F előtolással
6. művelet: – várakozás P értékkel
– főorsó leállítás: M5
– a vezérlő STOP állapotot vesz föl M0, ahonnan a kezelő valamelyik kézi működtető üzembe (MOZGATÁS, LÉPTETÉS, KÉZIKERÉK) kilépve kézzel működtetheti a gépet, vagyis elhúzhatja a szerszám hegyét a furat felületétől, és kihúzhatja a szerszámot a furatból. Ezután visszalépve AUTOMATA üzembe startra továbbmegy a megmunkálás.
7. művelet: G99 esetén: START után visszahúzás az R pontig, gyorsmenettel
8. művelet: G99 esetén: főorsó újraindítás: M3
9. művelet: G98 esetén: gyorsmeneti visszahúzás a kiindulási pontig
10. művelet: G98 esetén: főorsó újraindítás: M3

A ciklus ugyanaz, mint G87 A esete, csak várakozik a főorsó leállítása előtt.

18.1.14 Fúróciklus, talpponton várakozás, kiemelés előtolással (G89)



18.1.14-1 ábra

A ciklusban felhasznált változók:

G17 **G89** X_p Y_p C Z_p R P F L

G18 **G89** Z_p X_p C Y_p R P F L

G19 **G89** Y_p Z_p C X_p R P F L

A ciklus műveletei:

1. művelet: pozicionálás a kiválasztott síkban gyorsmenettel
2. művelet: -
3. művelet: gyorsmeneti mozgás az R pontig (megközelítési pont)
4. művelet: -
5. művelet: fúrás a talppontig F előtolással
6. művelet: várakozás P címen megadott értékig
7. művelet: visszahúzás az R pontig, F előtolással
8. művelet: -
9. művelet: G98 esetén: gyorsmeneti visszahúzás a kiindulási pontig
10. művelet: -

A ciklus megegyezik G85-tel, a várakozást kivéve.

18.2 Megjegyzések a fúróciklusok használatához

- Ciklusüzemmódban ha egy G kód nélküli mondat a következő címek valamelyikét tartalmazza, a fúróciklus végrehajtásra kerül:
X_p, Y_p, Z_p, C vagy R
Ellenkező esetben a fúróciklus nem hajtódik végre.
- Ciklusüzemmódban G04 P várakozási mondatot programozva a parancs végrehajtódik a programozott P szerint, de a várakozásra vonatkozó ciklusváltozó **n e m** törlődik, és **nem** íródik át.
- I, J, K, Q, E, P értékét olyan mondatban kell megadni, ahol fúrás is történik, különben **n e m** tárolódnak el az értékek.

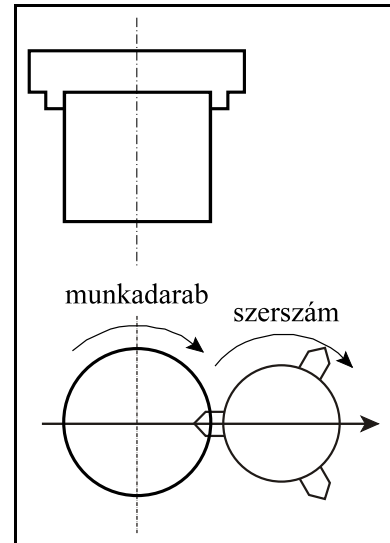
A fentiek illusztrálására tekintsük a következő példát:

G81	X_ C_ Z_ R_ F	(a fúróciklus végrehajtásra kerül)
	X	(a fúróciklus végrehajtásra kerül)
	F_	(a fúróciklus nem kerül végrehajtásra, F felülíródik)
	M_	(a fúróciklus nem kerül végrehajtásra, az M kód végrehajtásra kerül)
G4	P_	(a fúróciklus nem kerül végrehajtásra, a várakozás igen, a várakozási ciklusváltozó nem íródik át)
	I_ Q_	(a fúróciklus nem kerül végrehajtásra, a programozott értékek nem kerülnek ciklusváltozóként bejegyzésre)

- Ha a fúróciklus mellé funkciót is programozunk a funkció az első művelet végén kerül végrehajtásra a pozicionálás befejezése után. Ha a ciklusban L-et is programoztak, a funkció csak az első menetben kerül végrehajtásra.
- A vezérlés mondatonkénti üzemmódban cikluson belül az 1., 3., és a 10. művelet után áll meg.
- A STOP gomb nem hatásos G74, G84, ciklusok 5., 6., és 7. műveletében. Ha ezen műveletek közben nyomnak STOP-ot a vezérlés folytatja működését és csak a 7. művelet végén áll meg.
- Az előtolás és főorsó override függetlenül a kapcsoló állásától mindig 100% a G84.1, G84, G84.2, G84.3 ciklusok 5., 6., és 7. műveletében.
- Ha ciklusmondatban Tnnmm kerül programozásra, az új hosszkorrekciót mind a síkbeli pozicionáláskor, mind a fúráskor figyelembe veszi.

19 A sokszögesztergálás

Sokszögesztergáláskor a szerszámot is és a munkadarabot is forgatjuk egymáshoz képest egy meghatározott fordulatszámaránytal. A fordulatszámarányt és a forgó szerszám vágóéleinek számát változtatva különböző oldalszámú sokszöget kapunk. A fordulatszámarány és a vágóélek számának szorzata adja meg a kiadódó sokszög oldalszámát. Ha pl. a szerszám és a munkadarab fordulatszámaránya 2:1, és a forgó szerszám vágóéleinek száma 2 négyszöget, ha a vágóélek száma 3 hatszöget esztergálhatunk. Így egyszerűen lehet esztergályozással pl. hatlapfejű csavart, vagy anyát készíteni. Az így történő megmunkálás sokkal gyorsabb, mintha polárkoordináta interpoláció alkalmazásával marnánk a sokszöget, azonban a kiadódó oldallapok nem pontosan síkok.



19-1 ábra

19.1 A sokszögesztergálás működési elve

A sokszögesztergálás működési elvét az alábbiakban közöljük.

Legyen a munkadarab forgástengelye az $X=0, Y=0$ koordinátán.

A szerszám forgástengelye legyen a P_0 pont, amelynek koordinátái: $X=A, Y=0$, vagyis a szerszám forgástengelyének a munkadarab forgástengelyétől mért távolsága A .

A forgó szerszám sugara legyen B . A szerszám hegyét jelölje P_t . A szerszám hegyének koordinátája a $t=0$ pillanatban $X=A-B; Y=0$.

A munkadarab forgásának szögsebessége legyen α , a szerszám forgásé pedig β .

A t időpontban a szerszám hegyének koordinátái $P_t(x_t; y_t)$ vetkezők lesznek:

$$x_t = A \cos \alpha t - B \cos(\beta - \alpha)t$$

$$y_t = A \sin \alpha t + B \sin(\beta - \alpha)t$$

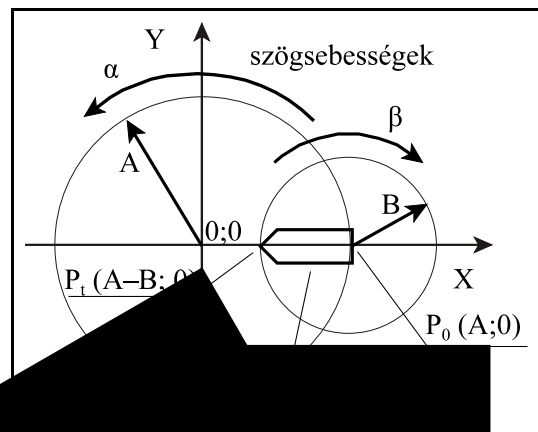
Ha feltételezzük, hogy a szerszám és a munkadarab fordulatszám aránya 2:1, akkor $\beta=2\alpha$. Helyettesítsük be a fenti egyenletbe:

$$x_t = A \cos \alpha t - B \cos \alpha t = (A - B) \cos \alpha t$$

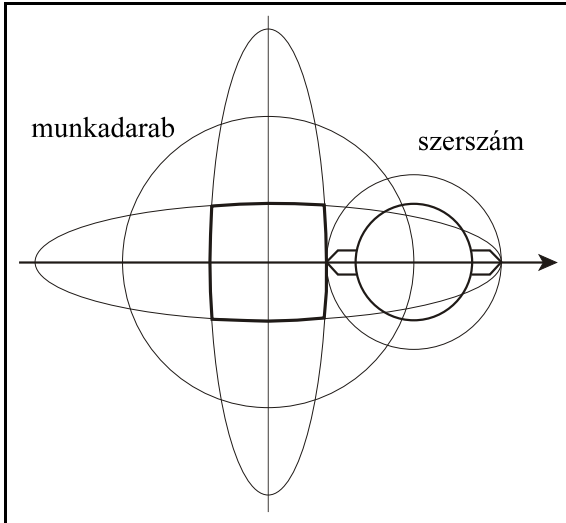
$$y_t = A \sin \alpha t + B \sin \alpha t = (A + B) \sin \alpha t$$

A fenti egyenletek egy olyan ellipszis egyenletei, amelynek nagytengelye $A+B$, kistengelye $A-B$ hosszúságúak.

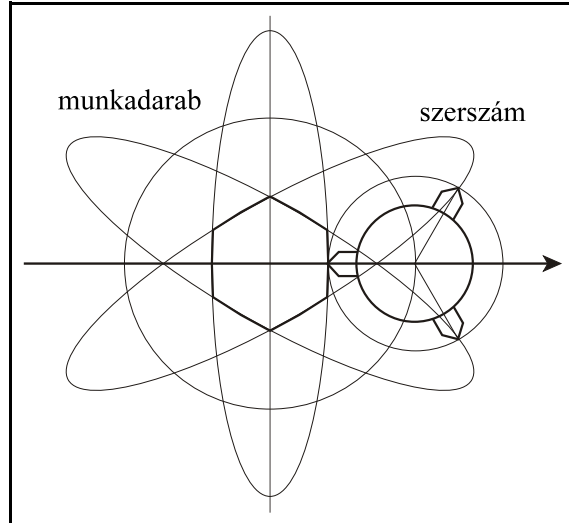
Ha a szerszámok szimmetrikusan egymáshoz képest 180° -ra állnak négyszöget, ha 120° -ra akkor hatszöget esztergálhatunk, feltéve, fordulatszámaránya 2:1.



19.1

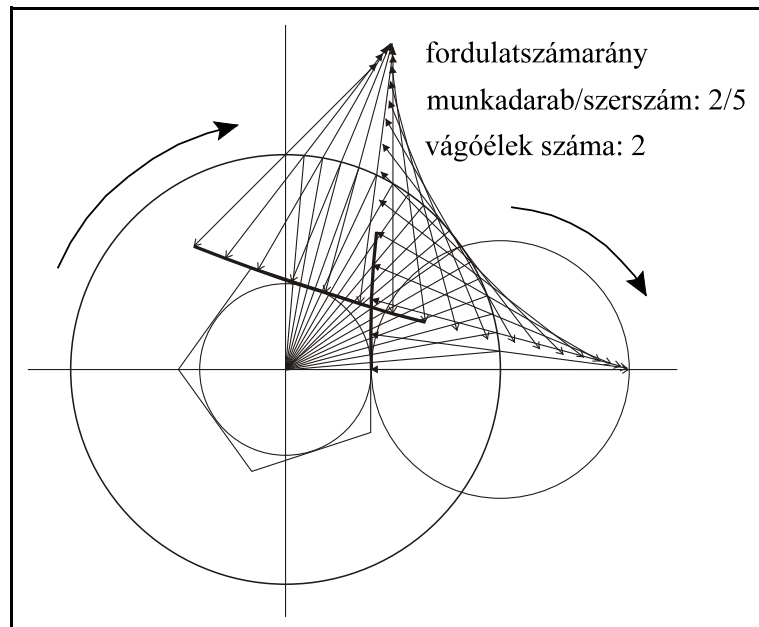


19.1-3 ábra



19.1-4 ábra

Más fordulatszámarányokkal a kiadódó görbék ellipszisztól különbözőek lesznek, de még ezekkel is esetleg jól közelíthetőek sokszögek oldalai.



19.1-5 ábra

19.2 A sokszögesztergálás programozása (G51.2, G50.2)

A

G51.2 P_ Q_

utasítás bekapcsolja a sokszögesztergálás funkcióját. A P és Q címeken lehet beállítani a munkadarabnak a szerszámhoz viszonyított fordulatszámarányát. Ha pl. a munkadarabnak és a szerszámnak 1:2 arányban kell forognia programozunk G51.2 P1 Q2-t. P és Q címek értéktartománya:

$$P = 1 \dots 127$$

$$Q = -127 \dots +127$$

Q címen negatív számot is megadhatunk, ekkor a szerszámorsó forgásiránya ellentétes lesz.

A

G50.2

utasítás kikapcsolja a sokszögesztergálást.

A G51.2 és a G50.2 utasítást mindig külön mondatban kell megadni.

A sokszögesztergálási funkció végrehajtásához ki kell a szerszámgépen építeni a második főorsót is, amely a szerszámot képes forgatni. Mindkét orsóra, a munkadaraborsóra is és a szerszámorsóra is jeladót kell szerelni. A funkció bekapcsolása során az NC először fölpörgeti a szerszámorsót arra a fordulatra, amely a munkadarab orsóra programozott értékből (S cím) és a P, Q címre programozott értékből kiadódik: $n_{\text{szerszámorsó}} = S \cdot (Q/P)$. Ezután a két jeladó nullimpulzusait összeszinkronozza, majd a munkadarab orsó jeladójáról mért elfordulás alapján forgatja a szerszámorsót a programozott Q/P arány alapján.

Az összeszinkronozás hatása hasonló, mint a menetvágásnál a főorsó nullimpulzusé. Amíg a munkadarabot nem vesszük ki a tokmányból és amíg az ütőkést ugyanolyan szöghelyzetben hagyjuk a szerszámorsóban, addig a munkadarab és a szerszám mozgása szinkron marad, többször végigmehetünk ugyanazon a felületen, például nagyolás majd símitás céljából. Ugyancsak kikapcsolhatjuk a szinkronizálást a G50.2 utasítással, majd visszakapcsolva azt ugyanúgy végigmehetünk az előzőleg esztergált sokszög felületén, feltéve, hogy a programozott fordulatszám arány ugyanaz, mint előzőleg.

☞ *Figyelem!*

Sokszögesztergálás programozása során ügyeljünk arra, hogy a szerszámorsóra kiadódó fordulatszám $n_{\text{szerszámorsó}} = S \cdot (Q/P)$ soha ne haladja meg az arra engedélyezett maximális fordulatszámot.

A szinkronfutást kikapcsolja

- a vészstop,
- a szervoköri hibák

Mintapélda

```

...
G0 X120 Z5 T505
S1000 M3 (munkadarab orsó indítás 1000 ford/perc)
G51.2 P1 Q2 (sokszögesztergálás bekapcsolása, szerszámorsó
fordulat 2000 ford/perc)
G0 X100 (fogásvétel X irányban)
G1 Z-50 F0.01 (esztergálás)
G4 P2 (várakozás)
G0 X120 (kiemelés X-ben)
G50.2 (sokszögesztergálás kikapcsolása)
...

```

20 Mérőfunkciók

20.1 Mérés a maradék út törlésével (G31)

A

G31 v (F) (P)

utasítás hatására a v koordinátájú pontra lineáris interpolációval elindul a mozgás. A mozgás addig tart, amíg egy külső jel (pl. egy mérő-tapintó jele) be nem érkezik, vagy a v koordinátákon megadott végponti pozíciót el nem érte a vezérlés. A jel beérkezése után a vezérlés lelassít és megáll.

P címen lehet megadni, hogy a vezérlésbe bevezethető 4 törlőjel közül melyiket vegye figyelembe a mozgás során:

P0: 1. törlőjel figyelembe vétele

P1: 2. törlőjel figyelembe vétele

P2: 3. törlőjel figyelembe vétele

P3: 4. törlőjel figyelembe vétele.

Ha a P címet nem töltjük ki, az 1. sz. törlőjelet vesz figyelembe.

A G31 utasítás nem öröklődik, csak abban a mondatban érvényes, amelyikben programozták. Ha a G31 utasításban szintaktikai hibát talál a vezérlés 3051 G22, G28, ... G31, G37 hibajelzést ad.

A mozgás sebessége:

- a megadott, vagy öröklődött F érték, ha a *SKIPF* paraméter =0
- a *G31FD* paraméterről felvett előtolásérték, ha a *SKIPF* paraméter =1.

A külső jel bejövetele pillanatában a tengelyek pozíciója eltárolódik az alábbi rendszerváltozókon:

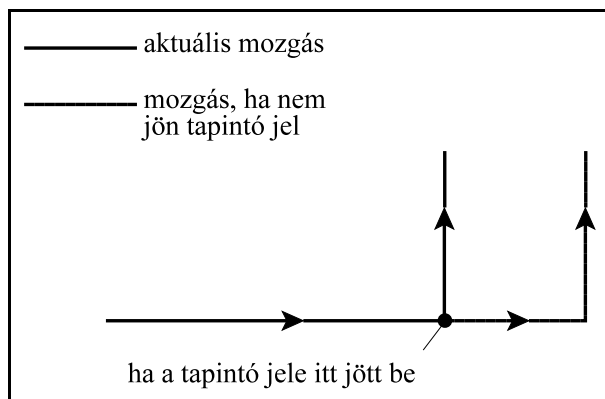
#5061.....1. tengely pozíciója

#5062.....2. tengely pozíciója

.

.

#5068.....8. tengely pozíciója



20.1-1 ábra

Az itt eltárolt pozíció:

- ha jött külső jel, a jel bejövételének pillanatában felvett pozíció,
- ha nem jött külső jel a G31 mondat programozott végpontjának pozíciója,
- mindig az aktuális munkadarab koordinátarendszerben értendő,
- az aktuális hosszkorrekció figyelembe vételével.

A külső jel bejövetele után a mozgás lineáris lassulással leáll. A G31 mondat végponti pozíciója ekkor a mondatban alkalmazott előtolás függvényében kis mértékben eltér a jel bejövetelekor a #5061... változókon letárolásra került pozícióktól. A mondatvégponti pozíciók a #5001... változókon érhetők el. A következő mozgásmondat ezektől a végponti pozícióktól kezdve fog érvényesülni.

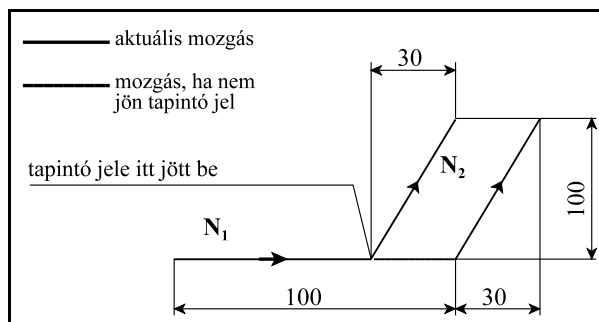
A mondat végrehajtása csak G40 állapotban lehetséges. G41 vagy G42 állapot esetén G31 programozása 3054 G31 ROSSZ ÁLLAPOTBAN hibajelzést vált ki. Szinténezt a hibajelzést adja, ha G95 állapot, ha G51, G51.1, G68 állapot, vagy, ha G16 állapot van érvényben.

A v koordinátákon megadott érték lehet inkrementális és abszolút is. Ha a következő mozgásparancs koordinátamegadása inkrementális, az elmozdulást a G31 mondat azon pontjától számítja, ahol a mozgás az előző mondatban abbamaradt.

Például:

```
N1 G31 G91 Z100
N2 X100 Z30
```

N1 mondatban elindít egy inkrementális mozgást Z irányban. Ha a külső jel bejövetele után a vezérlő az Z=86.7 koordinátájú ponton áll meg, ettől a ponttól számítva lép inkrementálisan 100-at X irányban, 30-at a Z irányban az N2 mondatban.

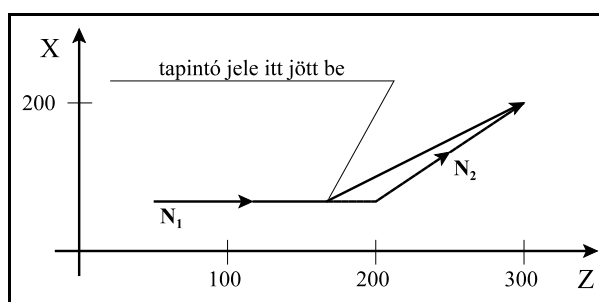


20.1-2 ábra

Ha abszolút adatmegadást programoztunk a mozgás a következőképp alakul:

```
N1 G31 G90 Z200
N2 X200 Z300
```

N1 mondat elindít Z irányban egy mozgást az Z=200 koordinátájú pontra. Ha a külső jel bejövetele után a vezérlő a Z=167 koordinátájú ponton áll meg, az N2 mondatban az Z irányú elmozdulás Z=300-167, azaz Z=133 lesz.



20.1-3 ábra

20.2 Automatikus szerszámhossz mérés (G36, G37)

A

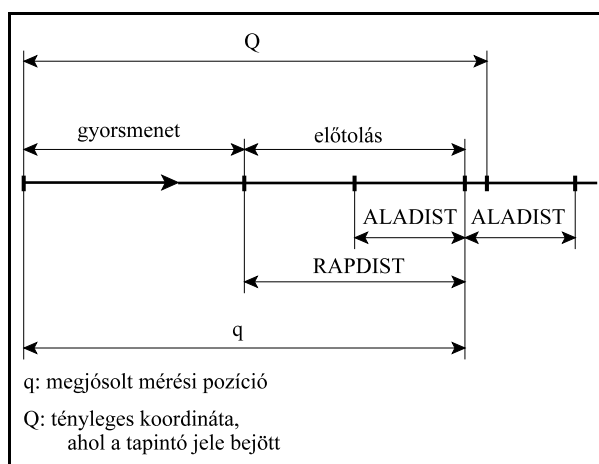
```
G36 X__ ,
G37 Z__
```

utasítás hatására az X koordinátán G36 esetén, illetve a Z koordinátán G37 esetén a megadott irányban gyorsmenettel elindul a mozgás. X, Z értéke mindig abszolút adatként kerül értelmezésre.

A mozgás az X - *RAPDISTX*, illetve a Z - *RAPDISTZ* pozícióig megy gyorsmenettel, ahol *RAPDISTX*, *RAPDISTZ* paraméteren beállított értékek. Ezután a mozgás a *G37FD* paraméteren megadott előtolással folytatódik, amíg a mérő-tapintó jele be nem jön, vagy amíg a vezérlés hibát nem jelez. *3103 HATÁRON KÍVÜL* hibajelzést akkor ad a készülék, ha az X, vagy a Z címen programozott pozíció (megjósolt mérési pozíció) *ALADISTX*, *ALADISTZ* sugarú környezetén kívül jön meg a tapintó jele.

Ha a mérés sikerrel járt, és a tapintó jele a Q koordinátájú ponton jött be, a vezérlés az érvényben lévő kopáskorrekciós regiszterhez

– hozzáadja (ha az *ADD* paraméter =1) a Q-q különbséget,



20.2-1 ábra

– vagy kivonja abból (ha az *ADD* paraméter =0).

A mérés megkezdése előtt a megfelelő korrekciós értéket le kell hívni Tnnmm utasítással.

– A G36, G37 utasítás egylovétű.

– A G36, G37 ciklus mindig az aktuális munkadarab koordinátarendszerben kerül végrehajtásra.

– A *RAPDIST* és *ALADIST* paraméterek mindig pozitív értékek. A két paraméterre a következő feltételnek kell teljesülni: *RAPDIST*>*ALADIST*

– Szintaktikai hiba esetén 3051 G22, G28, ... G31, G37 hibajelzést ad

– A korrekcióváltást (Tnnmm) a G36, G37 mondatban nem lehet megadni. Ellenkező esetben 3055 G37 ROSSZ ÁLLAPOTBAN hibajelzést ad.

– Szintén a fenti hibajelzést adja, ha G51, G51.1, G68 állapot van érvényben.

A G36, G37 funkció végrehajtása közben előjövő hibüzenetek a következők:

– 3103 *HATÁRON KÍVÜL* üzenet, ha a G36, G37 mondatban programozott végpozíció *ALADISTX*, illetve *ALADISTZ*, sugarú környezetén belül nem jön meg a tapintó jele.

20.3 Összefüggés a mérés pontossága és az alkalmazott előtolás között

Az NCT vezérlők hardver felépítése általában lehetővé teszi, hogy a mérés során alkalmazott előtolástól független legyen a mérés pontossága.

Az alábbi esetek kivételt képeznek:

– NCT104: azokon a tengelyeken, ahol ABSOLUTEN paraméter =1, azaz, ahol a tengely abszolút mérőrendszerrel van szerelve,

– NCT115: azokon a tengelyeken, ahol nincs XMU kártya használva, hanem a vezérlő a hajtástól kapja a pozíció adatokat.

A fenti esetekben a mérés pontossága: $(F/60) \cdot T_{\text{ciklusidő}}!$

NCT104 esetén a ciklusidő: $T_{\text{ciklusidő}}=0.001 \text{ sec}$

NCT115 esetén a ciklusidő: $T_{\text{ciklusidő}}=0.002 \text{ sec}$

21 Biztonsági funkciók

21.1 Programozható munkatér behatárolás (G22, G23)

A

G22 X Y Z I J K P

utasítás bekapcsolja a munkatér behatárolás figyelését. Az utasítással a tengelyek mozgástartományá korlátozható be. Az utasítás címeinek jelentése:

- X: az X tengelyen a pozitív irányú határ,
- I: az X tengelyen a negatív irányú határ,
- Y: az Y tengelyen a pozitív irányú határ,
- J: az Y tengelyen a negatív irányú határ,
- Z: az Z tengelyen a pozitív irányú határ,
- K: az Z tengelyen a negatív irányú határ,

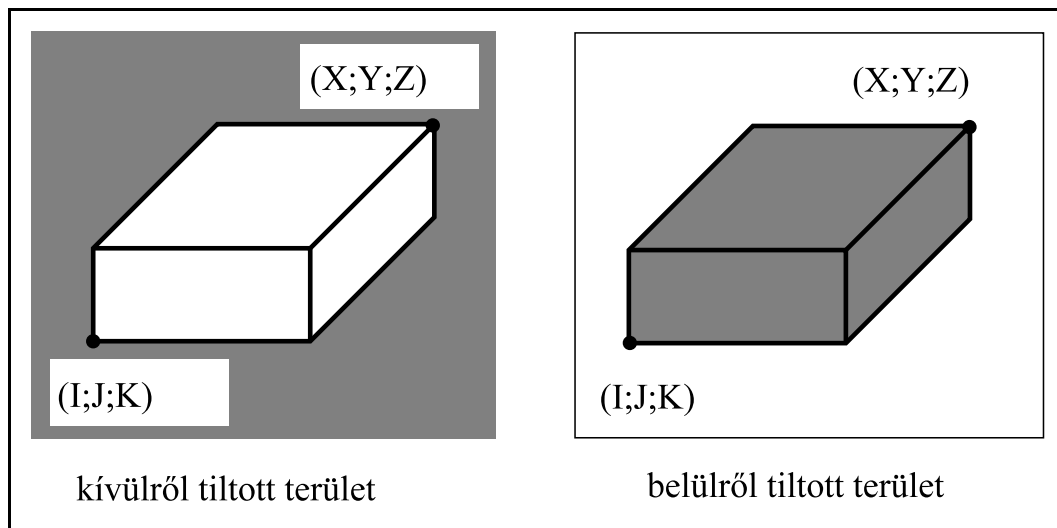
A következő feltételeknek kell teljesülni a megadott adatokra:

$$X \geq I, Y \geq J, Z \geq K$$

P címen adható meg, hogy a kijelölt teren kívülre, vagy belülré nem szabad menni.

P=0 esetén a kijelölt tér belseje van tiltva,

P=1 esetén a kijelölt tér külseje van tiltva.



21.1-1 ábra

A

G23

utasítás kikapcsolja a munkatérbehatárolás figyelését.

A G22, G23 utasítások közvetlenül átírják a paramétermező megfelelő elemeit.

A G22 utasítás az *STRKEN* paramétert 1-be, a G23 0-ba írja

A G22 P0 utasítás az *EXTER* paramétert 0-ba, a G22 P1 1-be írja.

A G22 utasításban előforduló X, Y, Z koordináták a megfelelő tengelyekhez tartozó *LIMP2n* paramétereket írják, az I, J, K koordináták a megfelelő tengelyekhez tartozó *LIMN2n* értékeket.

A G22 utasításban szereplő koordináták mielőtt a megfelelő paraméterre íródna át számításra kerülnek a gép koordináta-rendszerére úgy, hogy a bekapcsolt korrekciós eltolásokat is tartalmazzák. Tehát például, ha a G22 utasítás megadásakor be volt kapcsolva a hosszkorrekció Z

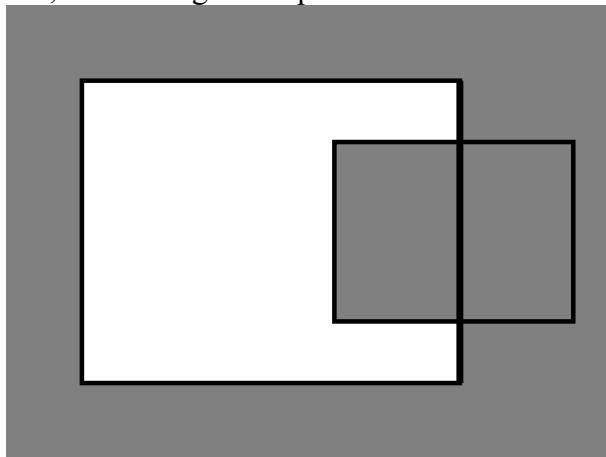
irányban, az erre a tengelyre megadott koordináta határadatok úgy határolják be a mozgást, hogy a szerszám hegyét nem engedik a határon túl. Ha viszont nincs korrekció bekapcsolva a szerszámtartó vonatkoztatási pontját nem engedi a tiltott területre. A szerszám hossz tengelyébe eső koordinátán a leghosszabb kinyúlású szerszámhoz célszerű beállítani a tiltott zóna határát.

- Munkatérbehatárolást csak a fő tengelyekre lehet megadni.
- A G22, G23 utasításokat önálló mondatban kell megadni.
- A munkatér behatárolása bekapcsolás és gépi referenciapont felvétel után lesz hatásos.
- Ha referenciapont felvétel után, vagy G22 programozás hatására a gép tiltott munkatérbe kerül, és a zóna belülről van tiltva kézi üzemmódban G23 programozásával oldani kell a tiltást, a tengely(ek)e)t kihozni onnan kézi mozgatással, majd G22 programozásával visszakapcsolni a tiltott terület figyelést. Ha a zóna kívülről van tiltva, a terület elhagyása ugyanaz, mint a végállásra futás után.
- Ha mozgás közben kerül valamelyik tengely a tiltott terület határára, onnan kézi mozgatással eltávolítható kézi üzemmódban.
- Ha $X=I$, $Y=J$, $Z=K$ és $P=0$ a teljes tér engedélyezve van.
- Ha $X=I$, $Y=J$, $Z=K$ és $P=1$ a teljes tér tiltva van.
- Ha a munkatér belülről van tiltva és a tiltott zónába, vagy annak határára jutnak a tengelyek *1400 BELÜLRŐL TILTOTT ZÓNA* hibüzenetet ad a vezérlő.
- Ha a munkatér kívül van tiltva *130n TILTOTT ZÓNA t+*, vagy *132n TILTOTT ZÓNA t-* hibajelzést ad a vezérlő, ahol "t" a tiltott zónára futó tengely neve.

21.2 Paraméteres végállás

A vezérlés paraméterein a gép építője tengelyenként meghatározhatja az adott gépen fizikailag megengedhető mozgástartományt, vagyis a végállások határait. Ha a vezérlés eléri ennek a tartománynak a határát ugyanúgy végállás hibát jelez, mintha végálláskapcsolóra futott volna.

- A paraméteres végállásfigyelést a vezérlő csak a gépi referenciapont felvétel után végzi.
- A paraméteres végállásfigyelés mindig külön teret tilt le.
- A paraméteres végállásfigyelés és a programozott munkatérbehatárolás által kijelölt tartományok átfedhetnek egymást.



21.2-1 ábra

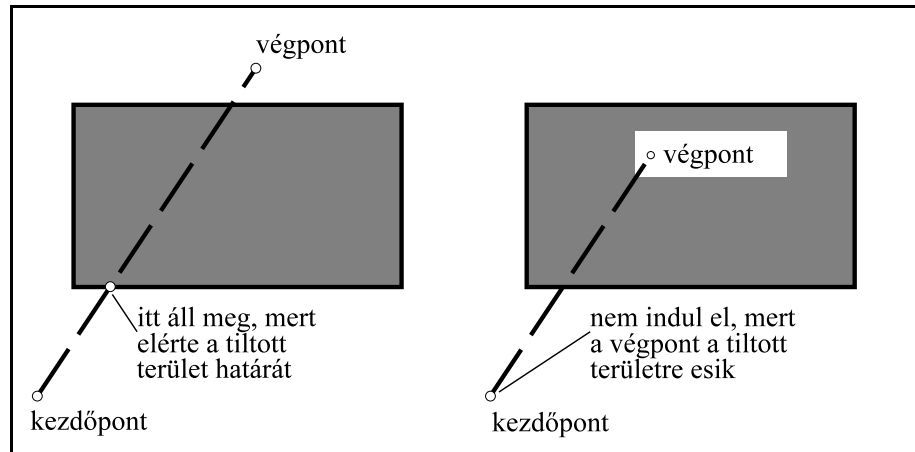
21.3 Tiltott tartomány figyelés mozgásindítás előtt.

A vezérlés két tiltott tartományt különböztet meg. Az első, amely a gép fizikailag lehetséges mozgástartományát határolja be. Ezen mozgástartomány szélső pozícióit nevezzük végállásoknak. Mozgás közben a vezérlő ezen, paramétereken meghatározott tartományon kívülre nem engedi a tengelyeket mozogni. A végállásokat a gép építője állítja be, ezeken a paramétereken a felhasználónak nem szabad módosítani.

A másik tiltott tartományt, amit programozható munkatér behatárolásnak is nevezünk a felhasználó állapítja meg. Ez történhet a G22 utasítás programozásával, vagy paraméterátírással.

Ha a paramétermezőben a *CHBFMOVE* paraméter 1-be van állítva a vezérlés mielőtt a mondatvégrehajtás során elindítaná a tengelyeket ellenőrzést végez,

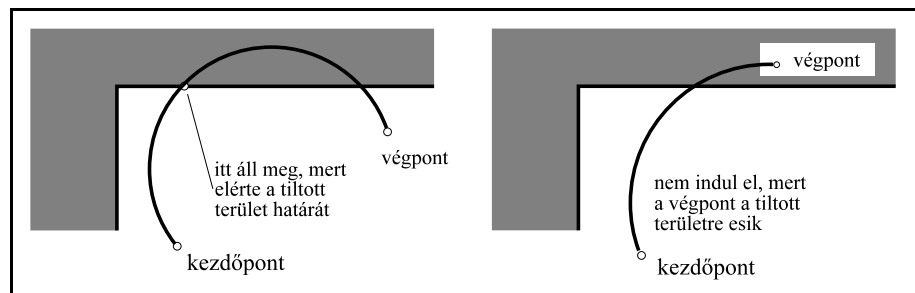
hogy az adott mondat programozott végpontja nem esik-e valamelyik tiltott területbe.



21.3-1 ábra

Ha az adott mondat végpontja a végállástartományon kívülre esik **3056 VÉGÁLLÁS** hibajelzést ad, ha a programozott tiltott munkatérbe esik **3057 TILTOTT TERÜLET** hibajelzést ad. Így

gyakorlatilag a mozgás el sem indul.



21.3-2 ábra

Mivel a mondatindítás előtt a vezérlő csak azt ellenőrzi, hogy a mondat végpontja nem esik-e a tiltott területek valamelyikébe az ábrákon látható esetekben csak a tartomány határán, mozgás után jön hibüzenet.

22 A felhasználói makró

22.1 Az egyszerű makróhívás (G65)

A

G65 P(programszám) L(ismétlési szám) <argumentum kijelölés>
 utasítás hatására a P címen (programszám) megadott számú makróprogram az L címen megadott számmal egymás után ismételtlen meghívódik.

A makróprogramnak argumentumokat lehet átadni. Az argumentumok meghatározott címeknek adott olyan konkrét számértékek, amelyek a makróhívás során a megfelelő lokális változóban kerülnek eltárolásra. A makróprogram ezeket a lokális változókat fel tudja használni, vagyis a makróhívás olyan speciális alprogramhívás, ahol az alprogramnak a főprogram változókat (paramétereket) tud átadni.

A G65 utasításban kétféle argumentumkijelölés lehetséges:

1. sz. argumentumkijelölés címlánca:

A B C D E F H I J K M Q R S T U V W X Y Z

A **G, L, N, O, P** címeken nem lehet értéket átadni a makróprogramnak, más szóval ezek a címek nem használhatók argumentumkijelölésre a G65 utasításban. A címeket tetszőleges sorrendben lehet kitölteni, nem szükséges, hogy ABC sorrendben írjuk be őket.

2. sz. argumentumkijelölés címlánca:

A B C I1 J1 K1 I2 J2 K2 ... I10 J10 K10

Az A, B, C címeken kívül az I, J, K címekre maximum 10 különböző argumentum jelölhető ki. A címeket tetszőleges sorrendben lehet kitölteni. Ha több argumentumot jelölünk ki ugyanarra a címre a változók a kijelölés sorrendjében veszik fel a megfelelő értéket.

<i>lv</i>	<i>1. ak</i>	<i>2. ak</i>
#1	A	A
#2	B	B
#3	C	C
#4	I	I1
#5	J	J1
#6	K	K1
#7	D	I2
#8	E	J2
#9	F	K2
#10	(G)	I3
#11	H	J3

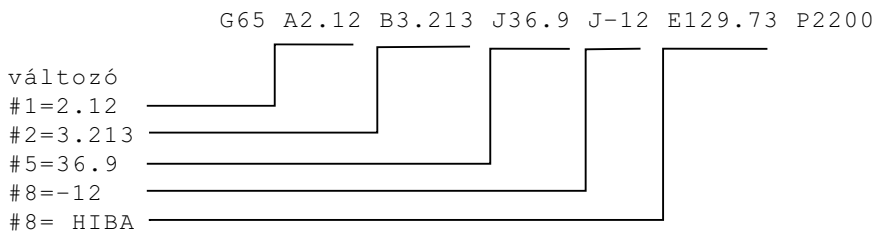
<i>lv</i>	<i>1. ak</i>	<i>2. ak</i>
#12	(L)	K3
#13	M	I4
#14	(N)	J4
#15	(O)	K4
#16	(P)	I5
#17	Q	J5
#18	R	K5
#19	S	I6
#20	T	J6
#21	U	K6
#22	V	I7

<i>lv</i>	<i>1. ak</i>	<i>2. ak</i>
#23	W	J7
#24	X	K7
#25	Y	I8
#26	Z	J8
#27	–	K8
#28	–	I9
#29	–	J9
#30	–	K9
#31	–	I10
#32	–	J10
#33	–	K10

– **rövidítések:** *lv*:lokális változó, *1.ak*: 1. sz. argumentumkijelölés, *2.ak*: 2. sz. argumentumkijelölés.

Az I, J, K címek utáni indexek az argumentumkijelölés sorrendjét mutatják.

Az 1.sz. és a 2. sz. argumentumkijelölés együtt is létezhet egy mondaton belül, a vezérlés elfogadja azt. Hibát akkor jelez, ha egy adott számú változóra kétszer akarunk hivatkozni. Például:



Ebben a példában #8–nak a második J cím (értéke –12) már adott értéket, és mivel az E cím értékét is a #8 változó veszi fel a vezérlés *3064 HIBÁS MAKRÓ KIFEJEZÉS* hibát üzen.

Az argumentumkijelölésre szánt címek értékhatára megegyezik normál felhasználásuk értékhatárával. Tizedespont, előjel, I operátor is átadható azokon a címeken, ahol ezek normál programban használhatók.

22.2 Az öröklődő makróhívás

22.2.1 Makróhívás minden mozgásparancs után: (G66)

A

G66 P(programszám) L(ismétlési szám) <argumentum kijelölés>
 utasítás hatására a P címen (programszám) megadott számú makróprogram az L címen megadott számmal egymás után ismételt meghívódik minden mozgásparancs végrehajtása után. P és L címek értelmezése, valamint az argumentumkijelölés szabálya megegyezik a G65 utasításnál elmondottakkal.

A kijelölt makró mindaddig meghívódik, amíg a

G67

makróhívás öröklődésének törlőparancsát nem programoztuk.

Például: az alkatrészprogram egy adott szegmensén minden mozgás után egy furatot kell készíteni:

főprogram

```

...
G66 P1250 Z-100 R-1 X2 F130      Z(furat talppontja) R(furat R pontja)
                                  X(várakozási idő) F(előtolás)

G91 G0 X100                      a fúrás minden pozicionálás után végre-
Y30                               hajtja
X150
...
G67

```

makróprogram

```
%O1250
G0 Z#18 (gyorsmeneti pozicionálás z irányban az R címen meg-
adott pontra)
G1 Z#26 F#9 (az F címen megadott előtolással fúrás a Z címen meg-
adott talppontig)
G4 P#24 (várakozás a furat alján az X címen megadott érték-
kel)
G0 Z-[#18+#26] (a szerszám visszahúzása a kiindulási pontra)
M99 (visszatérés a hívó programba)
%
```

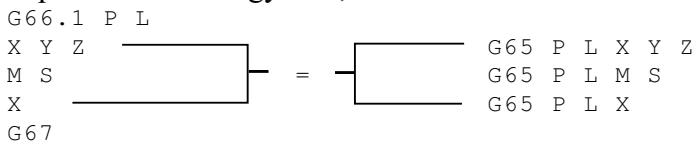
22.2.2 Makróhívás minden mondatból: (G66.1)

A

G66.1 P(programszám) L(ismétlési szám) <argumentum kijelölés>

parancs hatására az összes utána következő mondatot argumentumkijelölésnek értelmez és a P címen megadott számú makróhívást hívja be, azt hajtja végre az L címen megadott számú ismétléssel.

A parancs hatása ugyanaz, mintha minden mondat G65-ös makróhívás lenne:



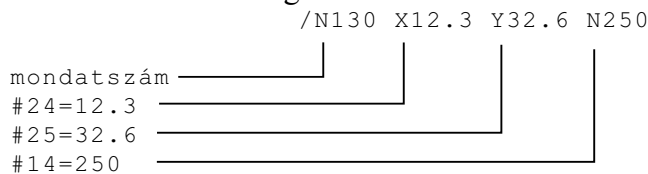
A kijelölt makró mindaddig meghívódik, amíg a

G67

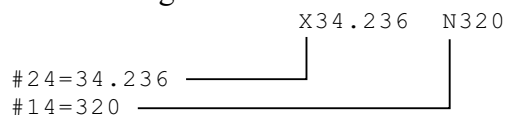
makróhívás öröklődésének törlőparancsát nem programoztuk.

Az argumentumkijelölés szabályai:

1. A bekapcsolást végző mondatban (ahol a G66.1 P L-t programoztuk):
Az argumentumkijelölésre használható címek ugyanazok, mint a G65 parancs esetén.
2. A G66.1 utasítást követő mondatokban:
A G65 parancs esetén használható címek, valamint
L: #12,
P: #16,
G: #10 azzal a megkötéssel, hogy a vezérlő egy mondatban csak egy G címre történő hivatkozást fogad el, ha több G címet programoztunk *3005 ILLEGÁLIS G KÓD* hibajelzést ad.
N: #14 ha egy N cím a mondat elején áll (legfőljebb a / feltételes mondat címe előzi meg) a második N címet argumentumnak veszi:



ha az N cím a mondat közepén található (bármely /-től különböző cím megelőzi) az N cím argumentumként kerül értelmezésre:



ha N cím már egyszer argumentumként került bejegyzésre a következő N címre történő hivatkozás már *3064 HIBÁS MAKRÓ KIFEJEZÉS* hibajelzést eredményez.

A mondatvégrehajtás szabályai G66.1 esetén:

A kijelölt makró már abból a mondatból meghívódik, ahol a G66.1 kódot megadtuk, figyelembe véve az 1. pont alatti argumentumkijelölési szabályokat.

A G66.1 kódot követő mondatról a G67 kódot tartalmazó mondatig minden NC mondat makróhívást eredményez a 2. pont argumentumkijelölési szabályai alapján. Nem hívódik meg a makró, ha üres mondatot talál, pl.: N1240, ahol csak egy N címre történt hivatkozás, illetve makróutasítást tartalmazó mondatból.

22.3 Felhasználói makróhívás G kódra

A paramétermezőben ki lehet jelölni legfeljebb 10 különböző G kódot, amelyre makróhívást akarunk kezdeményezni. Ekkor az

Nn G65 Pp <argumentum kijelölés>

utasítással helyett az

Nn Gg <argumentum kijelölés>

utasítással kell leírni. A paramétermezőben azt kell beállítani, hogy a hívó G kód melyik programszámot hívja. G65, G66, G66.1, G67 kód nem adható meg ilyen célra.

G(9010): G kód, amelyik az O9010 sz. programot hívja

G(9011): G kód, amelyik az O9011 sz. programot hívja

:

G(9019): G kód, amelyik az O9019 sz. programot hívja

Ha a paraméter mezőben negatív értéket adunk meg, akkor a kijelölt G kód modális hívást generál. Például, ha a G(9011)=−120, akkor a programban a G120 utasítás öröklődő hívást eredményez. Azt, hogy a hívás típusa milyen legyen a

MODGEQU=0: a G hívás G66 típusú

MODGEQU=1: a G hívás G66.1 típusú

paraméter állapota határozza meg. Ha a paraméter értéke 0, a makró minden mozgásmondat végén hívódik. Ha a paraméter értéke 1 a makró minden mondatra meghívódik.

Ha egy sztenderd G kódot jelölünk ki felhasználói hívásra (pl. G01-et) és a makró törzsében ismét erre a kódra hivatkozunk ez a hivatkozás már nem eredményez újabb makróhívást, hanem a vezérlés, mint közönséges G kódot értelmezi és hajtja végre azt.

Ha a felhasználói hívásban a makró törzsben ismételten a hívó G kódra hivatkozunk, és ez nem sztenderd G kód *3005 ILLEGÁLIS G KÓD* hibajelzést ad a vezérlő.

– felhasználói G kód hívásból felhasználói M, S, T, A, B, C hívása,

– felhasználói M, S, T, A, B, C hívásból felhasználói G kód hívása,

paraméterállástól függően engedélyezett:

FGMAC=0: nem engedélyezett (közönséges M,S, ...G kódként kerülnek végre hajtásra),

FGMAC=1: engedélyezett, azaz új hívás keletkezik.

A felhasználói G kódok argumentumkészlete:

– ha a kód G65 vagy G66 típusú a G65-höz rendelt argumentumkészlet, valamint P és L,

– ha a kód G66.1 típusú, akkor argumentumkészletére az ott elmondottak érvényesek.

Az öröklődő hívás törlése G67 utasítással történik.

22.4 Felhasználói makróhívás M kódra

A paramétermezőben ki lehet jelölni legfeljebb 10 különböző M kódot, amelyre makróhívást akarunk kezdeményezni. Ekkor az

Nn **Mm** <argumentum kijelölés>

utasítást kell leírni. Az M kód ebben az esetben nem kerül a PLC-nek átadásra, hanem a megfelelő programszámú makró kerül hívásra.

A paramétermezőben azt kell beállítani, hogy a hívó M kód melyik programszámot hívja.

M(9020): M kód, amelyik az O9020 sz. programot hívja

M(9021): M kód, amelyik az O9021 sz. programot hívja

:

M(9029): M kód, amelyik az O9029 sz. programot hívja

M kóddal mindig G65 típusú, tehát nem öröklődő hívás adható meg.

Ha a felhasználói hívásban a makrótörzsben ismételten ugyanarra az M kódra hivatkozunk, a makró nem hívódik újra, hanem az M kód a PLC-nek kerül átadásra.

Ha a makrótörzsben felhasználói G, S, T, A, B, C vagy más felhasználói M hívás történik:

FGMAC=0: nem engedélyezett (közönséges M,S, ...G kódként kerülnek végre hajtásra),

FGMAC=1: engedélyezett, azaz új hívás keletkezik.

A paramétermezőben kijelölt, makróhívást kezdeményező M kódot a mondatban csak a / és az N cím előzheti meg.

M kódra indított makróhívást tartalmazó mondatban csak egy M kód szerepelhet.

1. sz. argumentumkészlete:

A B C D E F G H I J K L P Q R S T U V W X Y Z

M funkcióval a 2. sz. argumentumkészlet is használható.

22.5 Felhasználói alprogramhívás M kódra

A paramétermezőben ki lehet jelölni legfeljebb 10 darab M kódot amire alprogramhívást lehet kezdeményezni. Ekkor az

Nn Gg Xx Zz M98 Pp

utasítás helyett a következő utasítás adható meg:

Nn Gg Xx Zz **Mm**

Ekkor a kijelölt M kód nem kerül a PLC-nek átadásra, hanem a megfelelő alprogram hívódik meg.

A paramétermezőben azt kell beállítani, hogy a hívó M kód melyik programszámot hívja.

M(9000): M kód, amelyik az O9000 sz. programot hívja

M(9001): M kód, amelyik az O9001 sz. programot hívja

:

M(9009): M kód, amelyik az O9009 sz. programot hívja

Ha a felhasználói hívásban az alprogramban ismételten a hívó M kódra hivatkozunk, az alprogram nem hívódik újra, hanem az M kód a PLC-nek kerül átadásra.

Ha az alprogramban felhasználói G, S, T, A, B, C vagy más felhasználói M hívás történik:

FGMAC=0: nem engedélyezett (közönséges M,S, ...G kódként kerülnek végre hajtásra),

FGMAC=1: engedélyezett, azaz új hívás keletkezik.

22.6 Felhasználói alprogramhívás T kódra

A T(9034)=1 paraméter érték mellett a programba írt T érték nem kerül a PLC-nek átadásra, hanem a T kód az O9034 alprogram hívását kezdeményezi.

Ekkor a

Gg Xx Zz Tt

mondat ekvivalens az alábbi két mondattal:

#199=t

Gg Xx Zz M98 P9034

A T címnek adott érték argumentumként átadódik a #199 globális változónak.

Ha a T kódra induló alprogramban újra T címre hivatkozunk, az alprogram nem hívódik újra, hanem a cím értéke már a PLC-nek kerül átadásra.

Ha az alprogramban felhasználói G, M, S, A, B, C hívás történik:

FGMAC=0: nem engedélyezett (közönséges M, S, ...G kódként kerülnek végre hajtásra),

FGMAC=1: engedélyezett, azaz új hívás keletkezik.

22.7 Felhasználói alprogramhívás S kódra

Az S(9033)=1 paraméter érték mellett a programba írt S érték nem kerül a PLC-nek átadásra, hanem az S kód az O9033 alprogram hívását kezdeményezi.

Ekkor a

Gg Xx Zz Ss

mondat ekvivalens az alábbi két mondattal:

#198=s

Gg Xx Zz M98 P9033

Az S címnek adott érték argumentumként átadódik a #198 globális változónak.

Ha az S kódra induló alprogramban újra S címre hivatkozunk, az alprogram nem hívódik újra, hanem a cím értéke már a PLC-nek kerül átadásra.

Ha az alprogramban felhasználói G, M, T, A, B, C hívás történik:

FGMAC=0: nem engedélyezett (közönséges M, S, ...G kódként kerülnek végre hajtásra)

FGMAC=1: engedélyezett, azaz új hívás keletkezik.

22.8 Felhasználói alprogramhívás A, B, C kódra

Ha a paramétermezőben az A(9030)=1, vagy a B(9031)=1, vagy a C(9032)=1 a programba írt A, B, vagy C érték nem kerül a PLC-nek átadásra, hanem az A, B, vagy C kód az O9030, O9031, vagy az O9032 alprogram hívását kezdeményezi. Ekkor pl. a

Gg Xx Zz Cc

mondat ekvivalens az alábbi két mondattal:

#197=c

Gg Xx Zz M98 P9032

Az A címnek adott érték a #195, a B címnek adott érték a #196, a C címnek adott érték pedig a #197 globális változónak adódik át.

Ha az A, B, vagy C kódra induló alprogramban újra ugyanarra a címre hivatkozunk, az alprogram nem hívódik újra, hanem a cím értéke már a PLC-nek kerül átadásra.

Ha az alprogramban felhasználói G, M, S, T, hívás történik:

FGMAC=0: nem engedélyezett (közönséges M,S, ...G kódként kerülnek végre hajtásra,

FGMAC=1: engedélyezett, azaz új hívás keletkezik.

22.9 Az alprogramhívás és makróhívás közti különbség

- Makróhívás tartalmazhat argumentumot, az alprogramhívás nem.
- Az alprogramhívás csak a mondatba programozott egyéb parancsok végrehajtása után ágazik el a hívott alprogramba, a makróhívás csak elágazik.
- A makróhívás megváltoztatja a lokális változók szintjét, az alprogramhívás nem. Például #1 értéke G65 hívás előtt más, mint a makró törzsében. #1 értéke M98 előtt ugyanaz, mint #1 értéke az alprogramban.

22.9.1 Többszörös hívás

Makróból újra hívható másik makró. Makróhívás négyszeres mélységig lehetséges, beleértve az egyszerű és öröklődő makróhívásokat is. Az alprogramhívásokkal együtt a hívások maximális mélysége nyolcszoros lehet.

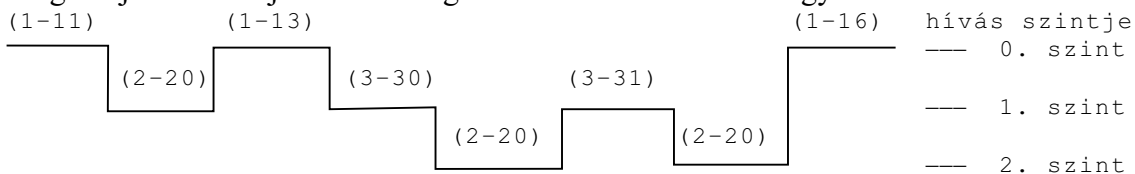
Öröklődő, G66 típusú makrók többszörös hívása esetén minden mozgásmondat végrehajtása után először a később megadott makró hívódik, és ebből hívódnak meg visszafelé haladva az előbb megadott makrók. Lássuk a következő példát:

```
%O0001
...
N10 G66 P2
N11 G1 G91 Z10      (1-11)
N12 G66 P3
N13 Z20             (1-13)
N14 G67             (G66 P3 hívás törlése)
N15 G67             (G66 P2 hívás törlése)
N16 Z-5             (1-16)
...

%O0002
N20 X4              (2-20)
N21 M99
%

%O0003
N30 Z2              (3-30)
N31 Z3              (3-31)
N32 M99
%
```

A végrehajtás sorrendje csak a mozgást tartalmazó mondatok figyelembe vételével:



A zárójelbe tett számok közül az első a végrehajtás alatt álló program száma a második pedig a végrehajtás alatt álló mondat száma.

Az N14 mondatban megadott G67 utasítás az N12 mondatban hívott makrót (O0002) törli, az N15 mondatban megadott az N10 mondatban hívottat (O0003).

G66.1 típusú makrók többszörös hívása esetén először a később megadott makró hívódik minden mondat beolvasásakor argumentumként kezelve ennek a mondatnak a címeit, majd ennek a makrónak a mondatait beolvasva és argumentumként kezelve az eggyel előbb megadott makró hívódik.

Ha makróból újra makrót hívunk a makró szintjével a lokális változók szintje is növekszik.

főprogram	makró	makró	makró	makró
0. szint	1. szint	2. szint	3. szint	4. szint
	O _____	O _____	O _____	O _____
G65 P	G65 P	G65 P	G65 P	
	M99	M99	M99	M99
lokális változók				
0. szint	1. szint	2. szint	3. szint	4. szint
#1	#1	#1	#1	#1
	:	:	:	:
#33	#33	#33	#33	#33

Az első makró hívásakor a főprogram lokális változói #1–től #33–ig eltárolódnak és az 1. szinten a lokális változók a híváskor megadott argumentumértékeket veszik föl. Az első szintről történő újabb makróhívás esetén az első szint lokális változói #1–től #33–ig eltárolódnak, és a második szinten a lokális változók a híváskor megadott argumentum értékeket veszik föl. Többszörös hívás esetén az előző szint lokális változói eltárolódnak és a következő szinten a lokális változók a híváskor megadott argumentumértékeket veszik fel. M99 esetén, amikor visszatér a hívott makróból a hívó programba, az előző szint eltárolt lokális változói ugyanabban az állapotban visszaállításra kerülnek, mint amilyen állapotban a híváskor eltárolódtak.

22.10 A felhasználói makró formátuma

A felhasználói makró programformátuma megegyezik az alprogramokéval:

O(programszám)

:

parancsok

:

M99

A programszám tetszőleges lehet, de az O9000–től O9034–ig terjedő programszámok speciális hívásokra vannak fenntartva.

22.11 A programnyelv változói

A főprogramban, az alprogramokban és a makrókban változókat is lehet adni konkrét számértékek helyett az egyes címeknek. A változóknak érték adható a megengedett értékhatáron belül. Változók használatával a programozás sokkal rugalmasabbá tehető.

Főprogramokban és alprogramokban a globális változók használatával a megfelelő méretadatok parametrizálhatók, így a hasonló, ám különböző méretű darabokhoz nem kell új programot írni, hanem a megfelelő globális változók átírásával áttérhetünk egy újabb méretű darabra.

A változók használatával a makró sokkal rugalmasabbá tehető, mint a hagyományos alprogram. Amíg az alprogramnak nem lehet paramétereiket átadni, addig a makrónak a lokális változókön keresztül argumentumok, vagy paraméterek átadhatók.

22.11.1 Változó azonosítása

Számos változó használható, és minden változót a száma azonosít. Változó-ra való hivatkozást a # jel vezet be, és az utána következő szám jelöli ki a változó számát. Például:

```
#12
#138
#5106
```

Változóra formulával is hivatkozhatunk: #[<formula>]

Például:

```
#[#120] jelentése: a 120-as változóban található az a szám, ahányadik változóra hivatko-
zunk.
#[#120-4] jelentése: a 120-as változóban található számból 4-et levonva kapjuk a hivat-
kozott változó számát.
```

22.11.2 Hivatkozás változóra

A programmondat szavaiban a különböző címek nem csak számértéket vehetnek föl, hanem változók értékeit is. A címek után változóra való hivatkozás esetén is használható a "-" mínusz előjel, illetve az I operátor, ahol ez megengedett számértékek esetén. Például:

```
G#102 ha #102=1.0, akkor ez a hivatkozás G1-gyel egyenértékű
XI-#24 ha #24=135.342, akkor ez a hivatkozás XI-135.342-vel egyenértékű
```

- programszám: O, mondatszám: N, feltételes mondat: / címei után változóra való hivatkozás nem megengedett. N címet akkor tekinti mondatszámnak, ha a mondatban legfeljebb a / cím előzi meg.
- Egy változó számát nem helyettesíthetjük változóval, azaz nem írhatjuk, hogy ##120. A helyes megadás: #[#120]
- Ha cím után használjuk a változót, a változó értéke nem haladhatja meg az adott címre megengedett értéktartományt. Például, ha #112=5630 az M#112 hivatkozás hibajelzést eredményez.
- Ha cím után használjuk a változót, a változó értéke a címnek megfelelő értékes jegyre kerekítődik. Például:

```
#112=1.23 esetén M#112 M1 lesz,
#112=1.6 esetén M#112 M2 lesz.
```

22.11.3 Üres változók

Az olyan változó értéke amelyekre még nem hivatkoztunk üres. A #0 változó értéke mindig üres:
#0=<üres>

22.11.4 A változók számábrázolása

A változókat 32 bit mantisszán és 8 bit karakterisztikán ábrázoljuk:

változó= $M \cdot 2^K$

Az **üres** változó ábrázolása: M=0, K=0

A **0** értékű változó ábrázolása: M=0, K=-128

Az üres változónak a természete összevetve azzal, ha egy változó értéke 0:

Üres változóra való *hivatkozás* címben:

ha #1=<üres>	ha #1=0
G90 X20 Y#1	G90 X20 Y#1
G90 X20	G90 X20 Y0

Üres változó *értékadó* utasításban:

ha #1=<üres>	ha #1=0
#2=#1	#2=#1
#2=<üres>	#2=0
#2=#1*3	#2=#1*3
#2=0	#2=0
#2=#1+#1	#2=#1+#1
#2=0	#2=0

Feltételvizsgálat esetén az **üres** és **0** értékű változó közti különbség:

ha #1=<üres>	ha #1=0
#1 EQ #0	#1 EQ #0
teljesült	nem teljesült
#1 NE 0	#1 NE 0
teljesült	nem teljesült
#1 GE #0	#1 GE #0
teljesült	nem teljesült
#1 GT 0	#1 GT 0
teljesült	nem teljesült

22.12 A változók típusai

A változókat felhasználásuk módja és tulajdonságaik alapján lokális, globális és rendszerváltozókra osztjuk. Azt, hogy egy változó melyik csoportba tartozik a változó száma mondja meg.

22.12.1 Lokális változók: #1 – #33

A lokális változó olyan változó, amelyet a makróprogram adott helyen, lokálisan használ. Ha az A makró hívja B-t és, ha az A makróban is hivatkozunk a #i lokális változóra, meg a B makróban is, az A makró szintjén lévő lokális #i változó értéke nem veszik el és nem íródik át miután a B makró meghívódott, annak ellenére, hogy a B makróban is hivatkozunk #i-re. A lokális változókat argumentumátvitelre használjuk. Az argumentum címe és a lokális változók közötti megfeleltetést az egyszerű makróhívást (G65) ismertető fejezetben közölt táblázat mutatja.

Az a lokális változó, amelynek címe nem szerepelt az argumentumkijelölésben üres és szabadon felhasználható.

22.12.2 Globális változók: #100 - #199, #500 - #599

A globális változók a lokális változókkal ellentétben nem csak a programhívások ugyanazon szintjein azonosak, hanem végig az egész programon keresztül függetlenül, hogy főprogramban, alprogramban, vagy makróban vagyunk, illetve, hogy a makró melyik szintjén. Ha tehát #i-t valamelyik makróban használtuk, annak például értéket adtunk, egy másik makróban #i-nek ugyanaz az értéke minaddig, amíg át nem írjuk. A globális változók a rendszerben teljesen szabad felhasználásúak, nincs semmi kitüntetett szerepük.

- A #100–tól #199–ig terjedő globális változók törölődnek kikapcsolásra és a program elejére történő resetelésre.
- A #500–tól #599–ig terjedő globális változók értéke kikapcsolás után is megőrződik.

A #500–tól #599–ig terjedő makrováltozók írásvédetté tehetők a *WRPROT1* és a *WRPROT2* paraméterek segítségével. A *WRPROT1* paraméterre a védeni kívánt tömb első elemét, a *WRPROT2* paraméterre a védetté nyilvánított tömb utolsó elemét írjuk. Például, ha a #530–tól a #540–ig tartó változókat írásvédetté akarjuk tenni *WRPROT1=530*–ra és *WRPROT2=540*–re kell állítani a paramétereket.

22.12.3 Rendszerváltozók

A rendszerváltozók kötött változók, amelyek a rendszer állapotáról adnak információt.

Interface bemenő jelek: #1000–#1015, #1032

A #1000–tól #1015–ig terjedő rendszerváltozókon 16 db. interface bemenő jel kérdezhető le egyenként:

rendszerváltozó neve	interface bemenet a PLC program értelmezése szerint
#1000	I [CONST+000]
#1001	I [CONST+001]
#1002	I [CONST+002]
#1003	I [CONST+003]
#1004	I [CONST+004]
#1005	I [CONST+005]
#1006	I [CONST+006]
#1007	I [CONST+007]
#1008	I [CONST+010]
#1009	I [CONST+011]
#1010	I [CONST+012]
#1011	I [CONST+013]
#1012	I [CONST+014]
#1013	I [CONST+015]
#1014	I [CONST+016]
#1015	I [CONST+017]

ahol $CONST = I_LINE * 10$ és *I_LINE* egy paraméter. Így két tetszőleges interface bemenőbyte kérdezhető le.

A fenti változók értéke;

- 0: ha a bemeneten az érintkező nyitott,
- 1: ha a bemeneten az érintkező zárt.

A #1032 változón a fenti 16 bemenet egyszerre is lekérdezhető. A lekérdezett érték az egyenkénti lekérdezéshez rendelt rendszerváltozók függvényében:

$$\#1032 = \sum_{i=0}^{15} \#[1000+i] * 2^i$$

Tehát ha a #1002 és a #1010 bemenetekre van 24 V kapcsolva, a többi bemenet nyitott, a #1032 változó értéke:

$$\#1032 = 1 * 2^2 + 1 * 2^{10} = 1028$$

Az interface bemenetek változói csak olvashatók, és nem szerepelhetnek egy aritmetikai kifejezés bal oldalán.

Interface kimenő jelek: #1100–#1115, #1132

A #1100–tól #1115–ig terjedő rendszerváltozókon 16 db. interface kimenő jel adható ki egyenként:

rendszerváltozó neve	interface kimenet a PLC program értelmezése szerint
#1100	Y[CONST+000]
#1101	Y[CONST+001]
#1102	Y[CONST+002]
#1103	Y[CONST+003]
#1104	Y[CONST+004]
#1105	Y[CONST+005]
#1106	Y[CONST+006]
#1107	Y[CONST+007]
#1108	Y[CONST+010]
#1109	Y[CONST+011]
#1110	Y[CONST+012]
#1111	Y[CONST+013]
#1112	Y[CONST+014]
#1113	Y[CONST+015]
#1114	Y[CONST+016]
#1115	Y[CONST+017]

ahol $CONST = O_LINE * 10$ és O_LINE egy paraméter. Így két tetszőleges interface kimenőbyte kérdezhető le, vagy két tetszőleges kimenő byte-ba lehet írni.

A fenti változók értéke;

- 0: a kimeneten az érintkező nyitott,
- 1: a kimeneten az érintkező zárt.

A #1132 változón a fenti 16 kimenet egyszerre is kiadható. A kiadott érték az egyenkénti kiadás-hoz rendelt rendszerváltozók függvényében:

$$\#1132 = \sum_{i=0}^{15} \#[1100+i] * 2^i$$

Tehát ha a #1102 és a #1109 kimeneteket kapcsoljuk be, és a többi kimenet nyitott, a #1132 változón a következő értéket kell kiadni:

$$\#1132 = 1 * 2^2 + 1 * 2^9 = 516$$

Szerszámkorrekciós értékek: #10001–#19999

A szerszámkorrekciós értékeket ki lehet olvasni a #10001–#19999 változókon, illetve értéket lehet nekik adni.

N	X		Y		Z		R		Q
	kopás	geom..	kopás	geom..	kopás	geom..	kopás	geom..	
1	#10001	#15001	#14001	#19001	#11001	#16001	#12001	#17001	#13001
2	#10002	#15002	#14002	#19002	#11002	#16002	#12002	#17002	#13002
...
99	#10099	#15099	#14099	#19099	#11099	#16099	#12099	#17099	#13099

Munkadarab nullponteltolások: #5201–#5328

A munkadarab nullpont eltolások értékeit ki lehet olvasni a #5201–#5326 változókon, illetve értéket lehet nekik adni.

változó száma	értéke	munkadarab koordinátarendszer	
#5201	közös munkadarab nullponteltolás 1. tengely	összes munkadarab koordinátarendszerben érvényes	
#5202	közös munkadarab nullponteltolás 2. tengely		
:			
#5208	közös munkadarab nullponteltolás 8. tengely		
#5221	munkadarab nullponteltolás 1. tengely		G54
#5222	munkadarab nullponteltolás 2. tengely		
:			
#5228	munkadarab nullponteltolás 8. tengely		
#5241	munkadarab nullponteltolás 1. tengely	G55	
#5242	munkadarab nullponteltolás 2. tengely		
:			
#5248	munkadarab nullponteltolás 8. tengely		
#5261	munkadarab nullponteltolás 1. tengely	G56	
#5262	munkadarab nullponteltolás 2. tengely		
:			
#5268	munkadarab nullponteltolás 8. tengely		
#5281	munkadarab nullponteltolás 1. tengely	G57	
#5282	munkadarab nullponteltolás 2. tengely		
:			
#5288	munkadarab nullponteltolás 8. tengely		
#5301	munkadarab nullponteltolás 1. tengely	G58	
#5302	munkadarab nullponteltolás 2. tengely		
:			
#5308	munkadarab nullponteltolás 8. tengely		
#5321	munkadarab nullponteltolás 1. tengely	G59	
#5322	munkadarab nullponteltolás 2. tengely		
:			
#5328	munkadarab nullponteltolás 8. tengely		

A tengelyek számozása a fizikai tengelyszámot jelenti. Azt, hogy a tengelyszám és a tengelyek elnevezése között mi az összefüggés a gép építője határozza meg az *AXIS1* – *AXIS6* paramétercsoportokon. Általában az 1. tengely az X, a 2. tengely az Z, a 3. tengely a C címhez van rendelve, de ettől eltérő megadások is lehetségesek.

Hibajelzés: #3000

A

#3000=nnn(HIBAJELZÉS)

értékadással számmal jelzett (nnn: max. három számjegy), és/vagy szöveges hibaüzenetet adhatunk. A szöveget (,) gömbölyű zárójelek közé kell tenni. Az üzenet hossza legfeljebb 25 karakter lehet.

Ha a makróban hibát észlel a program, azaz olyan ágra fut, ahol a #3000 változónak értéket adtunk, az előző mondatig végrehajtja a programot, majd a végrehajtást felfüggeszti és a képernyőn megjelenik a zárójelek között megadott hibaüzenet, illetve az üzenet kódja 4nnn formában,

vagyis a #3000 értéken megadott nnn számhoz hozzáad 4000-et. Ha számot nem adtunk az üzenet kódja 4000 lesz, ha szöveget nem adtunk, csak a kód jelenik meg. A hibaállapot a RESET gombbal szüntethető meg.

Milliszekundumos időmérő: #3001

A #3001 változó értéke írható és olvasható.

Két időpont között eltelt időt mérhetjük meg milliszekundumos mértékegységben, kb. 20 ms pontossággal. A #3001 számláló 65536-nál túlszordul. A #3001 változó értéke bekapcsoláskor nulláról indul, és felfele számol. Mindig számol, amikor a vezérlés be van kapcsolva.

Forgácsolási (fő)időmérő: #3002

A #3002 változó értéke írható és olvasható.

Két időpont között eltelt időt mérhetjük meg perc mértékegységben, kb. 20 ms pontossággal.

A #3002 változó értéke bekapcsoláskor a kikapcsoláskor érvényes értékkel indul, és felfele számol.

Akkor számol, amikor a START lámpa ég, azaz a vezérlés start állapotában méri az időt. A paramétertár *CUTTING2* nevű időmérőjén található.

Mondatonkénti végrehajtás elnyomása: #3003

Ha a #3003=1 a mondatonkénti végrehajtás állapotában mindaddig nem áll meg egy mondat végrehajtása után, amíg ennek a változónak az értéke 0 nem lesz.

Bekapcsolásra, program elejére történő resetelésre a változó értéke 0.

#3003 mondatonkénti végrehajtás

0 nincs elnyomva

1 el van nyomva

Előtolás stop, előtolás override, pontos megállás elnyomása: #3004

Az előtolás stop funkció elnyomása esetén a stop gomb megnyomása után akkor áll meg az előtolás amikor az elnyomás oldásra kerül.

Az előtolás override elnyomásakor az override értékét 100%-nak veszi, amíg az elnyomás nem kerül oldásra.

A pontos megállás elnyomásakor nem végez vizsgálatot addig a vezérlő, amíg az elnyomás nem került oldásra.

Bekapcsolásra, program elejére történő resetelésre a változó értéke 0.

#3004	pontos megállás	előtolás override	előtolás stop
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

0: a funkció hatásos

1: a funkció el van nyomva

Megállás üzenettel: #3006

A

#3006=nnn(ÜZENET)

értékkadás hatására a program végrehajtása megáll, és a gömbölyű zárójelek közé zárt üzenet megjelenik a képernyőn, illetve az üzenet kódja 5nnn formában, vagyis a #3006 értéken megadott nnn számhoz hozzáad 5000-et. Ha számot nem adtunk az üzenet kódja 5000 lesz, ha szöveget nem adtunk, csak a kód jelenik meg. A program végrehajtása a START gomb lenyomására folytatódik, ekkor az üzenet letörlődik a képernyőről. Az üzenet hossza legfeljebb 25 karakter lehet. Az utasítás hasznosan alkalmazható olyan esetben, amikor a program végrehajtása közben kezelői beavatkozás szükséges.

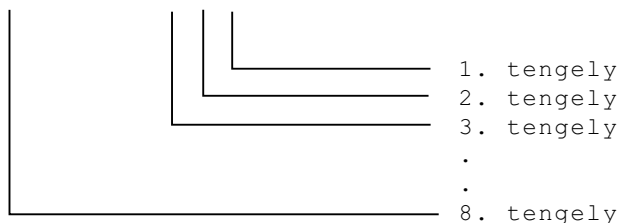
Tükrözési állapot: #3007

A #3007 változó olvasásával megállapítható, hogy melyik fizikai tengelyen van érvényes tükrözési parancs bejegyezve. A változó csak olvasható.

A változó értékét binárisan értelmezve:

1 1 1 1 1 1

5 4 3 2 1 0 9 8 7 6 5 4 3 2 1 0



Az egyes bitek jelentése:

0: nincs tükrözés

1: a tükrözés be van kapcsolva

Ha a változó értéke például 5, az 1. és a 3. tengelyen van a tükrözés bekapcsolva. A tengelyszám fizikai tengelyszámot jelent, paraméter határozza meg, hogy melyik fizikai tengelyszámhoz milyen nevű tengely tartozik.

Megmunkált darabok száma, elkészítendő munkadarabok száma: #3901, #3902

A megmunkált darabok számát a #3901-es számlálóban gyűjti a vezérlő. A számláló tartalmát minden M02, M30 vagy a *PRTCNTM* paraméterben kijelölt M funkció végrehajtásakor lépteti 1-gyel a vezérlő. Amikor a megmunkált darabok száma elérte a szükséges munkadarabszámot (#3902-es számláló) a végrehajtó egy flagen értesíti a PLC-t.

megmunkált darabok száma #3901

elkészítendő darabok száma #3902

A #3901 számláló a paramétertár *PRTTOTAL*, a #3902 számláló a paramétertár *PRTREQRD* paraméterén található.

Öröklődő információk: #4001–#4130, #4201–#4330

A megelőző mondatban érvényes öröklődő parancsok a #4001–#4130–as rendszerváltozók olvasásával állapíthatók meg.

A pillanatnyilag végrehajtás alatt álló mondatban érvényes öröklődő parancsok a #4201–4330–as változók olvasásával állapíthatók meg.

rendszer változó	előző mondat öröklődő információi	rendszer változó	végrehajtás alatt álló mondat öröklődő információi
#4001	G kód 1-es csoport	#4201	G kód 1-es csoport
:	:	:	:
#4020	G kód 20-as csoport	#4220	G kód 20-as csoport
#4101	A kód	#4301	A kód
#4102	B kód	#4302	B kód
#4103	C kód	#4303	C kód
#4107	D kód	#4307	D kód
#4108	E kód	#4308	E kód
#4109	F kód	#4309	F kód
#4111	H kód	#4311	H kód
#4113	elsőként beolvasott M kód	#4313	elsőként beolvasott M kód
#4114	mondatszám: N	#4314	mondatszám: N
#4115	programszám: O	#4315	programszám: O
#4119	S kód	#4319	S kód
#4120	T kód	#4320	T kód
#4150	2. M kód csoport: M11, M12, ..., M18	#4350	2. M kód csoport: M11, M12, ..., M18
#4151	3. M kód csoport: M03, M04, M05, M19	#4351	3. M kód csoport: M03, M04, M05, M19
#4152	4. M kód csoport: M07, M08, M09	#4352	4. M kód csoport: M07, M08, M09

Pozícióinformációk: #5001 – #5108**Mondatvégi pozíciók**

rendszer változó	pozíció információ jellege	beolvasás mozgás közben
#5001	1. tengely mondatvégi pozíciója	
#5002	2. tengely mondatvégi pozíciója	
:	:	lehetséges
#5008	8. tengely mondatvégi pozíciója	

A mondatvégi pozíció

- az aktuális munkadarab koordinátarendszerben,
- a koordinátaeltolások figyelembe vételével,
- derékszögű koordinátákban,
- az összes korrekció (hossz, sugár, szerszámeltolás) figyelmen kívül hagyásával kerül a változóba.

Pillanatnyi pozíciók a gép koordinátarendszerében

rendszer változó	pozíció információ jellege	beolvasás mozgás közben
#5021	1. tengely pillanatnyi pozíciója (G53)	
#5022	2. tengely pillanatnyi pozíciója (G53)	
:		
#5028	8. tengely pillanatnyi pozíciója (G53)	nem lehetséges

A pillanatnyi pozíció (G53)

- a gép koordinátarendszerében,
- a hossz korrekció figyelembe vételével kerül a változóba.

Pillanatnyi pozíciók a munkadarab koordinátarendszerében

rendszer változó	pozíció információ jellege	beolvasás mozgás közben
#5041	1. tengely pillanatnyi pozíciója	
#5042	2. tengely pillanatnyi pozíciója	
:		
#5048	8. tengely pillanatnyi pozíciója	nem lehetséges

A pillanatnyi pozíció

- az aktuális munkadarab koordinátarendszerben,
- a koordinátaeltolások figyelembe vételével,
- derékszögű koordinátákban,
- az összes korrekció (hossz, sugár, szerszámeltolás) figyelembe vételével kerül a változóba.

Tapintási pozíciók

rendszer változó	pozíció információ jellege	beolvasás mozgás közben
#5061	1. tengely tapintási pozíciója (G31)	
#5062	2. tengely tapintási pozíciója (G31)	
:		
#5068	8. tengely tapintási pozíciója (G31)	lehetséges

G31 mondatban az a pozíció, ahol a tapintó jele megjött

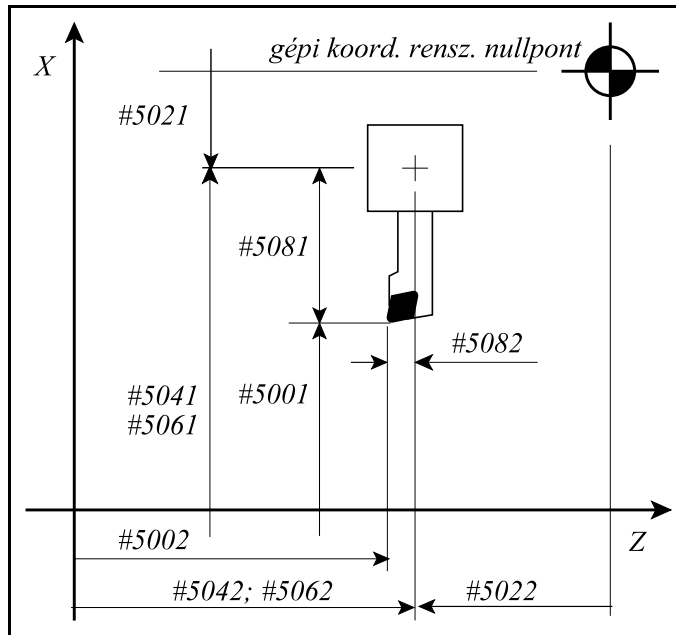
- az aktuális munkadarab koordinátarendszerben,
- a koordinátaeltolások figyelembe vételével,
- derékszögű koordinátákban,
- az összes korrekció (hossz, sugár, szerszámeltolás) figyelembe vételével kerül a változóba.

Ha a tapintó jele nem jött meg a fenti változók a G31 mondatban programozott végponti pozíciót veszik fel.

Szerszám–hosszkorrekciók

rendszer változó	pozíció információ jellege	beolvasás mozgás közben
#5081	1. tengelyen bejegyzett hosszkorrekció	nem lehetséges
#5082	2. tengelyen bejegyzett hosszkorrekció	
:		
#5088	8. tengelyen bejegyzett hosszkorrekció	

A kiolvasható szerszám–hosszkorrekció a végrehajtás alatt álló mondatban érvényes hosszkorrekció.



22.12.3-1 ábra

Szervo lemaradás

rendszer változó	pozíció információ jellege	beolvasás mozgás közben
#5101	1. tengelyen a szervo lemaradás értéke	nem lehetséges
#5102	2. tengelyen a szervo lemaradás értéke	
:		
#5108	8. tengelyen a szervo lemaradás értéke	

A kiolvasható lemaradás előjelhelyes érték, milliméterben megadva.

22.13 A programnyelv utasításai

A különböző utasítások leírásánál a

$\#i = \langle \text{formula} \rangle$

kifejezést használjuk. A $\langle \text{formula} \rangle$ tartalmazhat aritmetikai műveleteket, függvényeket, változókat, konstansokat.

Általánosságban a $\langle \text{formula} \rangle$ -ban a $\#j$ és a $\#k$ változókra hivatkozunk.

A $\langle \text{formula} \rangle$ nem csak az értékadó utasítás jobb oldalán állhat, hanem az NC mondatban a különböző címek is felvehetnek konkrét számérték vagy változó helyett formulát is.

22.13.1 Az értékadó utasítás: $\#i = \#j$

Az utasítás kódja: =

Az utasítás hatására a $\#i$ változó a $\#j$ változó értékét veszi fel, vagyis $\#i$ változóba bekerül $\#j$ változó értéke.

22.13.2 Aritmetikai műveletek és függvények

Egy operandusú műveletek

Egy operandusú mínusz: $\#i = - \#j$

A művelet kódja: –

A művelet hatására a $\#i$ változó a $\#j$ változóval abszolút értékben megegyező nagyságú, de ellentétes előjelű lesz.

Aritmetikai negáció: $\#i = \text{NOT } \#j$

A művelet kódja: **NOT**

A művelet hatására a $\#j$ változót először átalakítja 32 bites fixpontos számmá. Ha az így átalakított szám 32 biten nem ábrázolható **3091 HIBÁS MŰVELET #VAL** hibajelzést ad. Ezután ennek a fixpontos számnak a bitenkénti negált értékét veszi mind a 32 bitre, majd az így keletkezett számot visszaalakítja lebegőpontosá és beteszi a $\#i$ változóba.

Additív aritmetikai műveletek

Összeadás: $\#i = \#j + \#k$

A művelet kódja: +

A művelet hatására a $\#i$ változó a $\#j$ és $\#k$ változók értékének összegét veszi fel.

Kivonás: $\#i = \#j - \#k$

Az művelet kódja: –

A művelet hatására a $\#i$ változó a $\#j$ és $\#k$ változók értékének különbségét veszi fel.

Aritmetikai vagy: $\#i = \#j \text{ OR } \#k$

A művelet kódja: **OR**

A művelet hatására a $\#i$ változóba a $\#j$ és a $\#k$ változók bitenkénti értékeinek logikai összege kerül, mind a 32 bitre. Ahol a két szám azonos helyiértékén mindkét helyen 0 található, arra a helyiértékre az eredményben 0 kerül, egyébként 1.

Aritmetikai kizáró vagy: $\#i = \#j \text{ XOR } \#k$ A művelet kódja: **XOR**

A művelet hatására a $\#i$ változóba a $\#j$ és a $\#k$ változók bitenkénti értékei úgy összegződnek, hogy, ahol azonos helyiértéken azonos számértékek találhatók az eredményben arra a helyiértékre 0, ahol különböző számértékek találhatók oda 1 kerül, mind a 32 biten.

Multiplikatív aritmetikai műveletek**Szorzás:** $\#i = \#j * \#k$ A művelet kódja: *****

A művelet hatására a $\#i$ változó a $\#j$ és $\#k$ változók értékének szorzatát veszi fel.

Osztás: $\#i = \#j / \#k$ A művelet kódja: **/**

A művelet hatására a $\#i$ változó a $\#j$ és $\#k$ változók értékének hányadosát veszi fel. $\#k$ értéke nem lehet 0. Ellenkező esetben *3092 0-VAL OSZTÁS #* hibajelzést ad a vezérlő.

Modulusképzés: $\#i = \#j \text{ MOD } \#k$ A művelet kódja: **MOD**

A művelet hatására a $\#i$ változó a $\#j$ és $\#k$ változók osztási maradékát veszi fel. $\#k$ értéke nem lehet 0. Ellenkező esetben *3092 0-VAL OSZTÁS #* hibajelzést ad a vezérlő.

Példa: $\#120 = 27 \text{ MOD } 4$ esetén a $\#120$ változó értéke 3 lesz.

Aritmetikai és: $\#i = \#j \text{ AND } \#k$ A művelet kódja: **AND**

A művelet hatására a $\#i$ változóba a $\#j$ és a $\#k$ változók bitenkénti értékeinek logikai szorzata kerül, mind a 32 bitre. Ahol a két szám azonos helyiértékén mindkét helyen 1 található, arra a helyiértékre az eredményben 1 kerül, egyébként 0.

Függvények**Négyzetgyökvonás:** $\#i = \text{SQRT } \#j$ A függvény kódja: **SQRT**

A művelet hatására a $\#i$ változó a $\#j$ változó négyzetgyökét veszi fel. A $\#j$ változó értéke nem lehet negatív szám.

Színusz: $\#i = \text{SIN } \#j$ A függvény kódja: **SIN**

A művelet hatására $\#i$ változó $\#j$ változó szinuszát veszi fel. $\#j$ értéke mindig fokban értendő.

Koszínusz: $\#i = \text{COS } \#j$ A függvény kódja: **COS**

A művelet hatására $\#i$ változó $\#j$ változó koszínuszát veszi fel. $\#j$ értéke mindig fokban értendő.

Tangens: $\#i = \text{TAN } \#j$ A függvény kódja: **TAN**

A művelet hatására $\#i$ változó $\#j$ változó tangensét veszi fel. $\#j$ értéke mindig fokban értendő. $\#j$ értéke nem lehet $(2n+1)*90^\circ$, ahol $n=0, \pm 1, \pm 2, \dots$

Arkuszszinusz: #i = ASIN #j

A függvény kódja: **ASIN**

A művelet hatására #i változó #j változó arkuszszinuszát veszi fel. $-1 \leq \#j \leq 1$ feltételnek teljesülnie kell. Az eredmény, azaz #i értéke $+90^\circ$ és -90° közé esik.

Arkuszkoszinusz: #i = ACOS #j

A függvény kódja: **ACOS**

A művelet hatására #i változó #j változó arkuszkoszinuszát veszi fel. $-1 \leq \#j \leq 1$ feltételnek teljesülnie kell. Az eredmény, azaz #i értéke 0° és 180° közé esik.

Arkusztangens: #i = ATAN #j

A függvény kódja: **ATAN**

A művelet hatására #i változó #j változó arkusztangensét veszi fel. Az eredmény, azaz #i értéke $+90^\circ$ és -90° közé esik.

Exponenciális: #i = EXP #j

A függvény kódja: **EXP**

A művelet hatására a #i változó a természetes szám (e) #j-edik hatványát veszi fel.

Természetes alapú logaritmus: #i = LN #j

A függvény kódja: **LN**

A művelet hatására #i változó a #j szám természetes alapú logaritmusát veszi fel. #j értéke nem lehet 0, vagy negatív szám.

Abszolút érték képzés: #i = ABS #j

A függvény kódja: **ABS**

A művelet hatására a #i változó a #j változó abszolút értékét veszi fel.

Bináris szám binárisan kódolt decimálissá átalakítása: #i = BCD #j

A függvény kódja: **BCD**

A művelet hatására a #i változó a #j változó BCD értékét veszi fel. #j változó értéktartománya: 0 – 99999999.

Binárisan kódolt decimális szám binárisra alakítása: #i = BIN #j

A függvény kódja: **BIN**

A művelet hatására a #i változó a #j változó bináris értékét veszi fel. #j változó értéktartománya: 0 – 99999999. A #j változóban az egyes tetrádokon lévő számjegyek nagysága nem lehet 9-nél nagyobb.

Kerekítés abszolút értékben lefelé: #i = FIX #j

A függvény kódja: **FIX**

A művelet a #j változó törtrészét eldobja és ez az érték kerül #i változóba.

Például:

```
#130 = FIX 4.8 = 4
#131 = FIX -6.7 = -6
```

Kerekítés abszolút értékben felfelé: #i = FUP #j

A függvény kódja: **FUP**

A művelet a #j változó törtrészét eldobja és abszolút értékben 1-et ad hozzá.

Például:

```
#130 = FUP 12.1 = 13
#131 = FUP -7.3 = -8
```

Összetett aritmetikai műveletek végrehajtási sorrendje

A fentebb felsorolt aritmetikai műveleteket és függvényeket kombinálni lehet. A műveletek végrehajtásának sorrendje, vagy precedencia szabálya:

függvény – multiplikatív aritmetikai műveletek – additív aritmetikai műveletek.

Például:

$$\#110 = \#111 + \#112 * \underset{\text{1}}{\text{COS}} \#113$$

$\underset{\text{2}}{\text{—————}} \quad \underset{\text{3}}{\text{—————}} \quad \text{a műveletek sorrendje}$

A műveletek végrehajtási sorrendjének módosítása

[és] zárójelek használatával a műveletek végrehajtási sorrendje módosítható. A zárójelezés ötszörös mélységig megengedett. Ha ennél nagyobb mélység van a programban a vezérlés **3064 HIBÁS MAKRÓ KIFEJEZÉS** hibajelzést ad.

Példa háromszoros mélységű zárójelezésre:

$$\#120 = \text{COS} [[[\#121 - \#122] * \#123 + \#125] * \#126]$$

$\underset{\text{1}}{\text{—————}} \quad \underset{\text{2}}{\text{—————}} \quad \underset{\text{3}}{\text{—————}} \quad \underset{\text{4}}{\text{—————}} \quad \underset{\text{5}}{\text{—————}}$

A számok a műveletek végrehajtási sorrendjét mutatják. Látható, hogy az azonos szintű zárójeleken belül a műveletek végrehajtási sorrendjére a fent említett precedenciaszabály érvényes.

22.13.3 Feltételes kifejezések

A programnyelv a következő feltételes kifejezéseket ismeri:

egyenlő:	#i EQ #j
nem egyenlő:	#i NE #j
nagyobb, mint:	#i GT #j
kisebb, mint:	#i LT #j
nagyobb, vagy egyenlő:	#i GE #j
kisebb, vagy egyenlő:	#i LE #j

A feltételes kifejezés mindkét oldalán a változó helyettesíthető formulával is. A fenti feltételes kifejezések IF, vagy WHILE feltételvizsgáló utasítások után állhatnak.

☞ *Megjegyzés:* Mivel a fenti feltételes kifejezések mögött összeadás és kivonás áll, ezért a döntés pontosságánál vegyük figyelembe a lehetséges hibákat!

22.13.4 Feltétel nélküli elágazás: GOTO_n (MENJ)

A **GOTO_n** utasítás hatására a program végrehajtása feltétel nélkül ugyanannak a programnak az n számú mondatán folytatódik. n szám helyettesíthető változóval, vagy formulával is. A mondat-számnak, amelyikre a GOTO utasítással ugrunk a mondat elején kell szerepelni. Ha a kijelölt mondat-számot nem találja **3070 NEM LÉTEZŐ MONDATSZÁM** hibajelzést ad.

22.13.5 Feltételes elágazás: IF[<feltételes kifejezés>] GOTO_n (HA[<.>]MENJ)

Ha a kötelezően szögletes zárójelek közé tett [<feltételes kifejezés>] teljesül, a program végrehajtása ugyanannak a programnak az **n** számú mondatán folytatódik.

Ha a [<feltételes kifejezés>] nem teljesül, a program végrehajtása a következő mondaton folytatódik.

Ha az IF után nem feltételvizsgálat következik, *3091 HIBÁS MŰVELET #-VAL* hibajelzést ad.

Ha a feltételvizsgálatban szintaktikai hiba van *3064 HIBÁS MAKRÓ KIFEJEZÉS* hibát jelez.

22.13.6 Feltételes utasítás: IF[<feltételes kifejezés>] (THEN)utasítás (HA[<.>]AKKOR)

Ha a [<feltételes kifejezés>] teljesül, a THEN mögötti utasítás végrehajtásra kerül.

Ha a [<feltételes kifejezés>] nem teljesül, a program végrehajtása a következő mondaton folytatódik.

Az utasításban a THEN elhagyható az

IF[<feltételes kifejezés>] utasítás

utasítássor végrehajtása ugyanaz.

22.13.7 Ciklusszervezés: WHILE[<feltételes kifejezés>] DO_m (AMÍG[<.>]TEDD) ... END_m (VÉGE)

Amíg a [<feltételes kifejezés>] teljesül a DO *m* utáni mondatok az END *m* mondatig ismételtén végrehajtásra kerülnek. Vagyis a vezérlés megvizsgálja, hogy a feltétel teljesült-e, ha igen végrehajtja a DO *m* és END *m* közötti programrészt, majd END *m* utasítás hatására a program visszatér a WHILE utáni feltétel újbóli vizsgálatára.

Ha a [<feltételes kifejezés>] nem teljesül a program végrehajtása az END *m* utáni mondaton folytatódik.

Ha a WHILE [<feltételes kifejezés>] elhagyásra kerül, azaz a ciklust a DO *m* ... END *m* utasítások írják le a DO *m* és END *m* közötti programrészt végtelen ideig hajtja végre.

m lehetséges értékei: 1, 2, 3. Ettől eltérő értékek megadása *3091 HIBÁS MŰVELET #-VAL* hibajelzést eredményez. Ha a WHILE után nem feltételvizsgálat következik, *3091 HIBÁS MŰVELET #-VAL* hibajelzést ad. Ha a feltételvizsgálatban szintaktikai hiba van *3064 HIBÁS MAKRÓ KIFEJEZÉS* hibát jelez.

A ciklusszervezés szabályai:

– DO *m* utasítást END *m* utasítás előtt kell megadni:

```

:
END1
:
:           HIBÁS
:
DO1

```

– A DO *m* és END *m* utasításoknak párban kell állniuk:

```

:
DO1
:
DO1           HIBÁS
:
END1
:

```

vagy

```

:
DO1
:
END1          HIBÁS
:
END1
:

```

– Ugyanazt az azonosító számot többször lehet használni:

```

:
DO1
:
END1
:
:          HELYES
:
DO1
:
END1
:

```

– DO_m ... END_m párok háromszoros mélységig skatulyázhatók egymásba:

```

:
DO1
:
DO2
:
DO3
:
:          HELYES
:
END3
:
END2
:
END1
:

```

– A DO_m ... END_m párok nem lapolhatják át egymást:

```

:
DO1
:
DO2
:
:          HIBÁS
:
END1
:
END2

```

– Ciklus belsejéből a cikluson kívülre el lehet ágni:

```

:
DO1
:
GOTO150
:
:          HELYES
:
END1
:
N150
:

```


– Ciklusba kívülről nem lehet belépni:

```

:
GOTO150
:
DO1
:
:           HIBÁS
:
N150
:
END1
:

```

vagy

```

:
DO1
:
N150
:
:           HIBÁS
:
END1
:
GOTO150
:

```

– Ciklus belsejéből alprogram, vagy makrohívás lehetséges. Az alprogram, vagy a felhasználói makró belsejében a ciklusok ismét háromszoros mélységig skatulyázhatók egymásba:

```

:
DO1
:
M98...           HELYES
:
G65...           HELYES
:
G66...           HELYES
:
G67...           HELYES
:
END1
:

```

22.13.8 Adatkiadási parancsok

A vezérlés a következő adatkiadási parancsokat ismeri:

POPEN	periféria megnyitása
BPRNT	bináris adatkiadás
DPRNT	decimális adatkiadás
PCLOS	periféria zárása

Ezeket az adatkiadó parancsokat karaktereknek és változók értékeinek a kiadására lehet használni. A kiadás történhet a vezérlés memóriájába és a soros csatornán keresztül egy külső adattároló eszközre.

Periféria megnyitása: POPENn

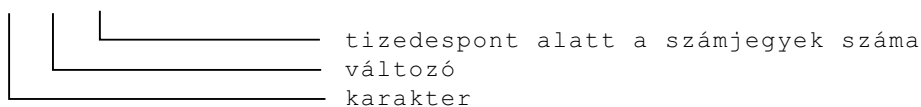
Mielőtt adatkiadó parancsot adunk, meg kell nyitni a megfelelő perifériát, amin keresztül az adatkiadás történni fog. A megfelelő periféria kiválasztása az n számmal történik:

n = 1	soros csatorna, RS-232C interface
n = 31	a vezérlés memóriája

A periféria megnyitásakor egy % karakter is kiküldésre kerül a perifériára, tehát minden adatkiadás egy % karakterrel kezdődik.

Bináris adatkiadás: BPRNT[...]

BPRNT[a #b [c] ...]



A karaktereket ISO, vagy ASCII kódban (paraméterállástól függően), a változókat pedig binárisan küldi ki a parancs.

- A karakterek ISO, vagy ASCII kódban kerülnek kiadásra. A kiküldhető karakterek:
 - alfabetikus karakterek: A, B, ..., Z
 - numerikus karakterek: 1, 2, ..., 0
 - speciális karakterek: *, /, +, –

A * karakter helyett a szóköz (space) ISO kódját (A0h) küldi ki a vezérlés.

- A változók értékeit 4 byte-on, azaz 32 biten adja ki a vezérlés, a legnagyobb helyiértékű byte-tól kezdve. A változók száma után zárójelben [] a tizedespont utáni számjegyek számát kell megadni. Ekkor a vezérlés a változó lebegőpontos értékét olyan fixpontos értéké alakítja, amelyben az értékes tizedesjegyek száma a [] zárójelben megadott érték. c lehetséges értékei: 1, 2, ..., 8. Például:

ha #120 = 258.647673 és [3] ——— kiadásra kerül 258648=0003F258h

- Az üres változót 00000000h bináris kóddal adja ki.
- Az adatkiadás végén a vezérlés automatikusan egy soremelés (LineFeed) karaktert ad ki.

Példa:

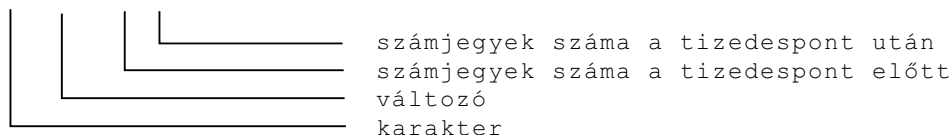
```
BPRNT [ C*/ X#110 [3] Y#120 [3] M#112 [0] ]
#110=318.49362      _____  318494=0004DC1Eh
#120=0.723415      _____  723=000002D3h
#112=23.9          _____  24=00000018h
```

A kiadásra kerülő karakterek:

7	6	5	4	3	2	1	0	
1	1	0	0	0	0	1	1	--- C
1	0	1	0	0	0	0	0	--- Szóköz (Space)
1	0	1	0	1	1	1	1	--- /
1	1	0	1	1	0	0	0	--- X
0	0	0	0	0	0	0	0	--- 00
0	0	0	0	0	1	0	0	--- 04
1	1	0	1	1	1	0	0	--- DC
0	0	0	1	1	1	1	0	--- 1E
0	1	0	1	1	0	0	1	--- Y
0	0	0	0	0	0	0	0	--- 00
0	0	0	0	0	0	0	0	--- 00
0	0	0	0	0	0	1	0	--- 02
1	1	0	1	0	0	1	1	--- D3
0	1	0	0	1	1	0	1	--- M
0	0	0	0	0	0	0	0	--- 00
0	0	0	0	0	0	0	0	--- 00
0	0	0	0	0	0	0	0	--- 00
0	0	0	1	1	0	0	0	--- 18
0	0	0	0	1	0	1	0	--- Soremelés (Line Feed)

Decimális adatkiadás: DPRNT[...]

DPRNT[a #b [c d] ...]



Az összes karakter és számjegy (paraméterállástól függően) ISO, vagy ASCII kódban kerül kiadásra.

- A karakterek kiadására vonatkozó szabályokat lásd **BPRNT** utasítás.
- Változók értékeinek kiadásához meg kell adni, hogy a változó hány decimális egész és tört jegyen kerüljön kiadásra. A számjegyek megadását [] zárójelek közé kell tenni. A számjegyek megadására a $0 < c + d < 9$ feltételnek teljesülni kell. A számok kiadása a legmagasabb helyiértékűtől kezdődik. A számjegyek kiadásánál a negatív előjel (–) és a tizedespont (.) is ISO kódjukkal kerülnek kiadásra. Ha a *PRNT* paraméter *PRNT=0* a + előjel és a vezető nullák helyén szóköz (space) kód kerül kiadásra, a tizedespont után, ha van, minden nulla a 0 kóddal kerül kiadásra. Ha a paraméter *PRNT=1* a + előjel és a vezető nullák nem kerülnek kiadásra, ha tizedespont definiálva van az utána következő nullák kiadásra kerülnek, ha tizedespont nincs definiálva sem tizedespontot sem nullát nem ad ki.
- Ha *d=0*, akkor a tizedespont kiadásra kerül, míg csak *c*-t megadva a tizedespontot sem adja ki a vezérlés.
- Az üres változót 0 kóddal adja ki.
- Az adatkiadás végén a vezérlés automatikusan egy soremelés (LF) karaktert ad ki.

Példa:

```
DPRNT [ X#130 [53] Y#500 [53] T#10 [2] ]
      #130=35.897421 _____ 35.897
      #500=-150.8 _____ -150.8
      #10=214.8 _____ 15
```

Adatkiadás PRNT=0 állásnál:

```
 7 6 5 4 3 2 1 0
  -----
1 1 0 1 1 0 0 0 --- X
1 0 1 0 0 0 0 0 --- Szóköz (space)
1 0 1 0 0 0 0 0 --- Szóköz (space)
1 0 1 0 0 0 0 0 --- Szóköz (space)
1 0 1 0 0 0 0 0 --- Szóköz (space)
0 0 1 1 0 0 1 1 --- 3
0 0 1 1 0 1 0 1 --- 5
0 0 1 0 1 1 1 0 --- Tizedespont (.)
1 0 1 1 1 0 0 0 --- 8
0 0 1 1 1 0 0 1 --- 9
1 0 1 1 0 1 1 1 --- 7
0 1 0 1 1 0 0 1 --- Y
0 0 1 0 1 1 0 1 --- Negatív előjel (-)
1 0 1 0 0 0 0 0 --- Szóköz (space)
1 0 1 0 0 0 0 0 --- Szóköz (space)
1 0 1 1 0 0 0 1 --- 1
0 0 1 1 0 1 0 1 --- 5
0 0 1 1 0 0 0 0 --- 0
0 0 1 0 1 1 1 0 --- Tizedespont (.)
1 0 1 1 1 0 0 0 --- 8
0 0 1 1 0 0 0 0 --- 0
0 0 1 1 0 0 0 0 --- 0
1 1 0 1 0 1 0 0 --- T
1 0 1 0 0 0 0 0 --- Szóköz (space)
1 0 1 1 0 0 0 1 --- 1
0 0 1 1 0 1 0 1 --- 5
0 0 0 0 1 0 1 0 --- Soremelés (LF)
```

Adatkiadás PRNT=1 állásnál:

```

 7 6 5 4 3 2 1 0
-----
1 1 0 1 1 0 0 0 --- X
0 0 1 1 0 0 1 1 --- 3
0 0 1 1 0 1 0 1 --- 5
0 0 1 0 1 1 1 0 --- Tizedes pont (.)
1 0 1 1 1 0 0 0 --- 8
0 0 1 1 1 0 0 1 --- 9
1 0 1 1 0 1 1 1 --- 7
0 1 0 1 1 0 0 1 --- Y
0 0 1 0 1 1 0 1 --- Negatív előjel (-)
1 0 1 1 0 0 0 1 --- 1
0 0 1 1 0 1 0 1 --- 5
0 0 1 1 0 0 0 0 --- 0
0 0 1 0 1 1 1 0 --- Tizedes pont (.)
1 0 1 1 1 0 0 0 --- 8
0 0 1 1 0 0 0 0 --- 0
0 0 1 1 0 0 0 0 --- 0
1 1 0 1 0 1 0 0 --- T
1 0 1 1 0 0 0 1 --- 1
0 0 1 1 0 1 0 1 --- 5
0 0 0 0 1 0 1 0 --- Soremelés (LF)

```

Periféria zárás: PCLOSn

A POPEN paranccsal megnyitott perifériát a PCLOS paranccsal le kell zárni. A PCLOS parancs után meg kell nevezni a lezárandó periféria számát. A lezáráskor még egy % karakter is kiküldésre kerül a perifériára, azaz minden adatkiadást egy % karakter zár le.

☞ *Megjegyzések:*

- Az adatkiadási parancsok sorrendje kötött: először POPEN paranccsal meg kell nyitni a megfelelő perifériát, utána jöhet az adatkiadás BPRINT, vagy DPRINT paranccsal, végül a megnyitott perifériát le kell zárni a PCLOS utasítással.
- A periféria megnyitása és lezárása bárhol a programban megadható. Például a program elején megnyitható, a program végén lezárható, és közben a program bármely, a két utasítás közé eső részén adat kiküldhető.
- Adatkiadás közben végrehajtott M30, vagy M2 parancs megszakítja az adatátvitelt. Ha ezt el akarjuk kerülni, az M30 parancs végrehajtása előtt várakozni kell adatátvitel közben.
- A megnyitott periféria paramétereinek (baud rate, stop bitek száma, stb.) helyes beállítására ügyelni kell. Ezek állítása a paramétermező SERIAL csoportjában lehetséges.

22.14 NC és makró utasítások.

A programnyelvben megkülönböztethetünk NC és makró mondatokat.

NC mondatoknak tekintjük a hagyományos, G, M stb. kódokkal leírt mondatokat, még akkor is, ha az egyes címek értékei, nem csak számértéket vesznek fel, hanem változókat, vagy formulát.

Makró utasításoknak a következő mondatokat tekintjük:

- az értékadó utasítást tartalmazó mondatot: #i=#j
- a feltételes, vagy ciklusszervező utasítást tartalmazó mondatot: IF, WHILE
- a kontrolparancsokat tartalmazó mondatokat: GOTO, DO, END
- a makrohívást tartalmazó mondatokat: G65, G66, G66.1, G67, vagy azok a G, vagy M kódok, amelyek makrohívást indítanak.

– az alprogramhívást (M98 P, vagy A, B, C, S, T, M-re indított alprogram)

A program végrehajtása szempontjából a makró mondatot síkbeli szerszámsugár korrekciószámítás bekapcsolt állapotában nem tekinti funkció, vagy a kiválasztott síkon kívüli mozgást tartalmazó mondatnak.

22.15 A makromondatok végrehajtása.

A makromondatokat végrehajthatja a vezérlő az NC mondatok végrehajtásával párhuzamosan, vagy azt követően. Az NC és makromondatok végrehajtását szabályozó paraméter a 9161 SBSTM. Ha a paraméter:

=0: az NC és makró mondatokat a programban leírt sorrendben hajtja végre,

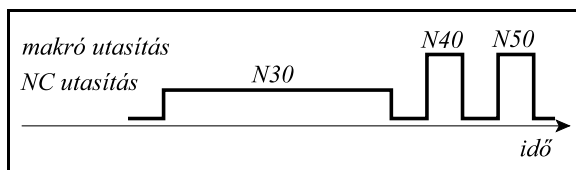
=1: az NC mondatok végrehajtása közben végrehajtja a makró utasításokat

Példa:

SBSTM=0

```
%O1000
...
N10 #100=50
N20 #101=100
N30 G1 X#100 Y#101
N40 #100=60 (értékkadás N30 után)
N50 #101=120 (értékkadás N30 után)
N60 G1 X#100 Y#101
```

Az N40 és N50 mondatokban leírt értékkadást az N30 mondat végrehajtása után végzi el.

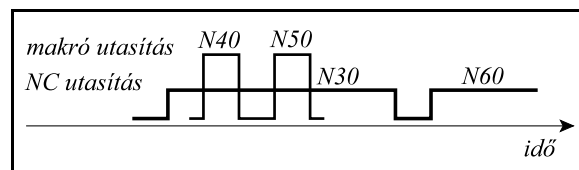


22.15-1 ábra

SBSTM=1

```
%O1000
...
N10 #100=50
N20 #101=100
N30 G1 X#100 Y#101
N40 #100=60 (értékkadás N30 közben)
N50 #101=120 (értékkadás N30 közben)
N60 G1 X#100 Y#101
```

Az N40 és N50 mondatokban leírt értékkadást az N30 mondatban folyó mozgás közben végzi el.



22.15-2 ábra

☞ **Következmények:**

- a program végrehajtása lassabb,
- ha az N30 mondat végrehajtását megszakítjuk, majd a megmunkálást újraindítjuk, mivel az N30 mondat változóit még nem írta át az N40, N50 mondat, a megmunkálás egyszerűen folytatható.

☞ **Következmények:**

- a program végrehajtása gyorsabb,
- ha az N30 mondat végrehajtását megszakítjuk, majd a megmunkálást újraindítjuk, mivel az N30 mondat változóit már átírta az N40, N50 mondat, a megmunkálás nem folytatható, csak, ha az N30 mondatra mondatkeresést indítunk.

22.16 Makrók és alprogramok kijelzése automata üzemmódban.

A makrók és alprogramok mondatait a vezérlés kijelezheti automata üzemmódban. Ha az *MD8* paraméter értéke 0, a 8000–tól 8999–ig számozott alprogramok és makrók végrehajtásakor a makró illetve alprogram mondatai nem kerülnek listázásra. *MD8* paraméter 1 állásánál ezek mondatai is listázásra kerülnek.

Ha az *MD9* paraméter értéke 0, a 9000–tól 9999–ig számozott alprogramok és makrók végrehajtásakor a makró illetve alprogram mondatai nem kerülnek listázásra. *MD9* paraméter 1 állásánál ezek mondatai is listázásra kerülnek.

22.17 STOP gomb használata makróutasítás végrehajtása közben.

A STOP gomb megnyomása, vagyis a programvégrehajtás felfüggesztése mindig a végrehajtása alatt álló makróutasítás befejezése után lesz hatásos.

Jegyzetek

Betűrendes index:

#0.	192	override.	29 , 47
#10001–#13999.	196	tartományváltás.	72
#1000–#1015.	194	Főprogram.	10
#1032.	194 , 195	Forgó tengely.	22
#1100–#1115.	195	Formátum.	10
#1132.	195	Fősík.	63
#195.	189	Főtengely.	63 , 182
#196.	189	Furat pozíciója.	154
#197.	189	Fúróciklus címei.	154
#198.	189	Fúróciklusok.	152
#199.	189	Fúróciklusok kódjai.	153
#1nn.	194	Fúróciklusok konfigurálása.	153
#3000.	197	Fúrótengely.	152
#3001.	198	Gyorsítás.	45
#3002.	198	Gyorsítás/lassítás.	45
#3003.	198	Hibajelzés.	197
#3004.	198	Hosszkorrekció.	15 , 60 , 79 , 174 , 178 , 181 , 202
#3006.	199	Hűtővíz.	14 , 72 , 73
#3007.	199	Ikremensrendszer.	16
#3901.	199	Inch.	37
#3902.	199	Inkremensrendszer.	37 , 38 , 44 , 80
#4001–#4130.	200	bemeneti.	17
#4201–#4330.	200	kimeneti.	17
#5201–#5326.	197	Inkrementális koordinátamegadás.	13
#5nn.	194	Interface.	194 , 195
Abszolút koordinátamegadás.	13	Interferenciavizsgálat.	106
Alkatrészprogram.	9	Interpoláció.	12
Alprogram.	10 , 72 , 74 , 75 , 101	Kezdőpont.	154
Átmérőben történő programozás.	14	Kicsinyítés.	111
Belső körív.	49	Kiesztergálás.	155 , 160 , 170 , 171
Belső sarok.	48 , 87 , 88 , 91 , 94 , 95 , 108	Kiindulási pont.	152
Bemeneti inkremensrendszer.	17	Kimeneti inkremensrendszer.	17
Bemeneti mértékrendszer.	16	Kimeneti mértékrendszer.	16
Biztonsági funkciók.	181	Konstans vágósebesség számítás.	14
Címlánc.	9	Kontúrkövetés.	101
decimal point.	212	iránya.	97
DNC üzemmód.	11	zavarproblémái.	106
Elágazás.	206 , 207	Koordinátaadatok.	37
Előkészítő funkciók.	12	Koordinátamegadás	
Előtolás.	12 , 198	abszolút.	13
Előtoláscsökkentés.	48 , 49	növekményes.	13
Értékhatar.	9 , 37 , 44 , 53 , 75 , 76 , 80	Koordinátarendszer.	13 , 57
Feltételes kifejezések.	206	eltolása.	60
Feltételes stop.	72	gépi.	57
Feltételvizsgálat.	193 , 207	létrehozása.	60
Főorsó.	14 , 43 , 65 , 72 , 73	lokális.	61
jeladó.	28	munkadarab.	58 , 197
orientálás.	67		

Kör.....	48 , 49 , 87 , 93 , 94 , 100	CLEG83.	156 , 163
adatmegadása.	23	CODES.	21 , 22 , 36 , 43 , 63
interpolálása.	12 , 23 , 42 , 63	CORNANGL.	48
iránya.	112	CORNOVER..	48
Korrekció.....	55 , 63 , 78 , 84	CRITFDIFn..	51
módosítása.	82 , 196	CRITICAN.	50
Követő nullák.....	37	CUTTING2..	198
Közbülső pont.	54 , 55 , 103	DECDIST.	48
Lassítás sarkoknál a pálya irányszögének		DELTV.....	106
.....	50	EXTER.	181
Lassítás sarkoknál a tengelyenkénti		FEED.	42
előtolás.....	51	FEEDCORN.	50
Lemaradás.	202	FEEDDIF.	50 , 51
Léptékezés.	111	G(901n)..	187
Lineáris gyorsítás.....	45 , 46	G31FD.	178
M kódok.....	72	G37FD.	179
Megközelítési pont.....	152	GEO.	51 , 52
Mérés.	178	INDEX_C1.	68
Mérőfunkciók.....	178	INDEX1.	68 , 165
Mértékrendszer.	37 , 38 , 44 , 80	INPOS.....	21
Metrikus.....	37	INTERFER.	106
Mondat.	9	LIMP2n.....	181
Mondatonkénti végrehajtás.	198	M(900n).	188
Mondatszám.....	200	M(902n).	188
Munkadarab koordinátarendszerek.	58	M_NUMB1.....	68
Munkatér behatárolás.....	181	MD8.	215
Nagyítás.....	111	MD9.	215
normális irányban fellépő gyorsulások		MULBUF.	20 , 46
.....	52	O_LINE.	195
Növekményes koordinátamegadás.....	13	ORIENT1.	67 , 160 , 170 , 171
Öröklődő funkciók.....	14 , 200	POSCHECK.	21
Override... ..	29 , 42 , 47 , 159 , 164 , 166 , 174 , 198	PRNT.	211
tiltás.....	19	PRTCNTM.	72 , 199
Paraméter		PRTREQRD.	199
A(9030).....	189	PRTTOTAL.	199
ACCDIST.....	48	RADDIF.	25
ACCN.	45 , 46 , 52	RAPDIST.	179 , 180
ACCTCn.....	46	RAPID6.	68
ADD.	179	REFPOS.	55
ALADIST.....	179 , 180	RETG73.	156 , 158
ANGLAL.	107 , 109	S(9033).	189
AXIS.....	197	SECOND..	53
B(9031).....	189	SERIAL.	213
C(9032).....	189	SKIPF.	178
CDEN.	50	STRKEN.....	181
CDIR6.....	68	T(9034).....	189
CHBFMOVE.	183	TAPDWELL.....	159 , 164
CIRCFMIN.....	52	WRPROT.	194
CIRCOVER.	49	Pontos megállás.....	47 , 198
		Pozícióinformációk.	200

Pozíciókijelzés.	60 , 61	Ugrás.	76
Pozicionálási sík.	152	Változó.	191
Pozícióvisszacsatolás.	67	0.	192
Program vége.	72	üres.	192
Programformátum.	10 , 191	Változó sugarú körív.	25 , 104 , 105
Programkezdet.	10	Változók.	193
Programnév.	10	globális.	189 , 191 , 194
Programozott stop.	72	lokális.	184 , 190 , 191 , 193
Programszám.	10 , 200	Várakozás.	53 , 91 , 101 , 156
Programvég.	10	Végállás.	182 , 183
R pont.	152	paraméteres.	54 , 182
Referenciapont.	13 , 54 , 55	Végrehajtási sorrend.	73
Rendszerváltozó.	178 , 194	Végtelen ciklus.	76
RS232.	11	Vektor megtartás.	99
Sarkok kerülése.	92 , 93 , 100 , 106	Vezérelt tengelyek.	16
Sarkok megmunkálása.	47 , 48	Vezető nullák.	37
Sarokív.	100	Visszahúzás.	152 , 156 , 163
Segédfunkció.	73		
Sík.	63		
váltás.	64 , 85		
Síkválasztás.	57 , 63		
STOP.	174 , 198 , 215		
feltételes.	72		
kapcsolók.	47		
programozott.	72		
tiltás.	42 , 47 , 159 , 164 , 166 , 167		
STOP állapot.	170 , 172		
Sugárkorrekció.	15		
Számábrázolás.	192		
Szerszámhossz mérés.	179		
Szerszámkezelés.	71		
Szerszám sugar korrekció			
irányváltás.	97		
síkbeli.	56 , 63 , 84		
térbeli.	55		
vektor megtartás.	99		
Szerszámszám.	14 , 71		
Szó.	9		
Tárolt pozíció.	178		
Teljes kör.	25 , 105		
Tengelyek			
elnevezése.	16		
inkremensrendszere.	16		
pozíciója.	178		
száma.	16		
Tiltott tartomány.	183		
Tizedespont.	37 , 185 , 210 , 211		
kiadása.	211		
Transzformációk.	101 , 110		
Tükrözés.	100 , 112 , 154-156 , 199		

