

**NCT<sup>®</sup> 99M**

**NCT<sup>®</sup> 2000M**

**Marógép és megmunkáló központ vezérlő**

**Programozási leírás**

---

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

H1148 Budapest Fogarasi út 7

✉ Levélcím: H1631 Bp. pf.: 26

☎ Telefon: (+36 1) 467 63 00

☎ Telefax:(+36 1) 363 6605

Villanyposta: [nct@nct.hu](mailto:nct@nct.hu)

Honlap: [www.nct.hu](http://www.nct.hu)

# Tartalomjegyzék

<b>1 Bevezetés</b>	<u>8</u>
1.1 Az alkatrészprogram	<u>9</u>
Szó	<u>9</u>
Címlánc	<u>9</u>
Mondat	<u>10</u>
Programszám és Programnév	<u>10</u>
Programkezdet, Programvég	<u>10</u>
Programformátum a tárban	<u>10</u>
Programformátum külső eszközzel történő kommunikációnál	<u>10</u>
Főprogram és alprogram	<u>10</u>
DNC csatorna	<u>11</u>
1.2 Alapfogalmak	<u>12</u>
<b>2 Vezérelt tengelyek</b>	<u>17</u>
2.1 A tengelyek elnevezése	<u>17</u>
2.2 A tengelyek mérték-, és ikremensrendszere	<u>17</u>
<b>3 Előkészítő funkciók (G kódok)</b>	<u>19</u>
<b>4 Az interpoláció</b>	<u>22</u>
4.1 A pozicionálás (G00)	<u>22</u>
4.2 Az egyenes interpoláció (G01)	<u>22</u>
4.3 A kör-, és a síkbeli spirális interpoláció (G02, G03)	<u>24</u>
4.4 A térbeli spirális interpoláció (G02, G03)	<u>27</u>
4.5 Egyenletes emelkedésű menet vágása (G33)	<u>29</u>
4.6 A polárkoordináta interpoláció (G12.1, G13.1)	<u>31</u>
4.7 A hengerinterpoláció (G7.1)	<u>35</u>
<b>5 A koordinátaadatok</b>	<u>37</u>
5.1 Abszolút és inkrementális programozás (G90, G91), az I operátor	<u>37</u>
5.2 Polárkoordinátás adatmegadás (G15, G16)	<u>37</u>
5.3 Inch/Metrikus átalakítás (G20, G21)	<u>39</u>
5.4 Koordinátaadatok megadása és értékhatára	<u>39</u>
5.5 Forgó tengelyek átfordulás kezelése	<u>40</u>
<b>6 Az előtolás</b>	<u>43</u>
6.1 A gyorsmeneti előtolás	<u>43</u>
6.2 A munkaelőtolás	<u>43</u>
6.2.1 Percenkénti (G94) és fordulatonkénti (G95) előtolás	<u>44</u>
6.2.2 A munkaelőtolás értékének behatárolása	<u>45</u>
6.3 Automatikus gyorsítás/lassítás	<u>46</u>
6.4 Az előtolásvezérlő funkciók	<u>47</u>
6.4.1 G09: pontos megállás	<u>47</u>
6.4.2 G61: pontos megállás üzemmód	<u>47</u>
6.4.3 G64: folyamatos forgácsolás üzemmód	<u>48</u>
6.4.4 G63: a százalék kapcsoló és stop tiltás üzemmód	<u>48</u>

6.4.5 Automatikus előtolás csökkentés belső sarkoknál. (G62)	48
6.4.6 Automatikus előtolás csökkentés belső köríveknél.	49
<b>7 A várakozás</b>	<b>50</b>
<b>8 A referenciapont</b>	<b>51</b>
8.1 Automatikus referenciapont felvétel (G28)	51
8.2 Az 1., 2., 3., 4. referenciapontra állás (G30)	52
8.3 Automatikus visszatérés a referenciaponttól (G29)	52
<b>9 Koordinátarendszerek, síkválasztás</b>	<b>54</b>
9.1 A gépi koordinátarendszer	54
9.1.1 A gépi koordinátarendszer beállítása	55
9.1.2 A gépi koordinátarendszer kiválasztása (G53)	55
9.2 A munkadarab koordinátarendszerek	55
9.2.1 A munkadarab koordinátarendszerek beállítása	55
9.2.2 A munkadarab koordinátarendszer kiválasztása	56
9.2.3 A munkadarab koordinátarendszerek eltolásának állítása programból	57
9.2.4 Új munkadarab koordinátarendszer létrehozása (G92)	57
9.3 A lokális koordinátarendszer	58
9.4 Síkválasztás (G17, G18, G19)	60
<b>10 A főorsó funkció</b>	<b>62</b>
10.1 A főorsó fordulatszám parancs (S kód)	62
10.2 A konstans vágósebesség számítás programozása	62
10.2.1 A konstans vágósebesség számítás megadása (G96, G97)	62
10.2.2 A konstans vágósebesség értékének behatárolása (G92)	63
10.2.3 Tengely kijelölése a konstans vágósebesség számításához	64
10.3 A főorsó pozíció visszacsatolás	64
10.4 Orientált főorsó megállás.	64
10.5 A főorsó pozícionálás (indexálás)	64
10.6 A főorsó fordulatszám ingadozás figyelése (G25, G26)	65
<b>11 A szerszámkezelés</b>	<b>68</b>
11.1 Szerszámhívási parancs (T kód)	68
11.2 Programformátum szerszámszám programozására	68
<b>12 Vegyes és segédfunkciók</b>	<b>70</b>
12.1 Vegyes funkciók (M kódok)	70
12.2 Segédfunkciók (A, B, C kódok)	71
12.3 A különböző funkciók végrehajtási sorrendje	71
<b>13 Az alkatrészprogram szervezése</b>	<b>72</b>
13.1 A mondat szám (N cím)	72
13.2 Feltételes mondat kihagyás (/ cím)	72
13.3 Főprogram és alprogram	72
13.3.1 Az alprogram hívása	73
13.3.2 Visszatérés alprogramból	73
13.3.3 Ugrás főprogramon belül	75

<b>14 A szerszámkorrekció</b> .....	<u>76</u>
14.1 Hivatkozás szerszámkorrekcióra (H és D) .....	<u>76</u>
14.2 Szerszámkorrekciós értékek módosítása programból (G10) .....	<u>77</u>
14.3 A szerszámhossz–korrekció (G43, G44, G49) .....	<u>77</u>
14.4 A szerszámeltolás (G45...G48) .....	<u>79</u>
14.5 A síkbeli szerszámsugár korrekció (G38, G39, G40, G41, G42) .....	<u>83</u>
14.5.1 A sugárkorrekció számítás bekapcsolása. Ráállítás a kontúrra. ....	<u>86</u>
14.5.2 A sugárkorrekció számítás bekapcsolt állapota. Haladás a kontúron. ....	<u>90</u>
14.5.3 A szerszámsugár korrekciószámítás kikapcsolása. Leállítás a kontúrról. ....	<u>93</u>
14.5.4 Irányváltás a sugárkorrekció számításban. ....	<u>96</u>
14.5.5 A vektor megtartás programozása (G38) .....	<u>98</u>
14.5.6 Sarokív programozása (G39) .....	<u>99</u>
14.5.7 Általános tudnivalók a síkbeli sugárkorrekció alkalmazása esetére .....	<u>100</u>
14.5.8 A kontúrkövetés zavarproblémái. Interferenciavizsgálat. ....	<u>105</u>
14.6 A háromdimenziós szerszámkorrekció (G41, G42) .....	<u>109</u>
14.6.1 A háromdimenziós szerszámkorrekció be-, kikapcsolása (G40, G41, G42) .....	<u>109</u>
14.6.2 A háromdimenziós korrekciós vektor .....	<u>110</u>
<b>15 Különleges transzformációk</b> .....	<u>111</u>
15.1 Koordinátarendszer elforgatása (G68, G69) .....	<u>111</u>
15.2 Léptékezés (G50, G51) .....	<u>112</u>
15.3 Tükrözés (G50.1, G51.1) .....	<u>113</u>
15.4 A különleges transzformációk programozási szabályai .....	<u>114</u>
<b>16 Automatikus geometriai számítások</b> .....	<u>116</u>
16.1 Letörés és lekerekítés programozása .....	<u>116</u>
16.2 Egyenes megadása irányszögével .....	<u>117</u>
16.3 Síkbeli metszéspontszámítások .....	<u>119</u>
16.3.1 Két egyenes metszéspontja .....	<u>119</u>
16.3.2 Egyenes és kör metszéspontja .....	<u>121</u>
16.3.3 Kör és egyenes metszéspontja .....	<u>123</u>
16.3.4 Két kör metszéspontja .....	<u>125</u>
16.3.5 A metszéspontszámítások láncolása .....	<u>127</u>
<b>17 Fúróciklusok</b> .....	<u>128</u>
17.1 A fúróciklusok részletes leírása. ....	<u>134</u>
17.1.1 Nagysebességű mélyfúróciklus (G73) .....	<u>134</u>
17.1.2 Balmenetfúró ciklus (G74) .....	<u>135</u>
17.1.3 Kiesztergálás automatikus szerszámelhúzással (G76) .....	<u>136</u>
17.1.4 A ciklusállapot kikapcsolása (G80) .....	<u>137</u>
17.1.5 Fúróciklus, kiemelés gyorsmenettel (G81) .....	<u>137</u>
17.1.6 Fúróciklus várakozással, kiemelés gyorsmenettel (G82) .....	<u>138</u>
17.1.7 Mélyfúróciklus (G83) .....	<u>139</u>
17.1.8 Menetfúró ciklus (G84) .....	<u>140</u>
17.1.9 Menetfúró ciklus kiegyenlítő betét nélkül (G84.2, G84.3) .....	<u>141</u>
17.1.10 Fúróciklus, kiemelés előtolással (G85) .....	<u>144</u>
17.1.11 Fúróciklus, gyorsmeneti kiemelés álló főorsóval (G86) .....	<u>145</u>

17.1.12 Fúróciklus, kézi működtetés a talpponton/ Kiesztergálás visszafelé, automatikus szerszámelhúzással (G87)	146
17.1.13 Fúróciklus, várakozás után kézi működtetés a talpponton (G88)	148
17.1.14 Fúróciklus, talpponton várakozás, kiemelés előtolással (G89)	149
17.2 Megjegyzések a fúróciklusok használatához	149
<b>18 Mérőfunkciók</b>	<b>151</b>
18.1 Mérés a maradék út törlésével (G31)	151
18.2 Automatikus szerszámhossz mérés (G37)	152
<b>19 Biztonsági funkciók</b>	<b>154</b>
19.1 Programozható munkatér behatárolás (G22, G23)	154
19.2 Paraméteres végállás	155
19.3 Tiltott tartomány figyelés mozgásindítás előtt.	156
<b>20 A felhasználói makró</b>	<b>157</b>
20.1 Az egyszerű makróhívás (G65)	157
20.2 Az öröklődő makróhívás	158
20.2.1 Makróhívás minden mozgásparancs után: (G66)	158
20.2.2 Makróhívás minden mondatból: (G66.1)	159
20.3 Felhasználói makróhívás G kódra	160
20.4 Felhasználói makróhívás M kódra	160
20.5 Felhasználói alprogramhívás M kódra	161
20.6 Felhasználói alprogramhívás T kódra	161
20.7 Felhasználói alprogramhívás S kódra	162
20.8 Felhasználói alprogramhívás A, B, C kódra	162
20.9 Az alprogramhívás és makróhívás közti különbség	162
20.9.1 Többszörös hívás	163
20.10 A felhasználói makró formátuma	164
20.11 A programnyelv változói	164
20.11.1 Változó azonosítása	165
20.11.2 Hivatkozás változóra	165
20.11.3 Üres változók	165
20.11.4 A változók számábrázolása	165
20.12 A változók típusai	166
20.12.1 Lokális változók	166
20.12.2 Globális változók	167
20.12.3 Rendszerváltozók	167
20.13 A programnyelv utasításai	175
20.13.1 Az értékadó utasítás	175
20.13.2 Aritmetikai műveletek és függvények	176
20.13.3 Feltételes kifejezések	179
20.13.4 Feltétel nélküli elágazás	179
20.13.5 Feltételes elágazás	179
20.13.6 Feltételes utasítás	179
20.13.7 Ciklusszervezés	180
20.13.8 Adatkiadási parancsok	182
20.14 NC és makró utasítások.	186
20.15 A makromondatok végrehajtása.	187

20.16 Makrók és alprogramok kijelzése automata üzemmódban. ....	<u>187</u>
20.17 STOP gomb használata makróutasítás végrehajtása közben. ....	<u>188</u>
20.18 Üregmaró makróciklus .....	<u>188</u>
<b>Jegyzetek</b> .....	<u>192</u>
<b>Betűrendes index</b> .....	<u>193</u>

November 29, 2002

© Copyright NCT November 29, 2002

E leírás tartalmára minden kiadói jog fenntartva. Utánnomáshoz – 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	szögkoordináták, hosszkoordináták, segédfunkciók	I, -, *
R	körsugár, segédadat	I, -, *
I, J, K	kör középpont koordináta, segédkoordináta	-, *
E	segédkoordináta	-, *
F	előtolás	*
S	főrsó fordulatszám	*
M	vegyes funkció	1 - 999
T	szerszámszám	1 - 9999
H, D	hossz-, és sugárkorrekció száma	1 - 99
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  $\text{LF}$  (Line Feed) karakter választja el egymástól. A mondatokban nem kötelező a mondatszám használata. Hogy a mondatvéget és a mondatkezdetet el lehessen különíteni egymástól a képernyőn a mondat elejét új sorban kezdjük, és  $\text{>}$  karaktert teszünk eléje és az egy sornál hosszabb mondatoknál az új sorba kerülő szavakat egy karakterrel beljebb kezdjük.

### 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  $\text{LF}$  (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/1N12345G1X0Y...LFG2Z5...LF....LF
...LF
...LF
N1G40...M2LF
%
```

A fenti karaktersorban

$\text{LF}$  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.

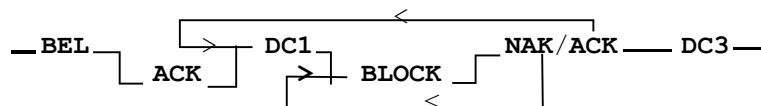
### DNC csatorna

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 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* hibüzenetet ad. Ha a vezérlés BLOCK hibát érzékel és NAK-kal válaszol meg kell ismételni a BLOCK-ot.

## 1.2 Alapfogalmak

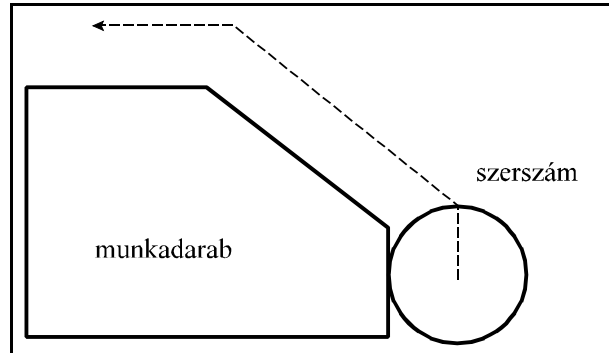
### Az interpoláció

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 Y__  
X__ Y__
```



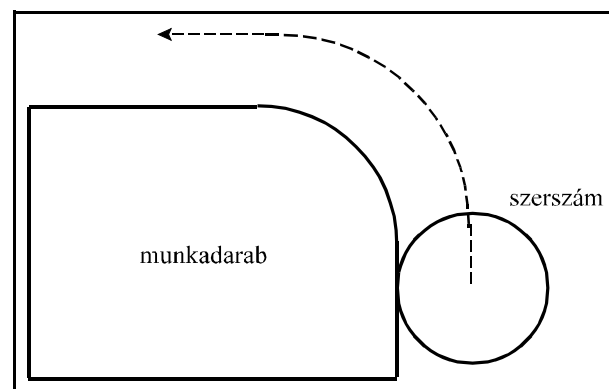
1.2-1 ábra

Szerszámmozgás körív mentén:

program:

```
G03 X__ Y__ R__
```

Habár a valóságban általában az asztal mozog a munkadarabbal és nem a szerszám, ebben a leírásban mindig a szerszám munkadarabhoz képesti mozgásáról fogunk beszélni a továbbiakban.



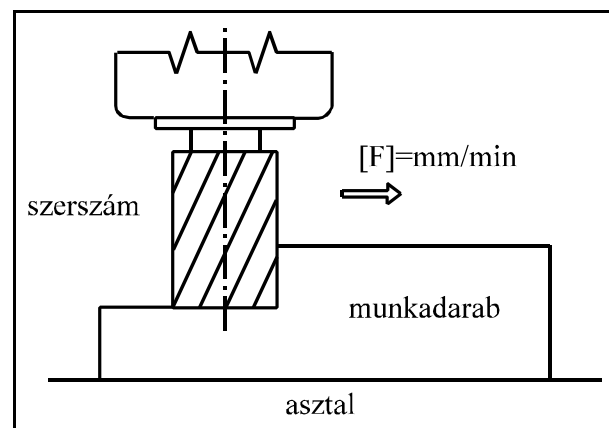
1.2-2 ábra

### 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: F150 jelentése 150 mm/perc.



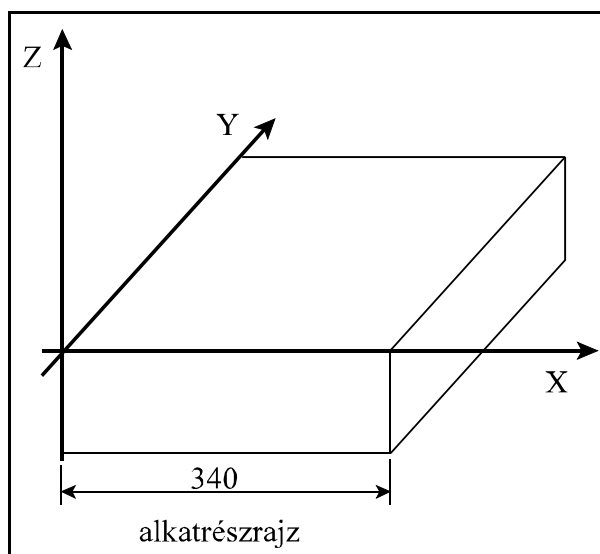
1.2-3 ábra

### Referenciapont

A referenciapont a szerszámgépen egy fix pont. A gép bekapcsolása után a szánokat referenciapontra kell küldeni. Ezután a vezérlés már abszolút koordinátájú adatokat is tud értelmezni.

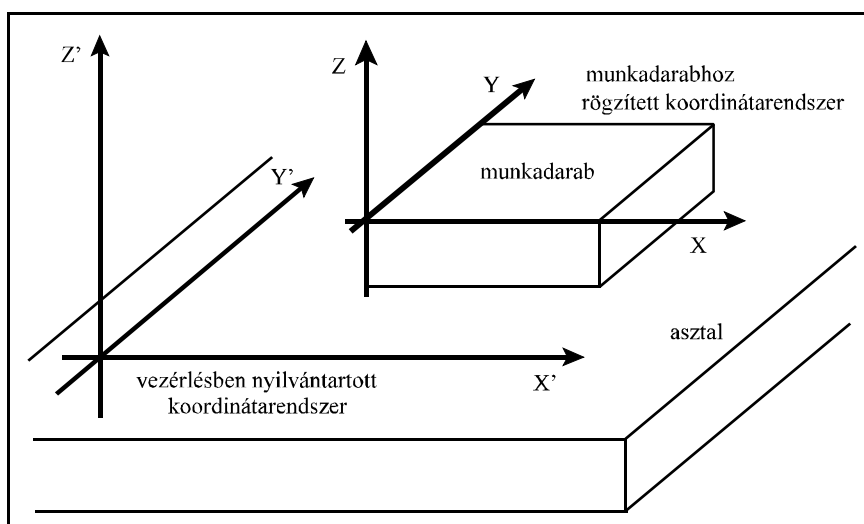
### Koordinátarendszer

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: X340 jelentése: a munkadarab koordinátarendszer 340 mm koordinátájú pontja.



1.2-4 ábra

A vezérlésben nyilvántartott koordinátarendszer, amelyben a vezérlés a méreteket értelmezi különbözik a munkadarab koordinátarendszerétől. Ahhoz, hogy a vezérlés korrekt munkadarabot készítsen ennek a két koordinátarendszernek a nullpontját ugyanarra a pozícióra kell hozni. Ezt elérhetjük például úgy, hogy: a szerszám középpontját a munkadarab egy



1.2-5 ábra

ismert pozíciójú pontjára visszük és a vezérlés koordinátarendszerét erre az értékre állítjuk át.

### Abszolút koordinátamegadás

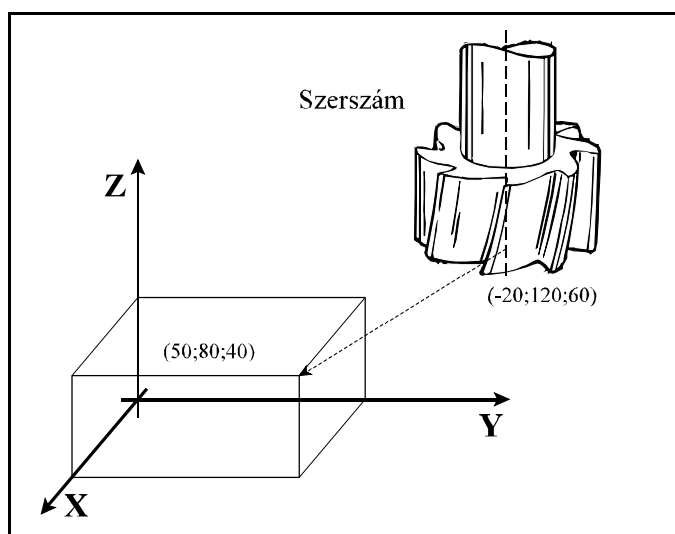
Abszolút koordinátamegadás esetén a szerszám a koordinátarendszer kezdőpontjától számított távolságra mozog, azaz a koordinátán megadott pozíciójú pontra.

Az abszolút adatmegadás kódja: G90.

A

G90 X50 Y80 Z40

utasítás sor a fenti pozíciójú pontra mozgatja a szerszámot, bárhol is állt a pa-



1.2-6 ábra

rancskiadás előtt.

### Növekményes (inkrementális) koordinátamegadás

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

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

A

```
G91 X70 Y-40 Z-20
```

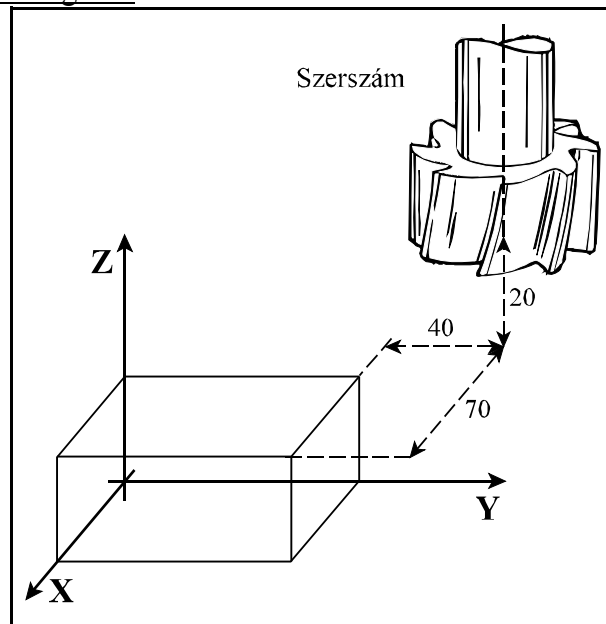
utasítás az előző pozíciótól a fenti távolságra mozgatja el a szerszámot.

Növekményes adatot koordinátánként is lehet definiálni. A koordináta címe utáni I karakter jelenti azt, hogy az adott koordináta értékét inkrementálisként kezelje.

A

```
G90 XI-70 Y80 Z40
```

utasításban X adatát inkrementálisan értelmezi, Y és Z adatát a G90 kód miatt abszolútként.



1.2-7 ábra

### 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 Y30 F180  
N16 X30  
N17 Y100
```

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.

### Szerszámszám

A megmunkálás során különböző szerszámokkal kell a különböző forgácsolási műveleteket elvégezni. A szerszámokat számokkal különböztetjük meg egymástól. A szerszámokra T kóddal

hivatkozhatunk. A programban a T25 utasítás azt jelenti, hogy váltsuk be a 25-ös számú szerszámot. A gép kialakításától függően a szerszámcsere történhet kézzel, vagy automatikusan.

### Vegyes funkciók

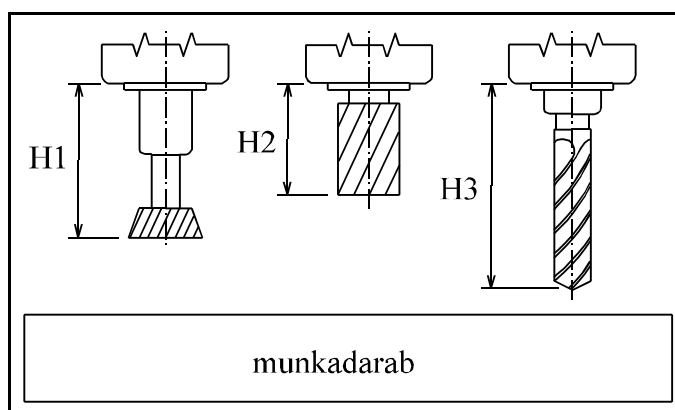
A megmunkálás során számos ki-, bekapcsolási műveletet kell elvégezni. Például: elindítani a főorsót, bekapcsolni a hűtővizet. Ezeket a műveleteket a vegyes vagy M funkciók segítségével lehet elvégezni. Például: az

M3 M8

utasításokban M3 jelentése: főorsó forgás óramutatóval megegyező irányba, M8 jelentése pedig: kapcsolj 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. Ha a programban azt

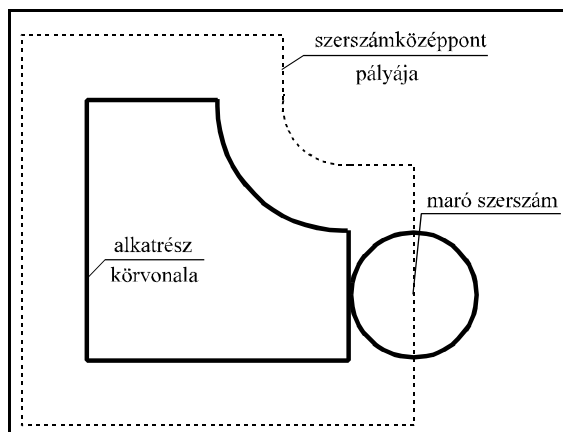


1.2-8 ábra

akarjuk, hogy a szerszám csúcsa mozogjon a megadott pontra, le kell hívni a hosszadatnak az értékét egy kísérő kóddal segítségével. Ez H címen lehetséges. Például: a H1 utasítás az 1-es számú hosszadatra vonatkozik. Ettől kezdve a szerszám csúcsát mozgatja a vezérlő a megadott pontra. Ezt a műveletet nevezzük a hosszkorrekció bekapcsolásának.

### Sugárkorrekció

Egy kontúr marásánál a darabot különböző sugarú szerszámokkal kell megmunkálni. Annak érdekében, hogy a programban ne a szerszámközpont pályáját kelljen leírni, figyelembe véve a szerszámok sugarát, hanem a darab tényleges kontúradatait, be kell vezetni a sugárkorrekciót. A sugárkorrekciók értékeit be kell állítani a vezérlőben. A programban D címen hivatkozhatunk a továbbiakban a sugárkorrekcióra.



1.2-9 ábra

### Kopáskorrekció

A megmunkálás során a szerszámok kopnak. Az így adódó méretváltozást, akár hossz-, akár sugárirányban a kopáskorrekción lehet figyelembe venni. A szerszám kopását be lehet állítani a vezérlőben. Minden korrekciós csoporthoz (a szám amelyre H vagy D címen hivatkozunk) tartozik egy geometriai érték, vagyis a szerszám eredeti hossza illetve sugara, és egy kopásérték. A vezérlés a korrekció lehívásakor a két érték összegével korrigálja a mozgást.



## 2 Vezérelt tengelyek

Tengelyek száma alapkiépítésben	3 tengely
Bővítőtengelyek száma	5 tengely (8 tengely összesen)
Egyidejűleg mozgatható tengelyek száma	8 tengely (lineáris interpolációval)

### 2.1 A tengelyek elnevezése

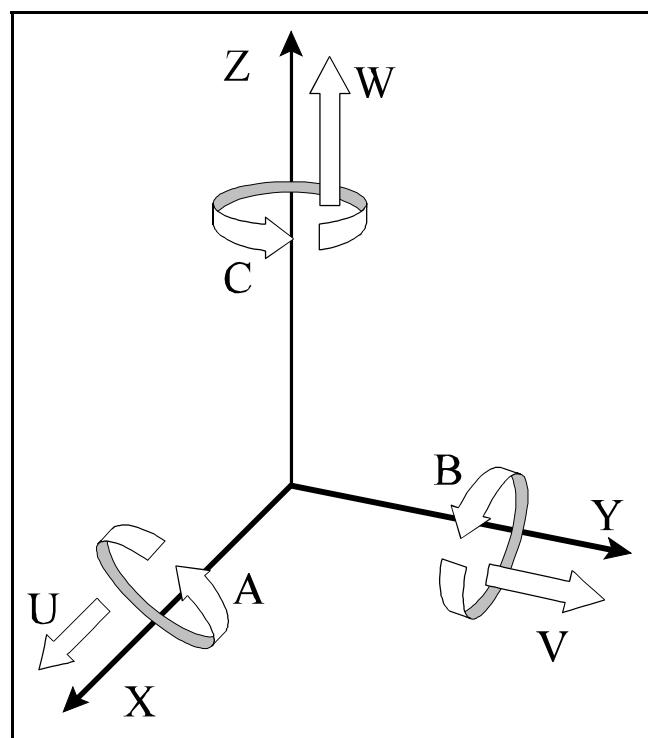
A vezérelt tengelyek elnevezését a paramétertárban lehet definiálni. Itt ki lehet jelölni, hogy melyik fizikai tengely milyen címre mozogjon.

Alapkiépítésben egy maróvezérlőben a tengelyek nevei: X, Y és Z.

A bővítőtengelyek elnevezése a tengely típusától függ.

A lineáris mozgást végző bővítőtengelyek lehetséges elnevezése: U, V és W. Ha ezek a 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.



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

### 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	<u>22</u>
G01*		egyenes interpoláció	<u>22</u>
G02		kör, spirális interpoláció ójm.	<u>24</u>
G03		kör, spirális interpoláció óje.	<u>24</u>
G04	00	várakozás	<u>50</u>
G05.1		többszörös előfeldolgozás üzemmód beállítása	
G09		pontos megállás az adott mondatban	<u>47</u>
G07.1		hengerkoordináta interpoláció	<u>35</u>
G10		programozott adatbeadás	<u>57</u> , <u>73</u> , <u>77</u>
G11		programozott adatbeadás tiltása	
G12.1	26	polárkoordináta interpoláció be	<u>31</u>
G13.1*		polárkoordináta interpoláció ki	<u>31</u>
G15*	17	polárkoordinátás adatmegadás kikapcsolva	<u>37</u>
G16		polárkoordinátás adatmegadás	<u>37</u>
G17*	02	$X_p Y_p$ sík választása	<u>60</u>
G18*		$Z_p X_p$ sík választása	<u>60</u>
G19		$Y_p Z_p$ sík választása	<u>60</u>
G20	06	inches adatmegadás	<u>39</u>
G21		metrikus adatmegadás	<u>39</u>
G22*	04	munkatér behatárolás bekapcsolása	<u>154</u>
G23		munkatér behatárolás kikapcsolása	<u>154</u>
G25*	8	főorsó fordulatszám ingadozás figyelés kikapcsolása	<u>65</u>
G26		főorsó fordulatszám ingadozás figyelés bekapcsolása	<u>65</u>
G28	00	programozott referenciapont felvétel	<u>51</u>
G29		visszatérés a referenciaponttól	<u>52</u>
G30		első, második, harmadik és negyedik referenciapontra állás	<u>52</u>
G31		mérés a maradék út törlésével	<u>151</u>
G33	01	menetvágás	<u>29</u>
G37	00	automatikus szerszámhosszmérés	<u>152</u>
G38		sugárkorrekciós vektor megtartása	<u>98</u>

3 Előkészítő funkciók (G kódok)

G kód	Csoport	Funkció	Oldal
G39		sarokív sugárkorrekcióval	99
G40*	07	szerszámsugár-korrekció számítás kikapcsolása	83, 89, 93
G41		szerszámsugár-korrekció számítás balról	83, 82, 86
G42		szerszámsugár-korrekció számítás jobbról	83, 82, 86
G43*	08	hosszkorrekció +	77
G44*		hosszkorrekció -	77
G45	00	szerszámeltolással növelés	79
G46		szerszámeltolással csökkentés	79
G47		kétszeres szerszámeltolással növelés	79
G48		kétszeres szerszámeltolással csökkentés	79
G49*	08	hosszkorrekció kikapcsolása	77
G50*	11	léptékezés kikapcsolása	112
G51		léptékezés	112
G50.1*	18	tükrözés kikapcsolása	113
G51.1		tükrözés bekapcsolása	113
G52	00	koordinátaeltolás	58
G53		pozicionálás a gép koordinátarendszerében	55
G54*	14	első munkadarab koordinátarendszer választása	56
G55		második munkadarab koordinátarendszer választása	56
G56		harmadik munkadarab koordinátarendszer választása	56
G57		negyedik munkadarab koordinátarendszer választása	56
G58		ötödik munkadarab koordinátarendszer választása	56
G59		hatodik munkadarab koordinátarendszer választása	56
G61	15	pontos megállás üzemmód	47
G62		előtoláscsökkentés sarkoknál	48
G63		override tiltás	48
G64*		folyamatos forgácsolás üzemmód	48
G65		egyszerű makrohívás	157
G66		öröklődő makrohívás minden mozgásparancs után	158
G66.1		öröklődő makrohívás minden mondatból	159
G67		öröklődő makrohívás törlése	158

G kód	Csoport	Funkció	Oldal	
G68	16	koordinátarendszer elforgatása	<u>111</u>	
G69*		koordinátarendszer–elforgatás kikapcsolása	<u>111</u>	
G73	09	nagysebességű mélyfűrőciklus	<u>134</u>	
G74		balmenetfűrő ciklus	<u>135</u>	
G76		kiesztergálás automatikus szerszámelhúzással	<u>136</u>	
G80*		ciklusállapot kikapcsolása	<u>137</u>	
G81		fűrőciklus, kiemelés gyorsmenettel	<u>137</u>	
G82		fűrőciklus várakozással, kiemelés gyorsmenettel	<u>138</u>	
G83		mélyfűrőciklus	<u>139</u>	
G84		menetfűrőciklus	<u>140</u>	
G84.2		jobbmenetfűrőciklus kiegyenlítőbetét nélkül	<u>141</u>	
G84.3		balmenetfűrőciklus kiegyenlítőbetét nélkül	<u>141</u>	
G85		fűrőciklus, kiemelés előtolással	<u>144</u>	
G86		fűrőciklus, gyorsmeneti kiemelés álló főorsónál	<u>145</u>	
G87		kiesztergálás visszafelé, automatikus/kézi szerszámelhúzással	<u>146</u>	
G88		fűrőciklus, kézi működtetés a talpponton	<u>148</u>	
G89		fűrőciklus, talpponton várakozás, kiemelés előtolással	<u>149</u>	
G90*		03	abszolút méretmegadás	<u>37</u>
G91*			növekményes méretmegadás	<u>37</u>
G92	00	koordinátarendszer beállítás	<u>57</u>	
G94*	05	percenkénti előtolás	<u>44</u>	
G95*		fordulatonkénti előtolás	<u>44</u>	
G96	13	konstans vágósebesség–számítás bekapcsolása	<u>62</u>	
G97*		konstans vágósebesség–számítás kikapcsolása	<u>62</u>	
G98*	10	visszatérés fűrőciklusból a kiindulási pontra	<u>129</u>	
G99		visszatérés fűrőciklusból az R (megközelítési) pontra	<u>129</u>	

☞ **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; G43, G44, G49; 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 megköötéssel, 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 pozicionálás (G00)

A

**G00** v

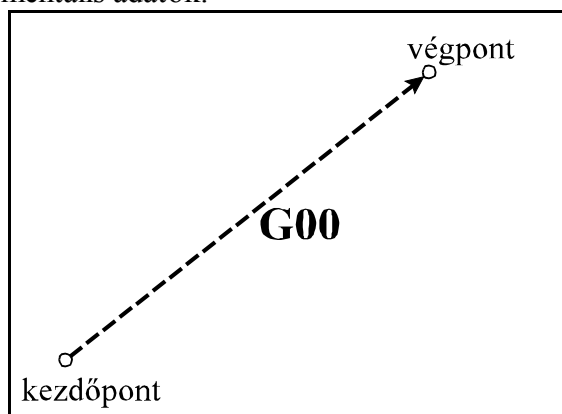
utasítás az aktuális koordináta-rendszerben való pozicionálásra vonatkozik.

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

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

A pozicionálás sebességét nem lehet programból állítani, az a szer-zámgép építője által paraméteren meghatározott, tengelyenként különböző értékkel történik. Több tengely egyidejű mozgása esetén a sebesség vektori eredőjét úgy számítja ki a vezérlő, hogy a pozicionálás 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 gyorsít, a mozgás befejezésekor lassít. A mozgás befejezése után [4.1-1](#) ábra



a vezérlés ellenőrzi a "pozícióban" jelet, ha a paraméterező *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 paraméterező *CODES* paramétercsoportjában meghatározott érték szerint G00, vagy G01 kód van érvényben.

### 4.2 Az egyenes interpoláció (G01)

A

**G01** v F

utasítás lineáris interpolációs módot állít be. A v értékre írt adatok lehetnek abszolút illetve inkrementális értékek, és az aktuális koordináta-rendszerben 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:

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

Előtolás az Y tengely mentén:

$$F_y = \frac{y}{L} F$$

.....  
 Előtolás az U tengely mentén:

$$F_u = \frac{u}{L} F$$

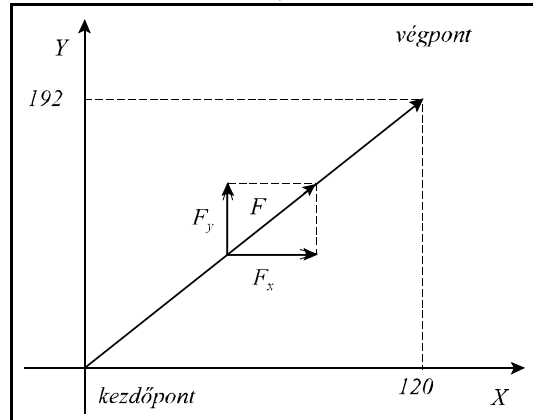
.....  
 Előtolás a C tengely mentén:

$$F_c = \frac{c}{L} F$$

x, y, .u, ..c a megfelelő tengelyek mentén programozott elmozdulásértékek,  
 L a programozott elmozdulás hossza:

$$L = \sqrt{x^2 + y^2 + \dots + u^2 + \dots + c^2}$$

G01 X100 Y80 F150



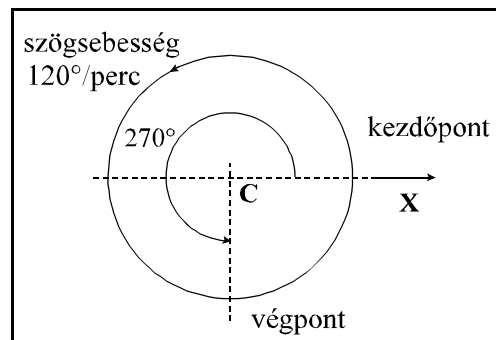
4.2-1 ábra

Forgó tengely mentén az előtolás °/perc dimenzióban értelmezett:

G01 B270 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 B45 F120

mondatban a Z illetve B irányú előtoláskomponensek:

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

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

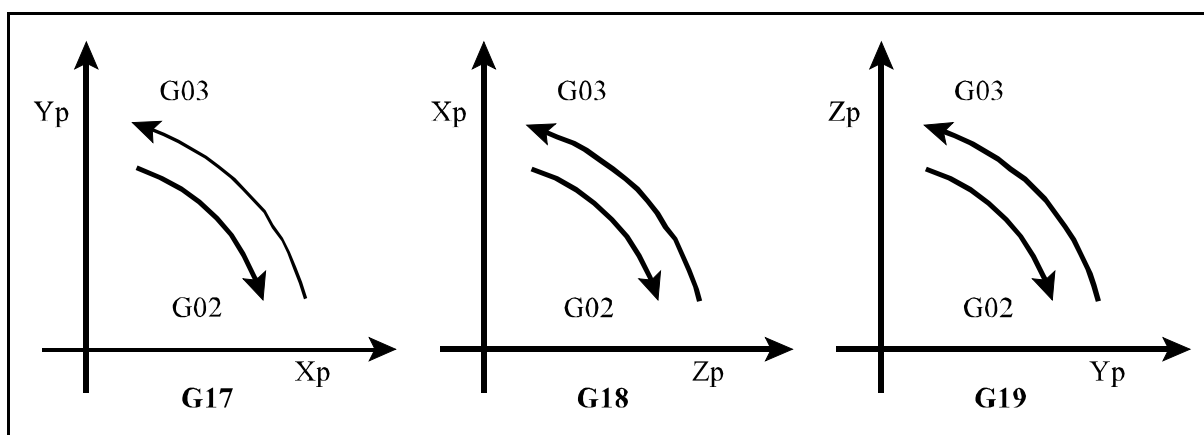
$$G17 \left\{ \begin{matrix} G02 \\ G03 \end{matrix} \right\} X_p Y_p \left\{ \begin{matrix} R \\ I J \end{matrix} \right\} F$$

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

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

utasítás sor 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

$X_p$ ,  $Y_p$ ,  $Z_p$  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.

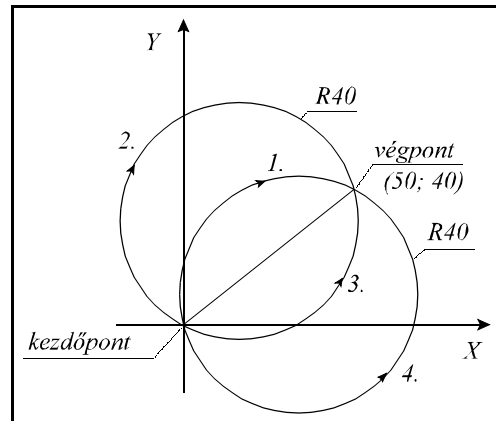
A kör további adatainak megadása kétféleképp történhet:

1. eset:

R címen, ahol R a kör sugara. Ekkor a vezérlés a kezdőpont 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  $180^\circ$ -nál kisebb ív mentén halad, ha R-en negatív számot adunk meg a  $180^\circ$ -nál nagyobb ívet járja be.

Például:

1. ívszakasz: G02 X50 Y40 R40
2. ívszakasz: G02 X50 Y40 R-40
3. ívszakasz: G03 X50 Y40 R40
4. ívszakasz: G03 X50 Y40 R-40

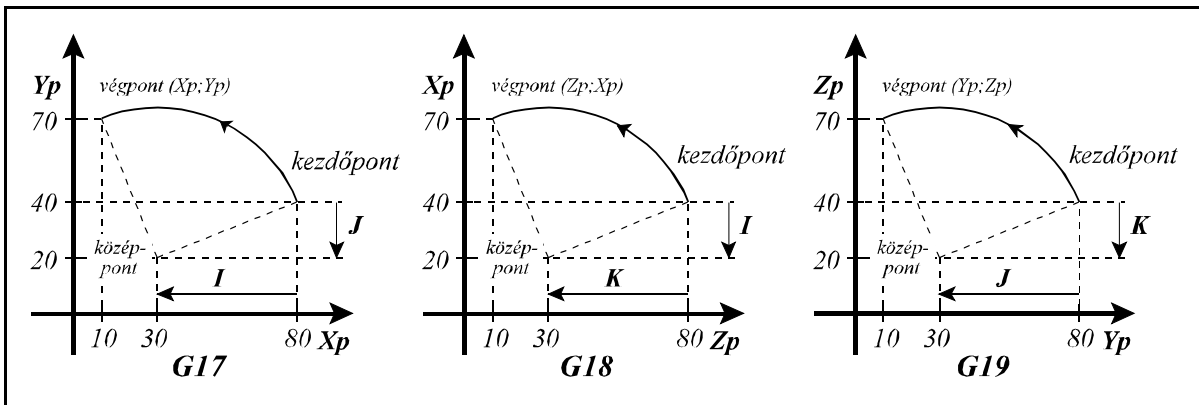


4.3-2 ábra

2. eset:

A kör középpontját I, J, K címen adjuk meg, az  $X_p$ ,  $Y_p$ ,  $Z_p$  tengelyekre. Az I, J, K címen megadott értékeket mindig inkrementálisan értelmezi a vezérlő, úgy, hogy az I, J, K értékek által definiált vektor a kör kezdőpontjából a kör középpontjába mutat. Például:

- G17 esetén: G03 X10 Y70 I-50 J-20  
 G18 esetén: G03 X70 Z10 I-20 K-50  
 G19 esetén: G03 Y10 Z70 J-50 K-20



4.3-3 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 Y100 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:

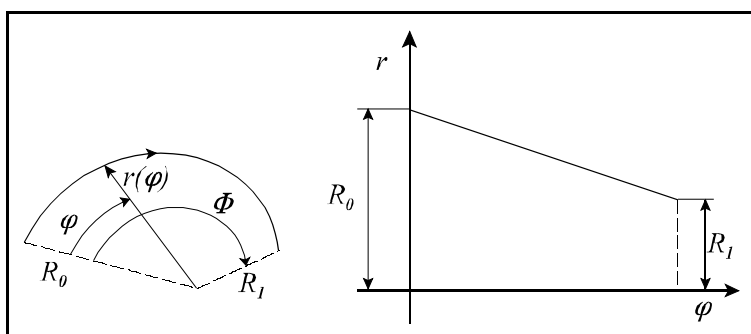
a. sem sugarat (R), sem I, J, K-t nem tartalmaz,

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 érték-nél a vezérlés a szerszámot olyan síkbeli spirális pálya mentén mozgatja, amelynél a sugár a központi szög függvényében lineárisan változik.

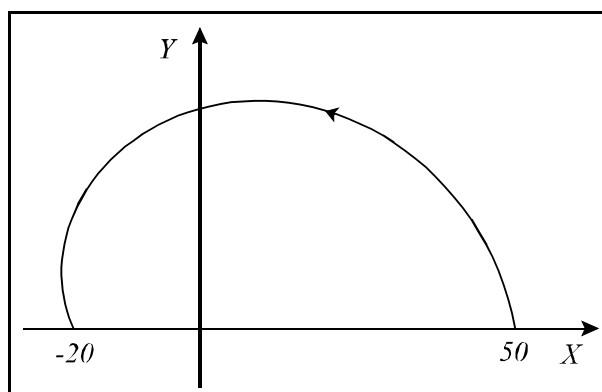
Változó sugarú körív interpolációjánál nem a pályamenti sebesség, hanem a szögsebesség lesz állandó.



4.3-5 ábra

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

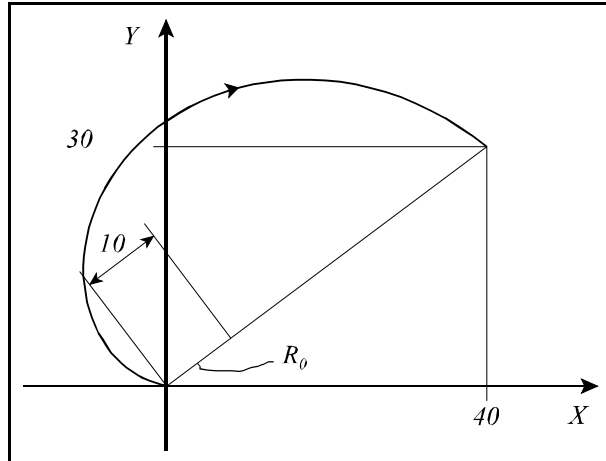
```
G17 G90 G0 X50 Y0
G3 X-20 I-50
```



4.3-6 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:

```
G17 G0 G90 X0 Y0
G2 X40 Y30 R10
```



4.3-7 ábra

#### 4.4 A térbeli spirális interpoláció (G02, G03)

A

$$G17 \begin{Bmatrix} G02 \\ G03 \end{Bmatrix} X_p Y_p q \begin{Bmatrix} R \\ I J \end{Bmatrix} F$$

$$G18 \begin{Bmatrix} G02 \\ G03 \end{Bmatrix} X_p Z_p q \begin{Bmatrix} R \\ I K \end{Bmatrix} F$$

$$G19 \begin{Bmatrix} G02 \\ G03 \end{Bmatrix} Y_p Z_p q \begin{Bmatrix} R \\ J K \end{Bmatrix} F$$

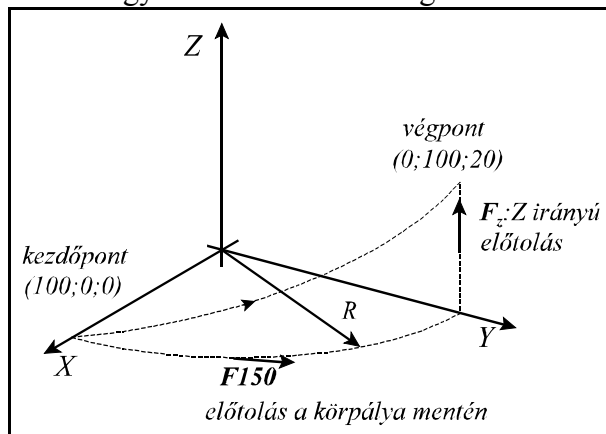
utasításon térbeli spirális interpolációt definiál.

A körinterpolációtól abban különbözik, hogy egy harmadik, a kör síkjába nem eső tengelyt "q"-t is a köradatba írunk. A q tengely mentén a vezérlés egyszerű elmozdulást végez.

Az F címen megadott előtolás a körpálya mentén érvényesül. A q tengely mentén az Fq előtoláskomponens a következő összefüggésből adódik:

$$F_q = \frac{L_q}{L_{iv}} F$$

ahol  $L_q$ : elmozdulás a q tengely mentén,  
 $L_{iv}$ : a körív hossza,  
 F: a programozott előtolás,  
 $F_q$ : előtolás a q tengely mentén



4.4-1 ábra

Például:

```
G17 G03 X0 Y100 Z20 R100 F150
```

A

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

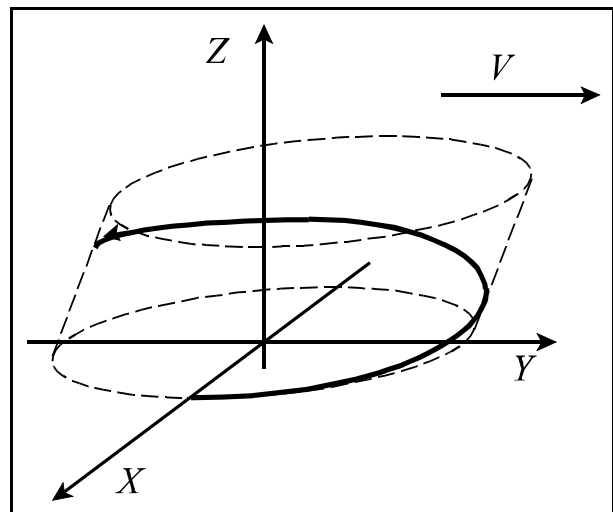
$$G18 \left\{ \begin{matrix} G02 \\ G03 \end{matrix} \right\} X_p Z_p q r s \left\{ \begin{matrix} R \\ I K \end{matrix} \right\} F$$

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

utasítássor olyan többdimenziós térbeli spirális interpolációt definiál, ahol q, r, s a körinterpolációban részt nem vevő opcionális tengelyek.

Például a

G17 G3 X0 Y-100 Z50 V20 I-100  
utasítássor ferde henger palástja mentén viszi a  
szerszámot, ha V az Y-nal párhuzamos tengely.



4.4-2 ábra

☞ *Megjegyzések:*

- Abban az esetben, ha a paramétermező *HELICALF* paramétere 1-re van állítva a vezérlő a programozott előtolást a térbeli pálya mentén érvényesíti.
- Abban az esetben, ha a kiválasztott síkban megadott kör sugara változó, az interpoláció a megadott kúp palástja mentén történik.
- A megadott szerszámsugár korrekció mindig a kör síkjában érvényesül.

#### 4.5 Egyenletes emelkedésű menet vágása (G33)

A

G33 v F Q

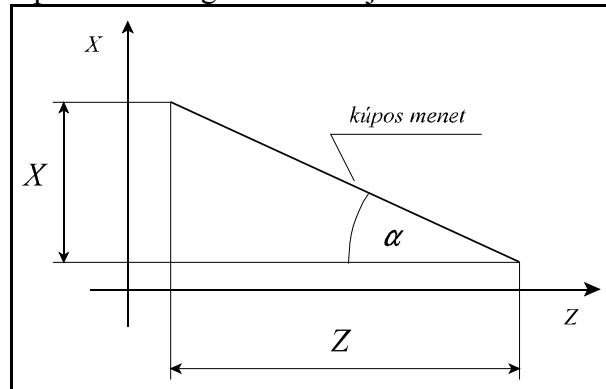
G33 v E Q

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.



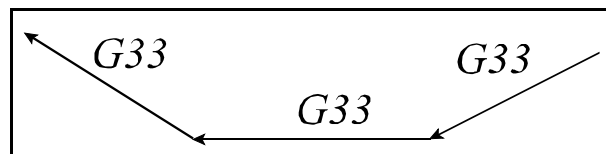
4.5-1 ábra

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 vágunk menetet:



4.5-2 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:

```
N50 G90 G0 X0 Y0 S100 M4
N55 Z2
N60 G33 Z-100 F2
N65 M19
N70 G0 X5
N75 Z2 M0
N80 X0 M4
N85 G4 P2
N90 G33 Z-100 F2
...
```

Magyarázat:

N50, N55: a szerszámmal a furat középpontja fölé áll, a főorsót az óramutató járásával ellentétes forgásirányban indítja,

N60: első menetvágási művelet, menetemelkedés 2 mm,

N65: orientált főorsó megállás (a főorsó egy fix pozícióban áll meg),

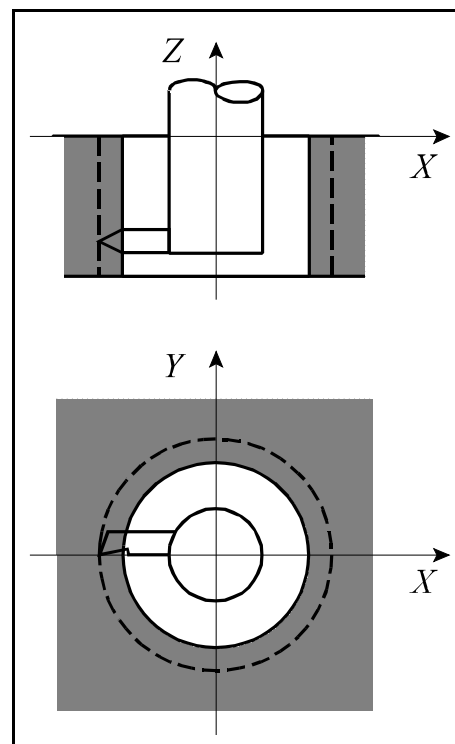
N70: szerszámelhúzás az X tengely mentén,

N75: szerszámvisszahúzás a furat tetejére, programozott megállás, a kezelő beállítja a szerszámot a következő menetvágási művelethez,

N80: visszaállás a furat közepére, a főorsó újraindítása,

N85: várakozás, hogy a főorsó felvegye a megfelelő fordulatot,

N90: második menetvágási művelet.



4.5-3 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<sub>n</sub> 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.

#### 4.6 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

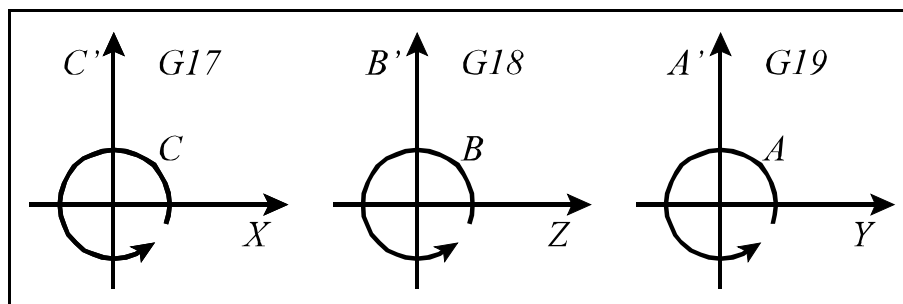
**G12.1** polárkoordináta interpoláció beutasítás bekapcsolja a polárkoordináta üzemet. Az ezután következő programrészben a marószerszám pályáját derékszögű koordinátarendszerben, a hagyományos módon, egyenes és körinterpoláció programozásával írhatjuk le, a szerszámsugár korrekció figyelembe vételével.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

A

**G13.1** polárkoordináta interpoláció kiutasítás 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 hossz méretek 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 kötelezően úgy kell megválasztani, hogy az egybeesik a körtengely forgástengelyével.

##### A tengelyek helyzete a polárkoordináta interpoláció bekapcsolása pillanatában

A polárkoordináta interpoláció bekapcsolása előtt (G12.1 utasítás) gondoskodni kell arról, hogy a **körtengely a 0 pozíciójű** 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. 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,
- pozicionálás a gépi koordinátarendszerben: G53.

Az előtolás a polárkoordináta interpoláció során

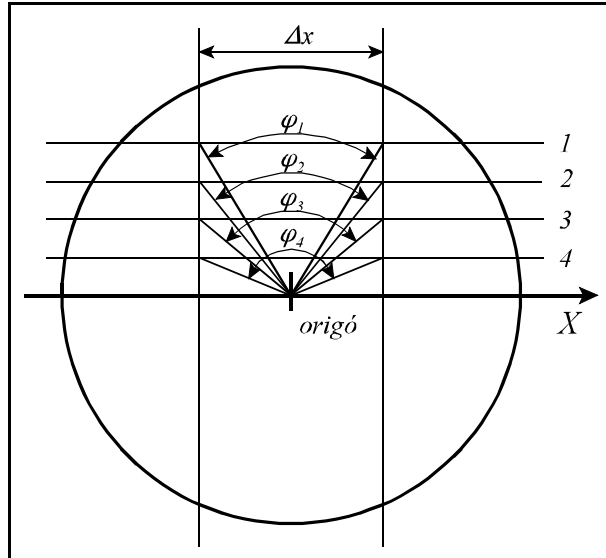
Az előtolás értelmezése a polárkoordináta interpoláció bekapcsolt állapotában a derékszögű interpolációnál megszokott módon, pályamenti sebességként történik: a munkadarab és a szerszám relatív sebességét adja meg.

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  $\Delta x$  elmozdulás tartozik. A  $\Delta x$  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 értéket a vezérlés fokozatosan csökkenti a pályamenti előtolást.

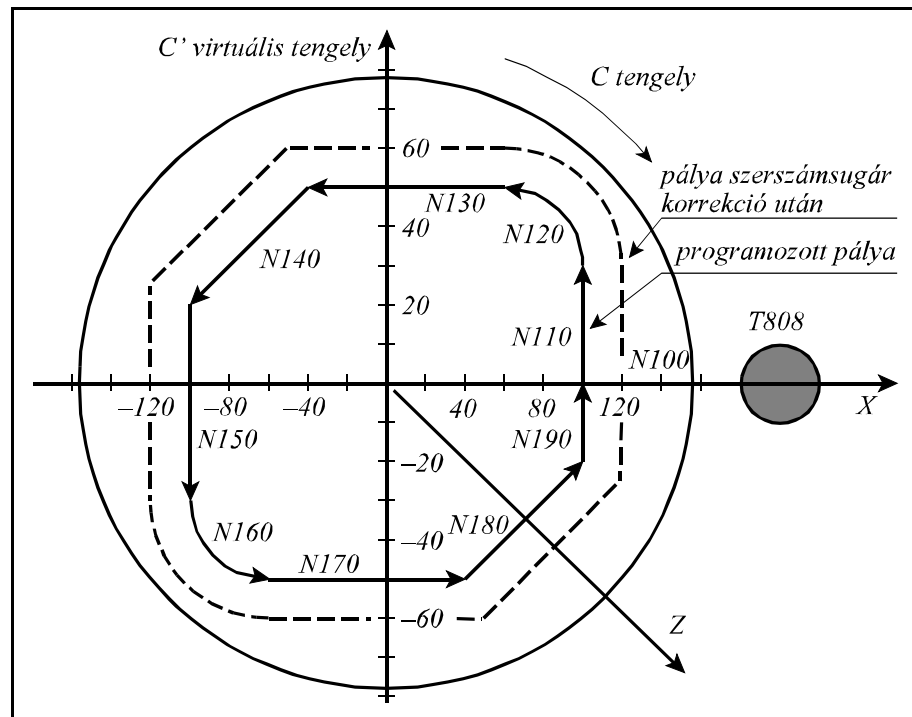
A fent elmondottak alapján kerüljük olyan programok írását, amelyeknél a szerszám középpontja az origó közelében halad.



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 hossztengelyen 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.*

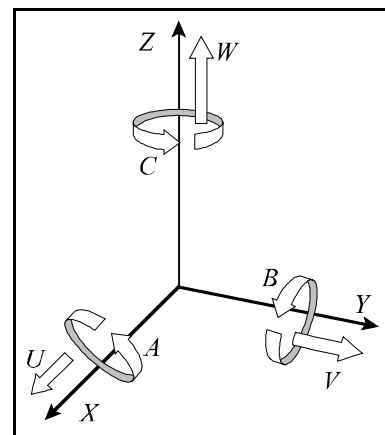
A kör sugara mindig mm-ben, vagy inch-ben kerül értelmezésre, soha nem fokban.

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:



4.7-1 ábra

- A hengerinterpoláció bekapcsolása (G7.1 Qr utasítás) csak G40 állapotban lehetséges,
- Ha a hengerinterpoláció állapotában bekapcsoltuk a G41, vagy G42-t, a hengerinterpoláció kikapcsolása (G7.1 Q0 utasítás) előtt G40-et kell programozni.

#### Programozási megszorítások a hengerinterpoláció során

A hengerinterpoláció bekapcsolt állapotában az alábbi utasítások nem használhatók:

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

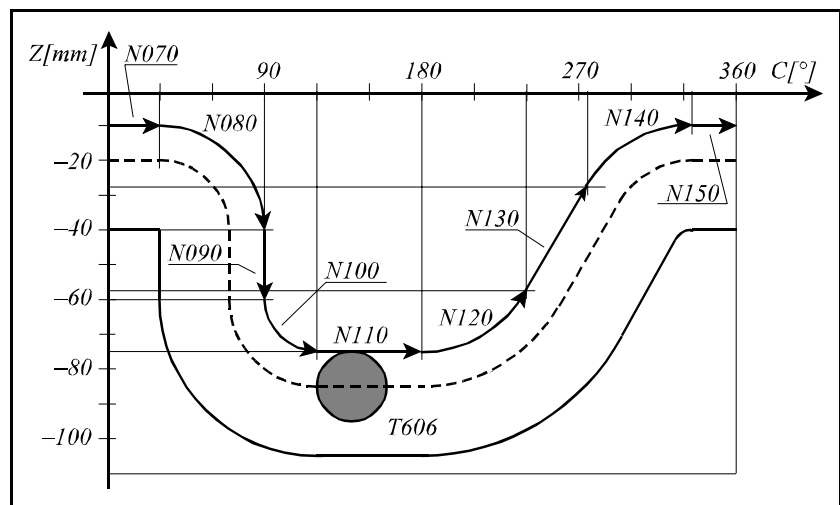
#### Mintapélda

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 ( $1^\circ$ ) eső elmozdulás:

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

Az ábrán látható tengelyrendezés G19 síkváltásnak felel meg.



4.7-2 ábra

```
%O7602 (HENGERINTERPOLACIO)
```

```
...
```

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

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

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

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

(G19: C-Z sík váltása)

(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
```

```
N180 G0 X100
```

(hengerinterpoláció kikapcsolása)

```
...
```

```
%
```

## 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:

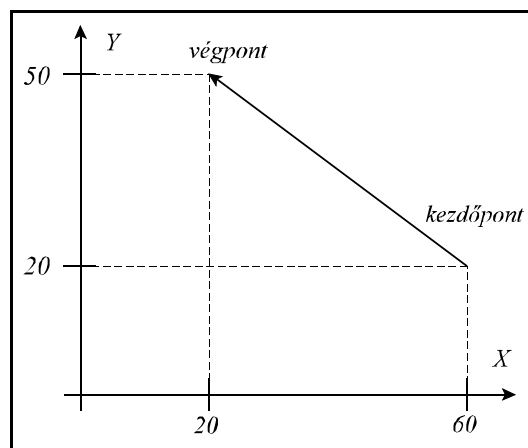
Az ábra alapján kétféleképp lehet a mozgást programozni:

```
G90 G01 X20 Y50
G91 G01 X-40 Y30
```

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

A fenti példa a következőképp is megoldható:

```
(G90) G01 XI-40 YI30
G01 X20 YI30
G01 XI-40 Y50
```



5.1-1 ábra

### 5.2 Polárkoordinátás adatmegadás (G15, G16)

A végpont koordináták értékei polárkoordinátás adatmegadással, azaz szög és rádiusz megadásával is bevihetők.

**G16:** Polárkoordinátás adatmegadás bekapcsolása

**G15:** Polárkoordinátás adatmegadás kikapcsolása

Resetre a vezérlés G15 állapotba kerül. A G15, G16 öröklődő funkciók.

A polárkoordinátás adat a G17, G18, G19 által meghatározott síkban érvényes. Adatmegadáskor a sík vízszintes tengelyének címét tekinti a sugárnak, függőleges tengelyét pedig a szögnek. Például: G17 állapotban az X (U) címre írt adat a sugár, Y (V) címre írt adat a szög. Vigyázat: G18 állapotban Z a vízszintes tengely (R adat) és X a függőleges tengely (szögadat).

A szög adatmegadása esetén az óramutató járásával ellentétes irány a szög pozitív iránya, az óramutató járásával megegyező irány a szög negatív iránya.

A többi tengely adatait Descartes (derékszögű) koordinátás adatnak veszi. A sugarat és a szöget is meg lehet adni úgy abszolút, mind növekményes értéként is.

Amikor a sugarat abszolút adatként adjuk meg, az aktuális koordinátarendszer origója lesz a polárkoordinátarendszer kezdőpontja:

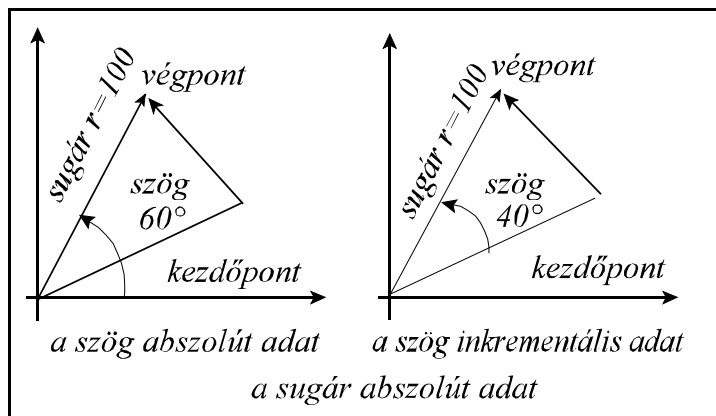
Példa:

G90 G16 G01 X100 Y60 F180

A szög is és a sugár is abszolút adat, a szerszám a 100 mm sugarú 60°-os pontra fut.

G90 G16 G01 X100 Y140 F180

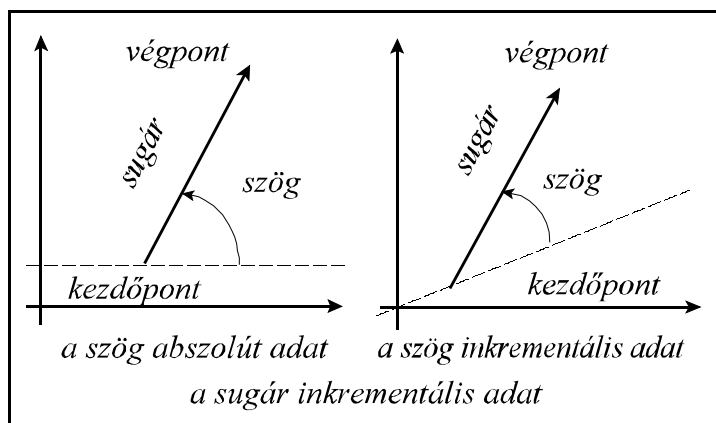
A szög növekményes adat. Az előző szöghelyzethez képest 40°-kal megy arrébb.



5.2-1 ábra

Amikor a sugarat növekményes értéként definiáljuk, a tengelyek mondat eleji pozíciójától számítva mozogja le a megadott sugarat a megadott szög irányában:

G16 polárkoordinátás adatmegadás bekapcsolt állapotában is lehet kört programozni. A kört meg lehet adni sugárral és I, J, K-val is. Az utóbbi esetben azonban I, J, K címet mindig derékszögű adatnak tekinti a vezérlés. Ha az aktuális koordinátarendszer középpontja egybeesik a kör középpontjával polárkoordinátás adatmegadásal többfordulatú kör, vagy spirális is programozható.



5.2-2 ábra

Példa:

(G17 G16 G90) G02 X100 Y-990 Z50 R-100

A fenti mondatban egy 2 egész 3/4 fordulatú spirált adtunk meg, az óramutató járásával megegyező forgásiránnyal. Többfordulatú kör programozásakor ügyeljünk, hogy G2 irány esetén negatív polárszöget, G3 irány esetén pedig pozitív polárszöget programozzunk.

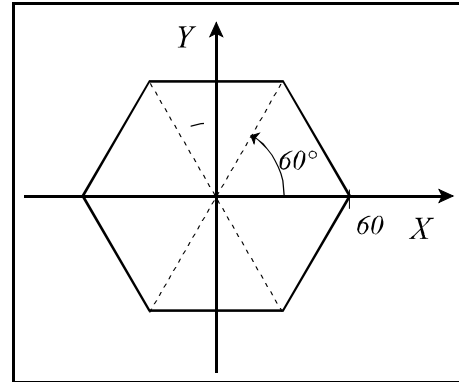
☞ *Megjegyzés:*

A következő utasításokban előforduló címetek a vezérlés nem tekinti polárkoordinátásnak, még ha a G16 állapot be is van kapcsolva:

- G10 beállító utasításban előforduló koordináták,
- G52 koordinátaeltolás,
- G92 koordinátabeállítás,
- G53 gépi koordinátarendszerben írt pozicionálás,
- G68 koordinátarendszer elforgatás,
- G51 léptékezés (nagyítás) bekapcsolása,
- G50.1 programozható tükrözés.

Mintapélda: hatszög marása

```
N1 G90 G17 G0 X60 Y0 F120
N2 G16 G1 Y60
N3 Y120
N4 Y180
N5 Y240
N6 Y300
N7 Y360
N8 G15 G0 X100
```



5.2-3 ábra

### 5.3 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 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.

### 5.4 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 inkrementsrendszer által meghatározott értéknél kisebb számot is értelmez a vezérlő. Például: X1.23456 parancsot, ha IR-B a beállított inkrementsrendszer:

- 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	$\pm 0.0001-9999.9999$	

### 5.5 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 mínusz végtelen között tartjuk nyilván, hanem a tengely periódikusságát figyelembe véve, pl.:  $0^\circ$  és  $360^\circ$  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°- +359.999° közötti tartományban 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 <b>C.ROTARY</b> =1, 0243 <b>ROLLOVEN_C</b> =1 0263 <b>ROLLAMNT_C</b> = =360000	Abszolút koordinátamegadással programozott mondat	A mondat hatására megtett elmozdulás	Pozíció a mondat végén
0246 <b>ABSHORT_C</b> =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 <b>ABSHORT_C</b> =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 koordináta- megadással 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.

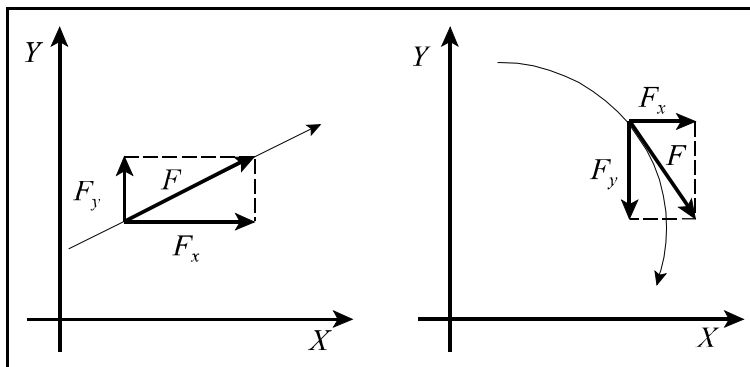
A programozott előtolás lineáris- (G01) és körinterpolációs (G02, G03) mondatokban érvényesül.

Az előtolás a programozott pálya mentén tangenciálisan érvényesül.

**F** : előtolás érintő irányú nagysága (programozott érték)

$F_x$ : előtoláskomponens az X irányban

$F_y$ : előtoláskomponens az Y irányban



6.2-1 ábra

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

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:** perenkénti 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 perenkénti dimenzióban értendő.

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

bemeneti mértékrendszer	kimeneti mértékrendszer	inkrementsrendszer	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 munkaelőtölés értékének behatárolása

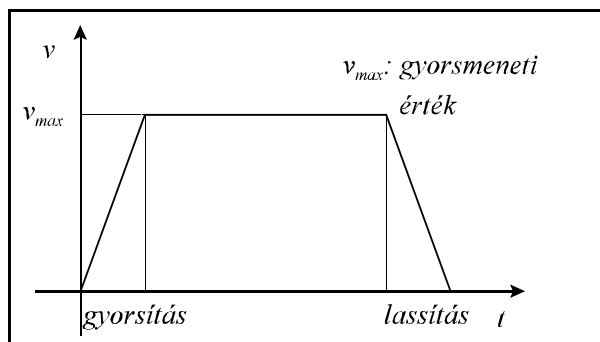
Az adott gépen programozható maximális előtolást a gép gyártója a paramétermezőben tengelyenként bekorlátozhatja. Az itt beállított érték mindig percenkénti dimenzióban értelmezett. Ez az érték egyben a SZÁRAZ FUTÁS kapcsoló bekapcsolt állapotában az előtoló mozgások sebessége. Ha ennél nagyobb előtolást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

A kézi mozdítás során az előtolás maximális értéke a paramétermezőben külön behatárolható,

hogyan az emberi reakcióidőt is be lehessen kalkulálni a megálláshoz.

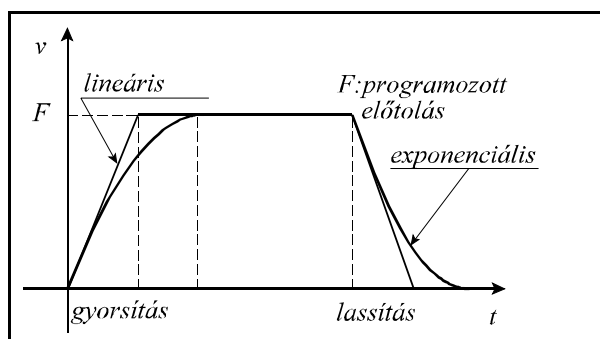
### 6.3 Automatikus gyorsítás/lassítás

Gyorsmeneti pozicionálásoknál a vezérlés automatikusan lineáris gyorsítást hajt végre induláskor és lassítást leálláskor. A gyorsítás mértékét a gép építője határozza meg tengelyenként, a paramétermező  $ACCn$  paraméterén, a gép dinamikus teherbírásának függvényében.



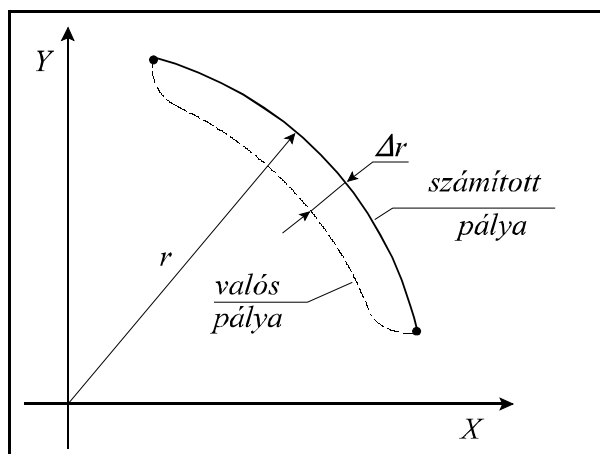
6.3-1 ábra

Előtoló mozgások esetén a tangenciális (programozott) előtolásértéket a vezérlő lineáris gyorsítással veszi fel, és lineáris lassítással csökkenti le. Ez a módszer a hagyományos, exponenciális gyorsítással szemben azt az előnyt kínálja, hogy a kívánt sebességet a gép hamarabb éri el, azonos időállandó beállítását tételezve fel mindkét esetben, így a gyorsulás/lassulás, azaz az effektív szánmozgás ideje lecsökken.



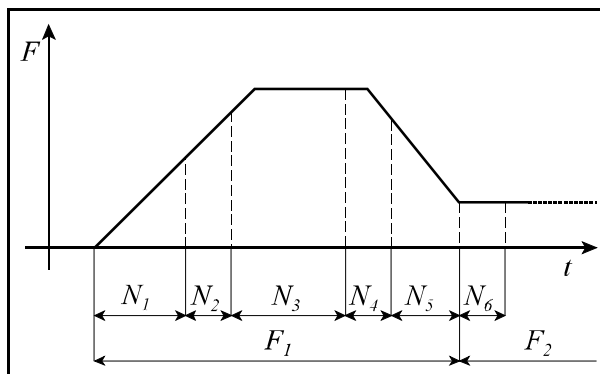
6.3-2 ábra

A lineáris gyorsítás másik előnye az exponenciálissal szemben, hogy kör nagysebességű megmunkálása esetén a szervorendszer reakcióideje miatt kialakult profiltorzulás, azaz sugárhiba, kisebb lineáris gyorsítással, mint exponenciálissal.



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 a vezérlő. Az 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 programozott sebességgel kezdje a megmunkálást.



6.3-4 ábra

Mozgatógombbal, és kézikerékkel történő mozgások esetén a gyorsítás/lassítás szintén lineáris. Értékét tengelyenként paraméter határozza meg.

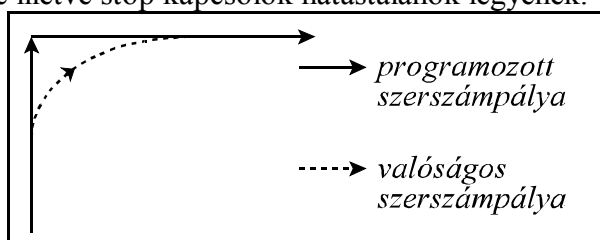
#### 6.4 Az előtolásvezérlő funkciók

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 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, 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.

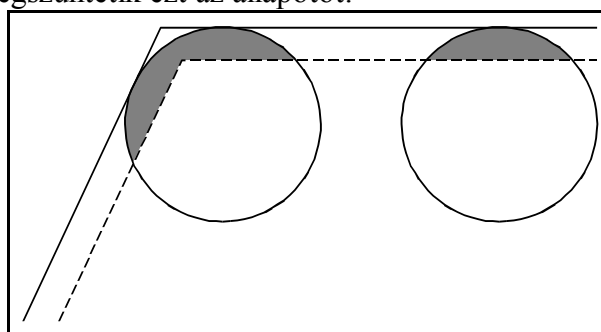
Belső sarkok megmunkálása esetén a szerszámra ható erő megnövekszik a sarok előtti és utáni szakaszon. Annak érdekében, hogy a szerszám ne rezegjen be, és a felület megfelelő maradjon, a vezérlő G62 bekapcsolt állapotában a belső sarkok előtti és utáni szakaszon az előtolást automatikusan csökkenti.

Az előtoláscsökkentés a következő feltételek mellett hatásos:

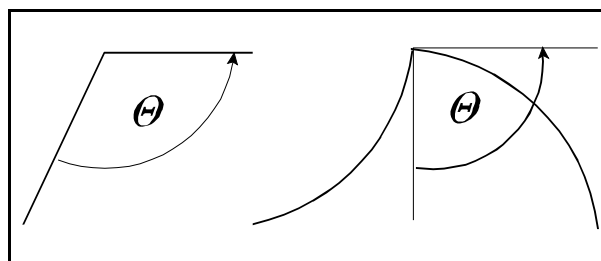
- 1. a síkbeli szerszámsugár korrekció bekapcsolt állapotában (G41, G42),
- 2. G0, G1, G2, G3 mondatok között,
- 3. a kiválasztott síkban végzett mozgásoknál,
- 4. ha a sarkot belülről kerüli a szerszám,
- 5. ha a sarok szöge kisebb, mint egy a paramétermezőben meghatározott szög,
- 6. a sarok előtt,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

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

A  $\Theta$  belső szög értékét a *CORNANGL* paraméteren lehet beállítani az 1–180° szögtartományban.

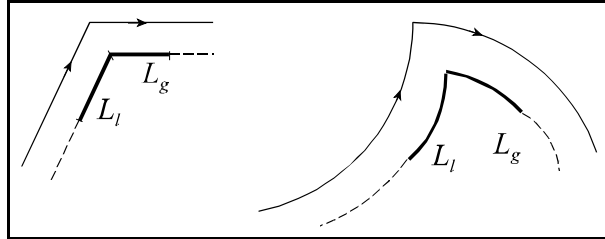


6.4.5-1 ábra



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 százalékos 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:

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

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

#### 6.4.6 Automatikus előtoláscsökkentés belső köríveknél.

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:

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

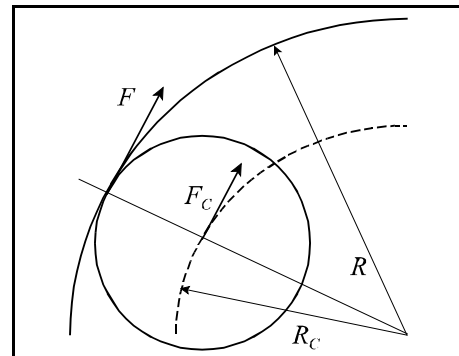
ahol  $F_c$ : a szerszámsugár középpont előtolása (korrigált előtolás)

$R$ : 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 az előtolás csökkentés minimumát. A körsugár miatti override összeszorozódik az előtolás és sarokoverride értékekkel, és így kerül kiadásra.



6.4.6-1 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 fordulaton.

P értékhatára: 0.001 - 99999.999 fordulat.

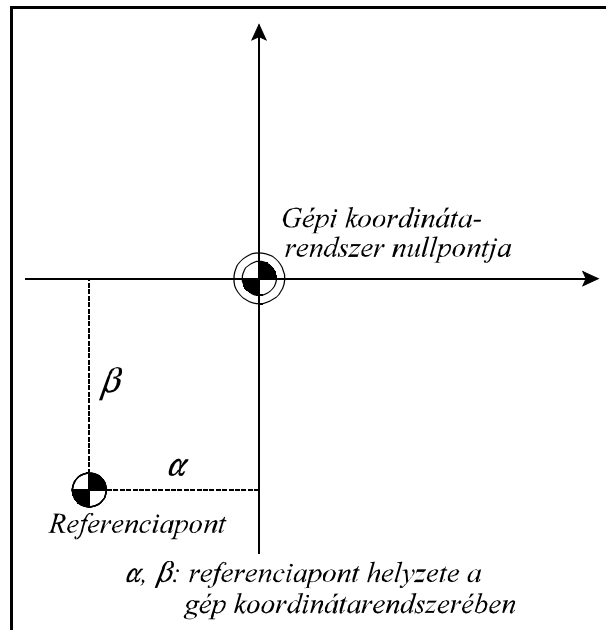
A *SECOND* paraméter függvényében a késleltetés vonatkozhat mindig másodpercre is G94, és G95 állapottól függetlenül.

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

A várakozás alatt az 5. interpolációs állapotot kijelző státuszablakban a **VÁR** felirat 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 mozgatsnál, a paramétermezőben meghatározott módon felveszi az alaphelyzetet.

Nem öröklődő kód.

#### ☞ Megjegyzések:

- 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, Y=0

G28 Y200      közbülső pont: X=100, Y=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, amelyekre nem történt koordinátaadás a G29-et megelőző G28 vagy G30 mondatban az eggyel korábbi értéket veszi figyelembe. A közbülső pontra a szerszámhossz, szerszámeltolás és a 3 dimenziós szerszámugár korrekció figyelembe vételével áll rá.

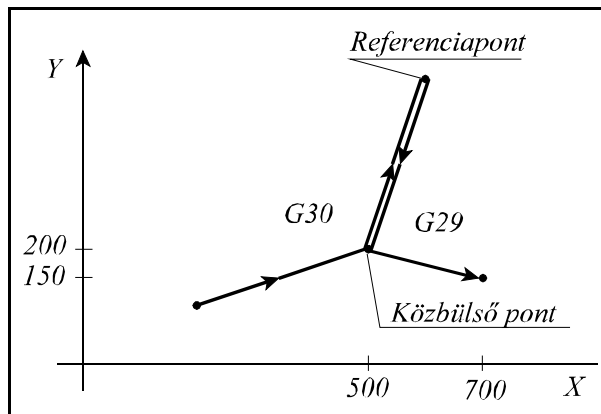
A közbülső pont koordinátája mindig az aktuális munkadarab koordinátarendszerben érvényes. Tehát, ha például a refpontra állás után és a G29 utasítás előtt munkadarab koordinátarendszer

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:

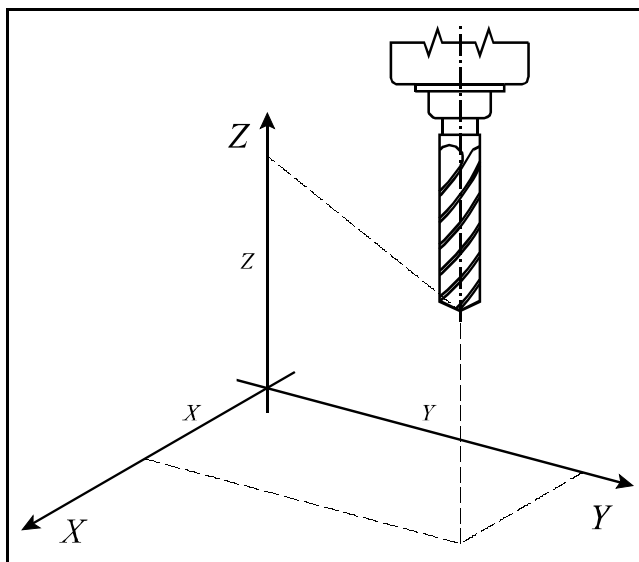
```
...  
G90  
...  
G30 P1 X500 Y200  
G29 X700 Y150  
...  
...
```



8.3-1 ábra

## 9 Koordinátarendszerek, síkválasztás

A programban egy pozíciót, ahova a szerszámot akarjuk mozgatni, koordinátaadatokkal adunk meg. Ha 3 tengelyünk van (X, Y, Z) a szerszám pozícióját három koordinátaadat X \_\_\_\_ Y \_\_\_\_ Z \_\_\_\_ fejezi ki:



9-1 ábra

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

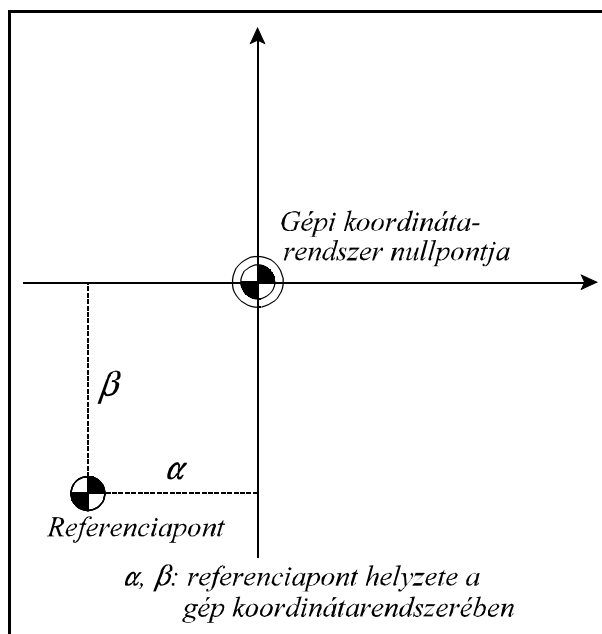
A vezérlés háromféle koordinátarendszert különböztet meg:

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

### 9.1 A gépi koordinátarendszer

A gép nullpontja, azaz a gépi koordinátarendszer nullpontja, egy olyan pont az adott szerzőgépen, 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.

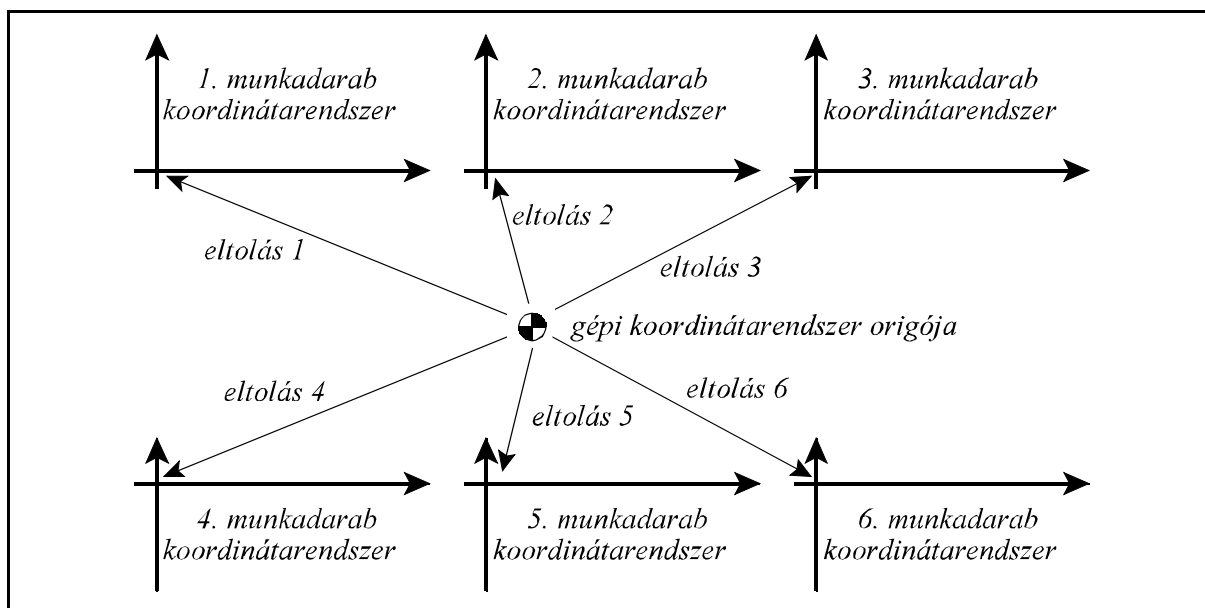
- G90, G91 állapottól függetlenül a v koordinátákat mindig abszolút koordinátaként kezeli,
- I operátor a koordináták címe után hatástalan,
- a mozgás mindig gyorsmenettel történik G00 utasításhoz hasonlóan,
- a pozicionálás mindig a beállított szerszámkorrekciók figyelembe vételével történik.

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.

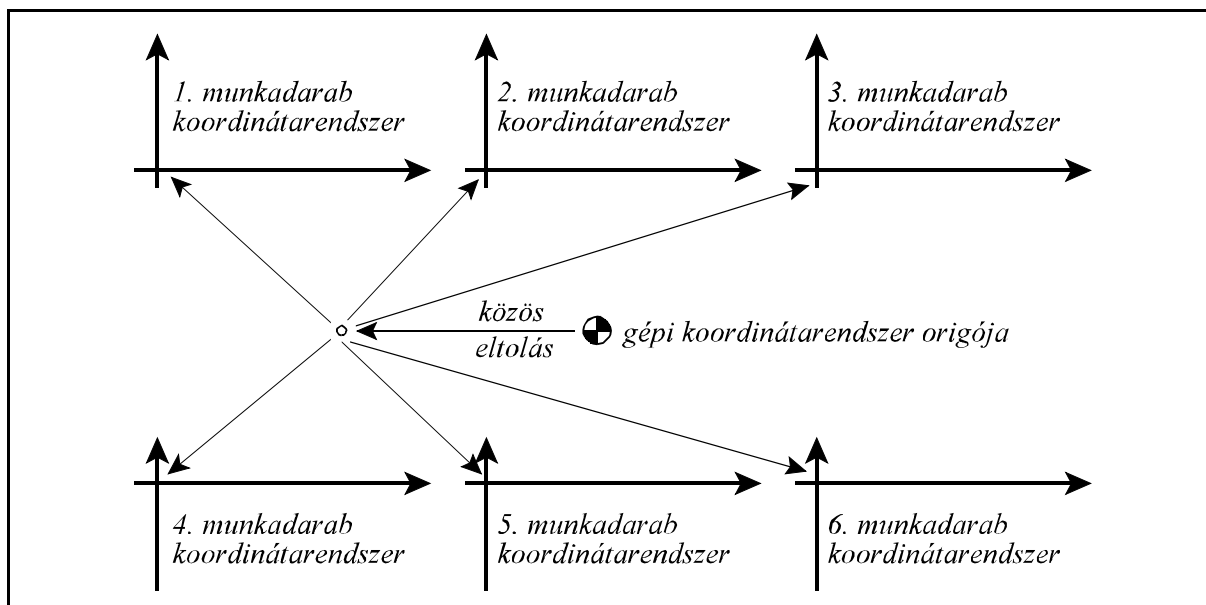
### 9.2.1 A munkadarab koordinátarendszerek beállítása



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

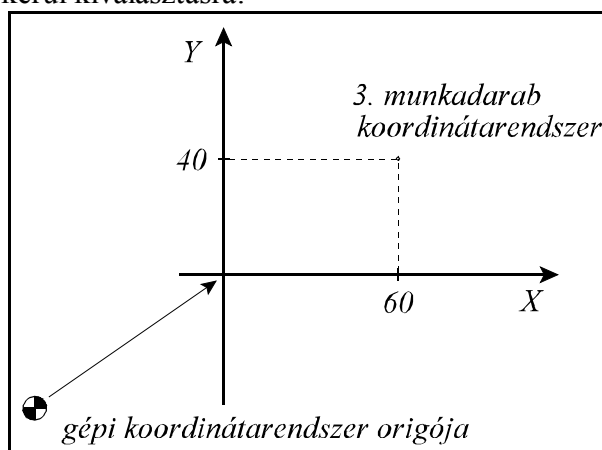
Öröklődő funkciók. Referenciapontfelvétel előtt választásuk hatástalan. Referenciapontfelvétel után a G54 1. munkadarab koordinátarendszer kerül kiválasztásra.

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

Például a

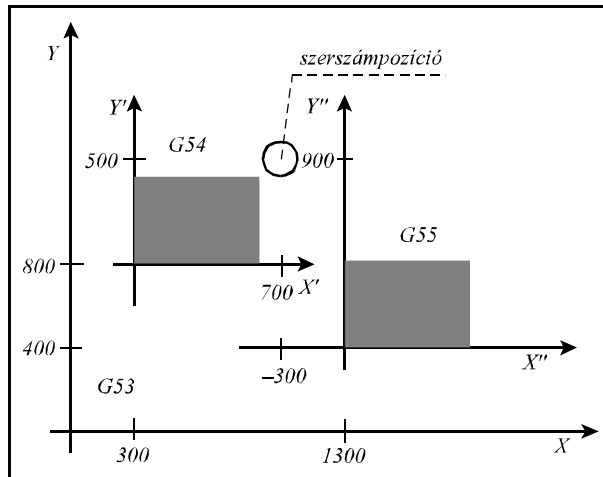
G56 G90 G00 X60 Y40

utasítás esetén a 3. munkadarab koordinátarendszer X=60, Y=40 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 asztalon két munkadarab helyezkedik el. Az egyik vonatkoztatási pontjához az első, G54 munkadarab koordinátarendszert állítottuk, amelynek az eltolása a gép koordinátarendszerében számítva  $X=300$ ,  $Y=800$ . A másik vonatkoztatási pontjához a második, G55 munkadarab koordinátarendszert állítottuk, amelynek az eltolása a gép koordinátarendszerében számítva  $X=1300$ ,  $Y=400$ . A G54  $X'$ ,  $Y'$  koordinátarendszerben a szerszám pozíciója  $X'=700$ ,  $Y'=500$ . A G55 utasítás hatására a szerszám pozíciója az  $X''$ ,  $Y''$  koordinátarendszerben kerül értelmezésre:  $X''=-300$ ,  $Y''=900$ .



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 Pp

utasítással, ahol

$p = 0$  közös eltolás állítása

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

$v$  (X, Y, Z, ...): tengelyenkénti eltolási érték

A koordinátaadatok mindig derékszögű, abszolút értéként kerülnek beolvasásra. G10 utasítás egylövétű.

### 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 értéként kerülnek értelmezésre.

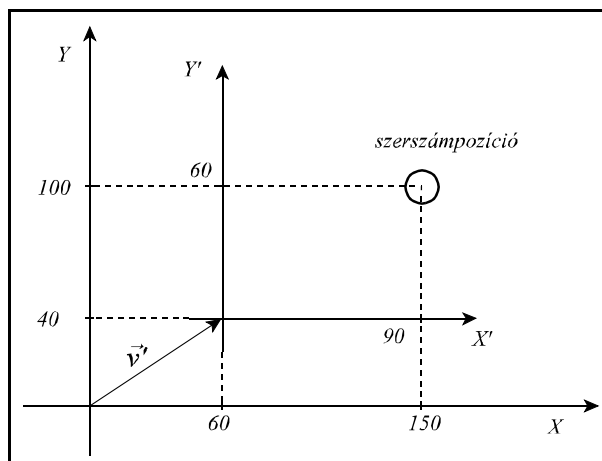
Ha például a szerszám az  $X=150$ ,  $Y=100$  koordinátájú ponton tartózkodik az aktuális  $X$ ,  $Y$  munkadarab koordináta-rendszerben, a

$G92 X90 Y60$

utasítás hatására egy új  $X'$ ,  $Y'$  koordináta-rendszer képződik, amelyben a szerszám az  $X'=90$ ,  $Y'=60$  koordinátájú pontra kerül. Az  $X$ ,  $Y$ - $X'$ ,  $Y'$  koordináta-rendszerek közötti  $V'$  eltolásvektor tengelyirányú komponensei:

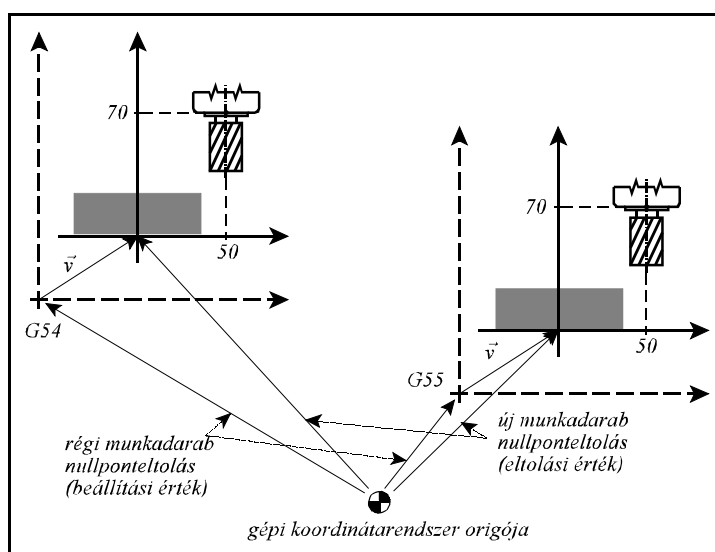
$$V'_x = 150 - 90 = 60, \text{ illetve}$$

$$V'_y = 100 - 60 = 40.$$



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.



9.2.4-2 ábra

#### ☞ Megjegyzések:

- A  $G92$  utasítással beállított munkadarab koordináta-rendszer eltolását az  $M2$ ,  $M30$  programvége utasítás végrehajtása, és a program elejére való reszettelés törli.
- $G92$  utasítás azokon a tengelyeken, amelyek az utasításban szerepelnek, törli a lokális koordináta-rendszer  $G52$  utasítással programozott eltolásait.
- A  $G92$  utasítással kényelmesen megoldhatjuk a több fordulatot is elmozduló körasztal ciklikus pozíciókijelzését. Ha például a  $B$  tengelyt elforgattuk a  $360^\circ$ -os pozícióba, akkor a  $G92 B0$  programozásával fizikai tengelymozgatás nélkül a  $0^\circ$ -os pozícióba vihető a tengely.

### 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 koordinátarendszer v koordinátájú pontjára esik.

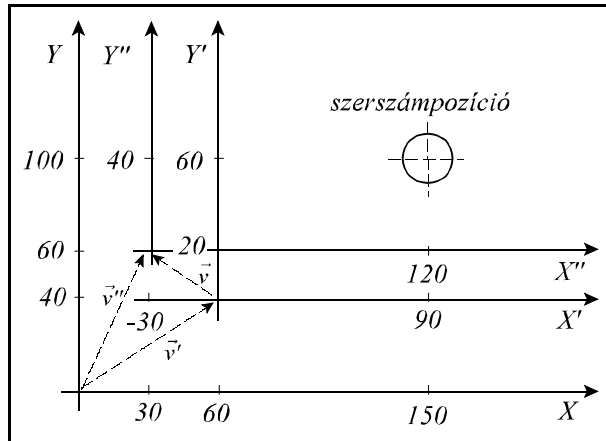
- Ha v koordináta inkrementális értéként van megadva a lokális koordinátarendszer origóját v vel eltolja, ha már előzőleg volt lokális koordinátarendszer definiálva, illetve ha nem, akkor az eltolás a munkadarab koordinátarendszer origójához képest képződik.

Ettől kezdve minden abszolút koordinátákkal megadott mozgásparancs az új koordinátarendszerben kerül végrehajtásra. A pozíciókijelzés is az új koordinátarendszerben történik. v koordináták értékeit mindig derékszögű koordinátaértékként kezeli.

Ha például a szerszám az X=150, Y=100 koordinátájú ponton tartózkodik az aktuális X, Y munkadarab koordinátarendszerben, a

```
G90 G52 X60 Y40
```

utasítás hatására egy új X', Y' lokális koordinátarendszer képződik, amelyben a szerszám az X'=90, Y'=60 koordinátájú pontra kerül. Az X, Y-X', Y' koordinátarendszerek közötti V' eltolásvektor tengelyirányú komponenseit határozzuk meg a G52 utasítással: V'<sub>x</sub>=60, illetve V'<sub>y</sub>=40.



Ha most a lokális koordinátarendszert az X'', Y'' pozíciójú pontra akarjuk vinni az eljárás 9.3-1 ábra kétféle lehet:

- abszolút adatmegadással: (G90) G52 X30 Y60 utasítás az X, Y munkadarab koordinátarendszerben az X'', Y'' koordinátarendszer origóját az X=30, Y=60 koordinátájú pontra állítja. V'' vektor komponensei V''<sub>x</sub>=30, V''<sub>y</sub>=60 értékadással képződnek.
- inkrementális adatmegadással: G91 G52 X-30 Y20 utasítás az X', Y' koordinátarendszerben az X'', Y'' koordinátarendszer origóját az X'=-30, Y'=20 koordinátájú pontra állítja. V vektor komponensei V<sub>x</sub>=-30, V<sub>y</sub>=20 értékadással képződnek. A V vektor, amely az új lokális koordinátarendszer helyzetét mutatja az X, Y munkadarab koordinátarendszerben: V''=V'+V. Ennek komponensei: V''<sub>x</sub>=60+(-30)=30, V''<sub>y</sub>=40+20=60.

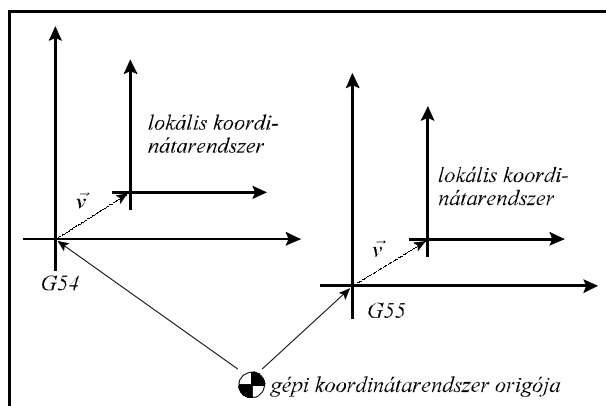
A szerszám pozíciója az X'', Y'' koordinátarendszerben: X''=120, Y''=40.

A

```
G90 G52 v0
```

utasítás törli a v koordinátájú pontokon az eltolásokat.

A lokális koordinátarendszer eltolása az összes munkadarab koordinátarendszerben érvényesül.



9.3-2 ábra

A G92 utasítás programozása azokon a tengelyeken, amelyeknek értéket adtunk törli a G52

utasítás által képzett eltolásokat, mintha G52 v0 parancsot adtunk volna ki.

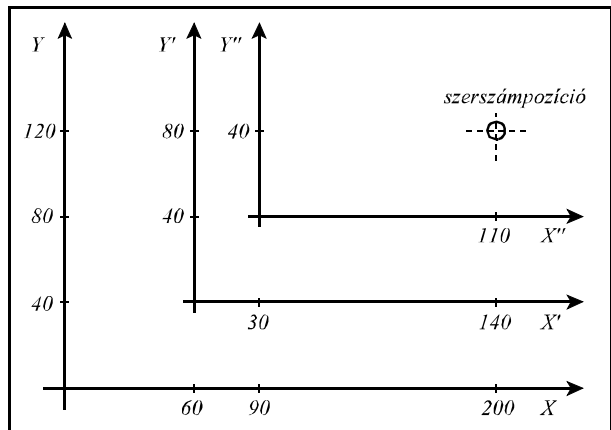
Ha a szerszám az X, Y munkadarab koordináta-rendszer X=200, Y=120 koordinátájú pontján tartózkodik, a

```
G52 X60 Y40
```

utasítás hatására az X', Y' lokális koordináta-rendszerben a pozíciója X'=140, Y'=80 lesz. Ezután a

```
G92 X110 Y40
```

parancs hatására az X'', Y'' új munkadarab koordináta-rendszerben a szerszám pozíciója X''=110, Y''=40 lesz. Tehát az X', Y' lokális koordináta-rendszer G92 parancs hatására törlődik, mintha G52 X0 Y0 parancsot adtunk volna ki.



9.3-3 ábra

**Megjegyzés:**

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

**9.4 Síkválasztás (G17, G18, G19)**

A sík, amelyben

- a körinterpoláció,
- a polárkoordinátás adatmegadás,
- a koordináta-rendszer elforgatás,
- a síkbeli szerszámsugár korrekció,
- a fűróciklusok pozícionálása

érvényesül a következő G kódokkal választható ki:

**G17**.....X<sub>p</sub>Y<sub>p</sub> sík

**G18**.....Z<sub>p</sub>X<sub>p</sub> sík

**G19**.....Y<sub>p</sub>Z<sub>p</sub> sík,

ahol: X<sub>p</sub>: X, vagy a vele párhuzamos tengely,  
 Y<sub>p</sub>: Y, vagy a vele párhuzamos tengely,  
 Z<sub>p</sub>: 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,

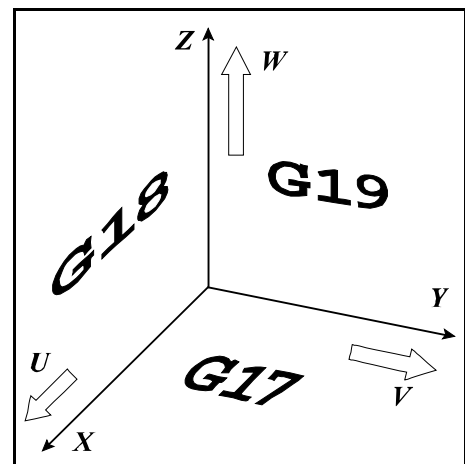
G18 X\_\_\_ W\_\_\_ az XW síkot,

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:



9.4-1 ábra

G17 X\_\_\_ Y\_\_\_ XY sík  
U\_\_\_ Y\_\_\_ marad az XY sík.

Ha a G17, G18, G19 mondatban nincs tengelycím megadva, akkor a fő tengelyeket 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 paramétermező *CODES* paraméterén megadott sík: G17, vagy G18 kerül kiválasztásra.

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 szerszámgép felépítésének függvényében értelmezheti kódként, vagy fordulatszám/perc dimenziójú értéként is.

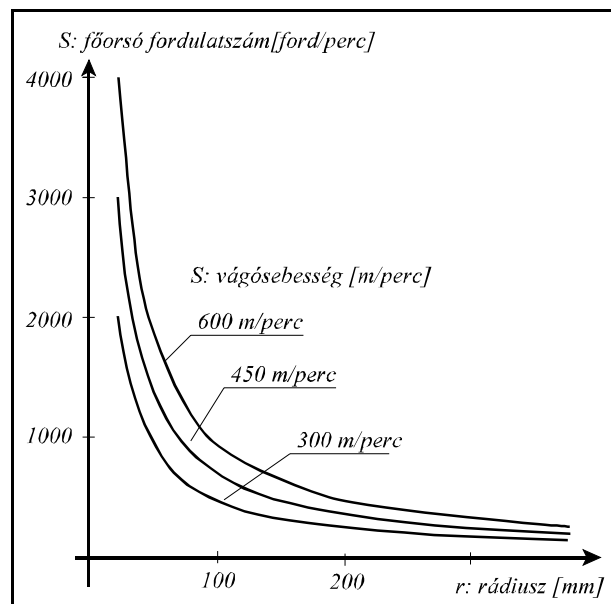
Ha mozgásparancsot és főorsó fordulatszámot (S) programozunk ugyanabba a mondatba az S funkció a mozgásparancs végrehajtása alatt, vagy után kerül végrehajtásra. A végrehajtás mikéntjét a gép építője határozza meg.

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 szerszámgép 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 számítását. 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 számítását 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 határoztuk 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, vagy a vezérlést nem kapcsoljuk ki.



### 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 nevezzük főorsó pozicionálásnak, vagy indexálásnak.

A pozicionálás 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 pozicionálni 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 nevezzük 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 *ORIENT1* 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 megkeresi a főorsó jeladó nullimpulzusát. Ezután 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 átírással került paraméterek megőrződnek kikapcsolásra is. A

#### G26 Pp Qq Rr Dd

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

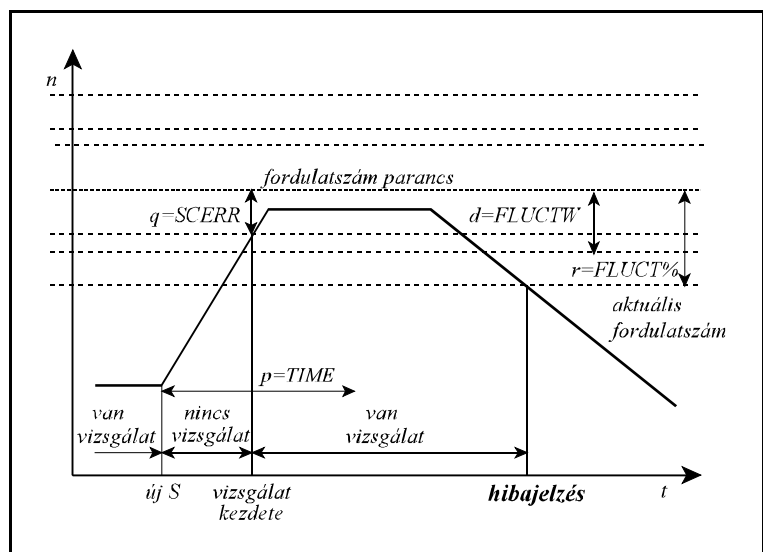
név	paraméter	jelentése	egység	értékhatar
p	5001 TIME	a fordulatszám parancs kiadása és az ellenőrzés megkezdése közötti maximális időtartam	100 msec	65535
q	5002 SCERR	a fordulatszám parancs és a tényleges fordulatszám között megengedhető százalékos eltérés	%	1-50
r	5003 FLUCT%	a fordulatszám ingadozás 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

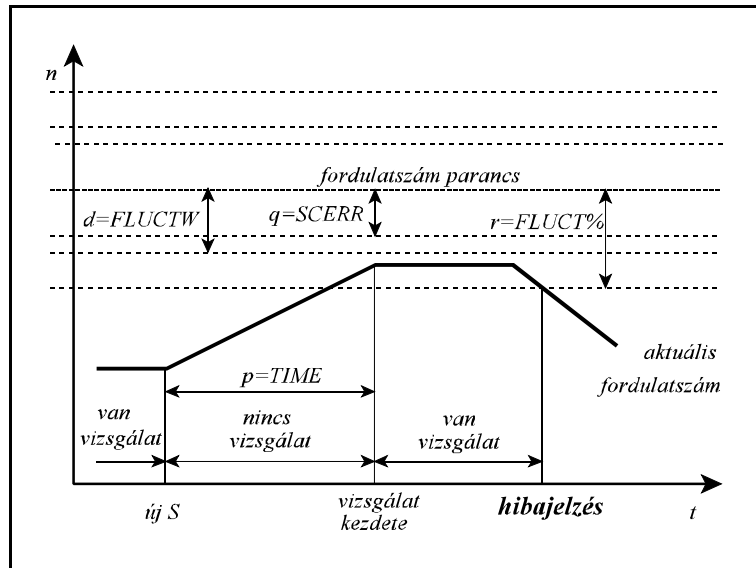
Új fordulatszám parancs hatására a figyelést felfüggeszti a vezérlő. A fordulatszám ingadozás figyelése akkor kezdődik, amikor

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



**10.6-1** ábra

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



10.6-2 ábra

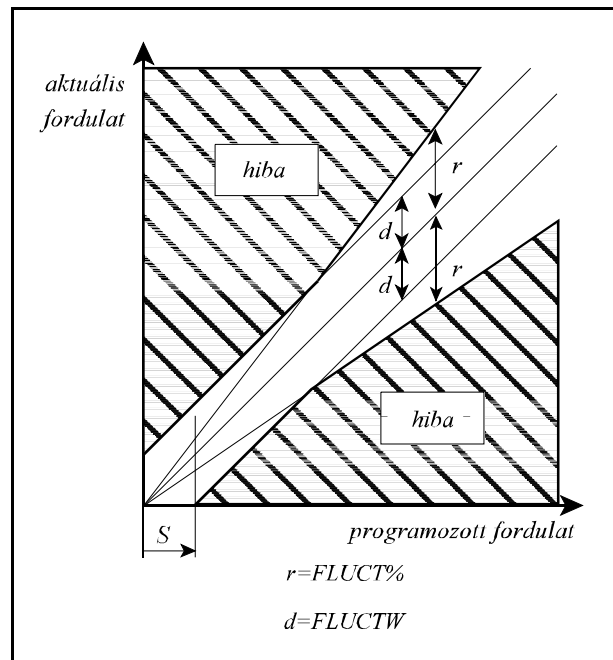
### Hiba detektálása

A figyelés során a vezérlő akkor ad hibajelzést, amikor az aktuális fordulatszám eltérése a fordulatszám parancsától 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 üzen a PLC-nek. A 3. ábrán látható az a fordulatszám tartomány, amelyre az NC hibajelzést küld. Ha a programozott főorsó fordulatszám 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 határos, ha a főorsóra jeladó van szerelve.
- A fordulatszám parancs, amihez képest az aktuális fordulatszámot figyeli, az override, a tartományi fordulatszámhatárok, és G96 állandó vágósebesség-számítási 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) határos.
- A G26 parancsot önálló mondatban kell programozni.



10.6-3 ábra

## 11 A szerszámkezelés

### 11.1 Szerszámhívási parancs (T kód)

T címre egy legfeljebb négyjegyű számot írva az NC egy kódot ad át a PLC-nek.

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.

### 11.2 Programformátum szerszámszám programozására

Az alkatrészprogramban alapvetően kétféle módon lehet a szerszámváltásra hivatkozni. A kétféle módszer a szerszámgép kiépítésétől függ. Az alkatrészprogramban alkalmazható szerszámhívási technikát a szerszámgép építője adja meg.

#### A. eset

A szerszámcsere a gépen kézzel, vagy revolver típusú szerszámváltóval történik. Ekkor T kódra történő hivatkozáskor:

- kézi csere esetén a kijelzőn megjelenik a beváltandó szerszám száma, amit kézzel kell a főorsóba behelyezni, majd start hatására a megmunkálás folytatódik,
- revolver típusú szerszámváltó esetén T kód hatására a szerszámot automatikusan beváltja a gép. Technikailag tehát a szerszámszámra történő hivatkozás azonnali cserét vált ki abban a mondatban, amelyikben T-t megadtuk.

#### B. eset

A szerszámcsere a gépen előkészítést igényel. Ezek lépései a következők:

- A szerszámtárban a beváltandó szerszámot meg kell keresni. Ekkor az alkatrészprogramban T címre történő hivatkozás a megfelelő szerszámot cserehelyzetbe hozza. Ez a művelet a háttérben folyik párhuzamosan a megmunkálással.
- A szerszámokat, vagy csak az egyiket, cserepozícióba kell küldeni.
- A szerszámcsere végrehajtása M06 funkcióval történik a programban. A szerszámcsere végrehajtására addig vár a vezérlő amíg az előkészítés alatt levő T szerszám cserehelyzetbe nem kerül. Ennek hatására az új szerszámot beteszi a főorsóba. Innen folytatódhat a forgácsolás.
- A régi szerszámot visszateszi a szerszámtárba. Ez a tevékenység a háttérben zajlik, a forgácsolással párhuzamosan.
- Elkezd keresni a szerszámtárban az új szerszámot.

Az alkatrészprogramban ennek a folyamatnak a leírása a következőképp történik:

alkatrészprogram	magyarázat
.....	
...Tnnnn.....	a Tnnnn számú szerszám keresése,
.....	fut az alkatrészprogram, háttérben zajlik a Tnnnn szerszám keresése,
...M06 Tmmmm....	Tnnnn szerszámot beteszi a főorsóba,
.....	az előző szerszámot visszateszi a szerszámtárba,
.....	elkezdni keresni a Tmmmm szerszámot, közben folyik a forgácsolás.
...M06 Tpppp.....	Tmmmm szerszámot beteszi a főorsóba,
.....	Tnnnn szerszámot visszateszi a szerszámtárba, elkezdni keresni a Tpppp szerszámot,
.....	közben folyik a megmunkálás.

## 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

**M06:** szerszámváltás kódja

**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 M06: szerszámváltás
2. csoport M11, ..., M18: főorsó tartományváltás
3. csoport M03, M04, M05, M19: főorsó kezelés
4. csoport M07, M08, M09: hűtővíz kezelés
5. csoport Mnnn: tetszőleges egyéb M funkció
6. csoport főorsó indexálás M kódjai
7. 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** címeken legfeljebb három számjegyet adhatunk meg, ha ezeknek a címeknek valamilyike, 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 a mozgásparancs végrehajtásával párhuzamosan, vagy a mozgásparancs végrehajtása után kerül végrehajtásra.

A végrehajtás sorrendjét a szerszámgép építője dönti el, és a szerszámgép 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.  | szerszámváltás:         | M06                                    |
| 2.  | szerszámhívás:          | T                                      |
| 3.  | főorsó tartományváltás: | M11, ..., M18                          |
| 4.  | főorsó fordulatszám:    | S                                      |
| 5.  | főorsó kezelés:         | M03, M04, M05, M19                     |
| 6.  | hűtővíz:                | M07, M08, M09                          |
| 7.  | egyéb M funkció:        | Mnnn                                   |
| 8.  | főorsó indexálás:       | M funkcióval                           |
| 9.  | A funkció:              | A                                      |
| 10. | B funkció:              | B                                      |
| 11. | C funkció:              | C                                      |
| 12. | 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 sorszámokat jelentenek. Az 1-es sorszámú **FELTÉTELES MONDAT** kapcsoló a vezérlő kezelőpaneljén található.

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.

### 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 mondattaknak.

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:

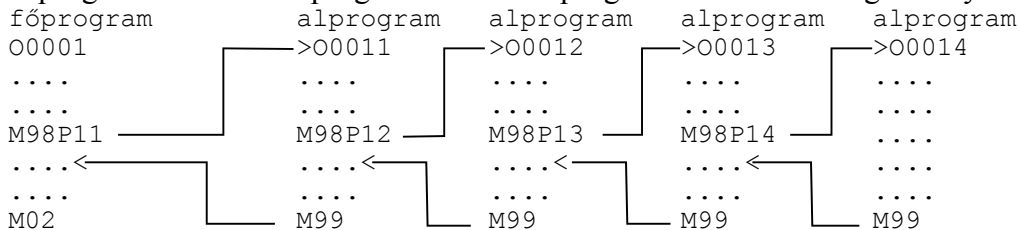
hívó program		alprogram	megjegyzés
O0010			az O0010 program végrehajtása
.....			
.....			
M98 P0011	--->	O0011	az O0011 alprogram hívása
		.....	az O0011 alprogram végrehajtása
		.....	
		.....	
következő mondat	<---	M99	visszatérés a hívó programba
.....			az O0010 program folytatása
.....			

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 hívd 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
O0010 ..... ..... .....			az O0010 program végrehajtása
N101 M98 P0011	--->	O0011 ..... ..... .....	az O0011 alprogram hívása az O0011 alprogram végrehajtása
N102 .....	<---	M99	visszatérés a hívó program következő mondatára az O0010 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
O0010 ..... ..... .....			az O0010 program végrehajtása
N101 M98 P0011	--->	O0011 ..... ..... .....	az O0011 alprogram hívása az O0011 alprogram végrehajtása
N250 .....	<---	M99 P250	visszatérés a hívó program N250 mondatára az O0010 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 O0011 alprogramot és onnan M99 L5 utasítással térünk vissza az O0011 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 eredményez:

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

The diagram shows a vertical list of lines: O0123, N1... followed by a left-pointing arrow, three dots, two lines of six dots, and M99. A bracket on the right side connects the M99 line back to the N1... line, indicating a jump.

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
```

The diagram shows a vertical list of lines: O0011, three dots, two lines of six dots, N128... followed by a left-pointing arrow, two lines of six dots, and M99 P128. A bracket on the right side connects the M99 P128 line back to the N128... line, indicating a jump.

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

The diagram shows a vertical list of lines: O0011, three dots, M99 P225, two lines of six dots, N225 followed by a left-pointing arrow, and three dots. A bracket on the right side connects the M99 P225 line back to the N225 line, indicating a jump.

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

### 14.1 Hivatkozás szerszámkorrekcióra (H és D)

A szerszámhossz korrekciókra: **H** címen,  
 a szerszámsugár korrekciókra: **D** címen  
 tudunk hivatkozni. A cím utáni szám, a korrekció száma mutatja meg, hogy melyik korrekciós érték kerül le hívásra. H és D cím értékhatára: 0-999.  
 A korrekciós tár felosztását az alábbi táblázat mutatja:

korrekció-szám	H kód		D kód	
	geometriai érték	kopás érték	geometriai érték	kopás érték
01	-350.200	0.130	-32.120	0.012
02	830.500	-0.102	52.328	-0.008
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.

A 00 korrekciószám nem szerepel a táblázatban, az ezen lévő korrekciós értékek mindig nullák.

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

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

Ha programban H, vagy D címen egy korrekciós értékre hivatkozunk a vezérlés korrekció gyanánt mindig a geometriai-, és kopásérték összegét veszi figyelembe. Például, ha H2-re hivatkozunk a programban, akkor a fenti táblázat alapján a hosszkorrekció értéke a 02 sor szerint:  $830.500 + (-0.102) = 830.398$ .

H és D cím öröklődik, vagyis a vezérlés mindaddig ugyanazt a korrekciós értéket veszi figyelembe, amíg egy másik H vagy D parancsot nem kap, azaz amikor egy D vagy H parancsral a korrekciós értéket kiolvastuk, úgy a korrekciós táblázat módosítása (például G10 programozásával) már nincs hatással a kiolvasott értékre.

A korrekciós tár korrekciós értékei kikapcsolásra megőrződnek.

A korrekciós tárat mint alkatrész programot a programtárba is elmenthetjük.

A geometriai és kopás értékek értékhatárai:

bemeneti mérték-rendszer	kimeneti mérték-rendszer	ink-remens-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 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 ismételten hivatkoznunk kell az aktuális D illetve H korrekciós regiszterre, mert csak ebben az esetben kerül figyelembevételre a módosított érték. Egy adott vezérlésben H és D cím értékhatárát, azaz a vezérlésben megadható hossz-, és sugárkorrekciók számát az adott vezérlés tárkiépítése határozza meg. Minimális tárkiépítés esetén a korrekciók száma 99, azaz H és D cím értékhatára: 0-99.

## 14.2 Szerszámkorrekciós értékek módosítása programból (G10)

A

### G10 R L P

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

R címen adjuk meg a korrekció értékét. G90 abszolút adatmegadás parancsállapotban az R címre írt érték kerül a megfelelő korrekciós regiszterbe. G91 inkrementális adatmegadás parancsállapotban, vagy I operátor használata esetén az R címre írt adat hozzáadódik a megfelelő korrekciós regiszter tartalmához.

L címen adjuk meg, hogy milyen korrekciós értéket kívánunk módosítani:

L=10 jelentése: a beállítás a hosszkorrekció (H kód) geometriai értékére vonatkozik,

L=11 jelentése: a beállítás a hosszkorrekció (H kód) kopásértékére vonatkozik,

L=12 jelentése: a beállítás a sugárkorrekció (D kód) geometriai értékére vonatkozik,

L=13 jelentése: a beállítás a sugárkorrekció (D kód) kopásértékére vonatkozik.

P címen adjuk meg, hogy milyen számú korrekciós értéket akarunk módosítani.

☞ *Megjegyzés:* a szerszámsugár korrekció programozott módosításakor az R címen megadott értéket minden esetben sugárként kell értelmezni, függetlenül a *TOOLRAD* paraméter állapotától. A vezérlés *3001 ÉRTÉKHATÁR X,Y,...F* üzenetet ad ha a megadott értékek túllépik a fenti táblázatban megadott értékhatárokat.

## 14.3 A szerszámhossz-korrekció (G43, G44, G49)

A

**G43** q H, vagy a**G44** q H

utasítás bekapcsolja a szerszámhossz–korrekciós üzemmódot.

q cím jelentése: a szerszámhossz–korrekció a q tengelyen érvényesül. (q: X, Y, Z, U, V, W, A, B, C)

H cím jelentése: a szerszámhossz–korrekció értékét az ezen a címen megadott korrekciós rekeszből veszi.

G43 utasítás a végrehajtás során akár abszolút, akár inkrementális adat q, a kiadódó végponti koordinátához hozzáadja a H címen megadott korrekciós értéket:

**G43: + korrekció**

G44 utasítás a végrehajtás során akár abszolút, akár inkrementális adat q, a kiadódó végponti koordinátából levonja a H címen megadott korrekciós értéket:

**G44: – korrekció**

A G43 G91 Z0 H1, vagy a G44 G91 Z0 H1 utasítások, mivel Z-n 0 inkrementális elmozdulást programoztunk pont a szerszámhosszal megegyező méretű elmozdulást eredményeznek. G43 esetén pozitív irányban, ha a H1 alatti korrekciós érték pozitív, negatív irányban ha a H1 alatti korrekciós érték negatív. G44 estén pont fordítva. A parancs végrehajtása után Z koordinátán a kijelzett pozíció ugyanaz lesz, mint a parancs végrehajtása előtt, mert a hosszkorrekció bekapcsolása után a szerszám hegyének pozíciója kerül kijelzésre.

Szerszámhossz–korrekciót egyszerre több tengelyen is lehet definiálni. Például:

G43 Z250 H15

G43 W310 H16

Ha egy mondatban több tengelyt jelölünk ki mindegyik kijelölt tengelyen figyelembe veszi a szerszámhossz–korrekciót:

G44 X120 Z250 H27

Ha a korrekciós értéket új H cím lehívásával megváltoztatjuk, a régi érték törlődik, és az új érték kerül érvényesítésre:

H1=10, H2=20

G90 G00

G43 Z100 H1.....a Z=110 pontra mozog

G43 Z100 H2.....a Z=120 pontra mozog

G43 és G44 hatása öröklődik egészen addig, amíg ebből a csoportból egy másik parancsot nem kap.

A

**G49** vagy a**H00**

parancs az összes tengelyen kikapcsolja a szerszámhossz–korrekciót, mozgással, ha a mondatba mozgást is programoztunk, vagy transzformációval, ha a mondatba nincs mozgás programozva.

A két parancs közti különbség, hogy a H00 utasítás csak a korrekciót törli és a G43 vagy G44 állapotot változatlanul hagyja. Ha ezek után új, nullától különböző H címre történik hivatkozás a G43 vagy a G44 állapot függvényében az új szerszámhossz–korrekció bekapcsolódik.

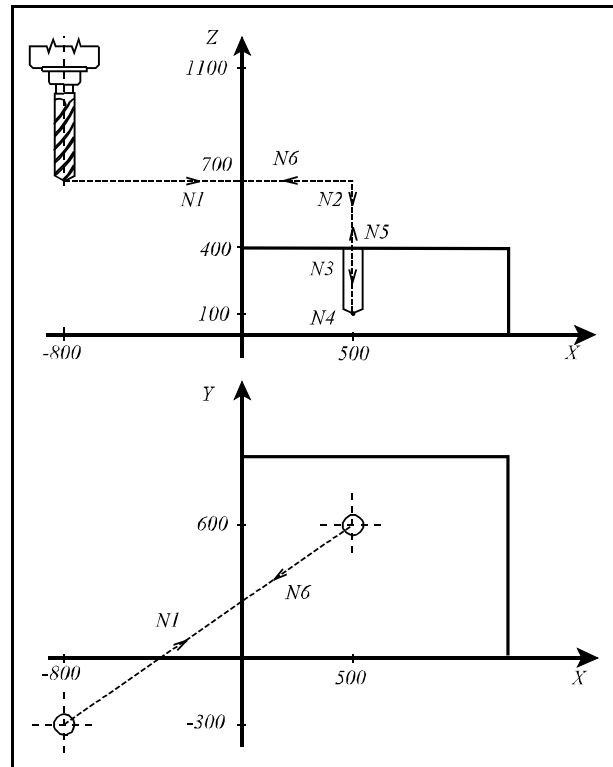


Ha viszont G49 utasítást használunk, utána addig elsül a levegőben minden H címre történő hivatkozás, amíg G43-at vagy G44-et nem programoztunk.

Bekapcsolásra a paramétermező *CODES* paramétercsoportján meghatározott érték dönti el, hogy melyik (G43, G44, G49) kód van érvényben.

Az alábbi mintapélda egy egyszerű fúrási műveletet mutat be a szerszámhossz-korrekció figyelembevételével:

a fúrószerszám hossza: H1=400



14.3-1 ábra

N1 G90 G0 X500 Y600	(X, Y síkban pozícióra áll)
N2 G43 Z410 H1	(Z410-re mozog H1 hosszkorrekcióval)
N3 G1 Z100 F180	(Z100-ig fúr F180 előtolással)
N4 G4 P2	(2 másodpercig vár)
N5 G0 Z1100 H0	(kiemeli a szerszámot a hosszkorrekció kikapcsolásával, szerszám hegye X700 ponton)
N6 X-800 Y-300	(X, Y síkban gyorsmenettel visszaáll)

#### 14.4 A szerszámeltolás (G45...G48)

**G45:** a korrekciós értékkel növeli az elmozdulást

**G46:** a korrekciós értékkel csökkenti az elmozdulást

**G47:** a korrekciós érték kétszeresével növeli az elmozdulást

**G48:** a korrekciós érték kétszeresével csökkenti az elmozdulást

A G45...G48 parancs a D kóddal kiválasztott korrekcióval hatásos, mindaddig amíg más értéket nem hívunk le G45...G48 parancs kíséretében.

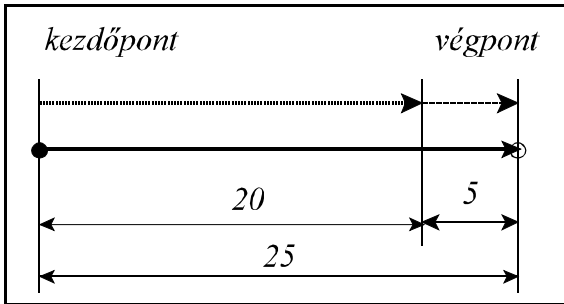
Nem öröklődő kódok, csak abban a mondatban érvényesek, amelyekben specifikálták őket.

Az elmozdulás abszolút adatmegadás esetén az aktuális mondatban definiált végpont és az előző mondat végpontja közötti különbség. A növelés illetve csökkentés az így képződő mozgásirányban értendő.

G45 programozása esetén:

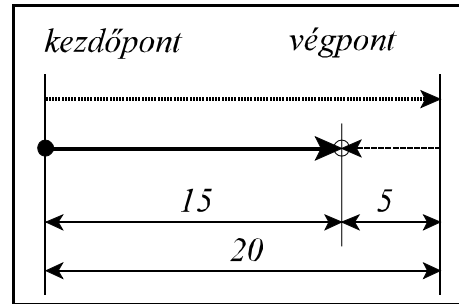
(a korrekciós értékkel növeli az elmozdulást)

- a. mozgásparancs: 20  
korrekciós érték: 5



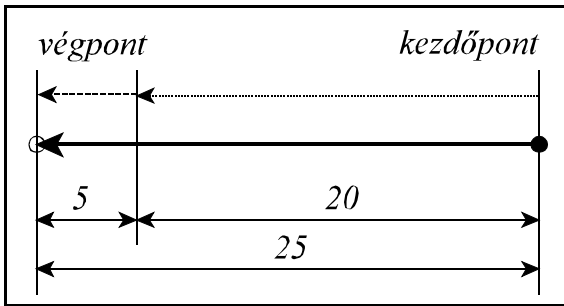
14.4-1 ábra

- b. mozgásparancs: 20  
korrekciós érték: -5



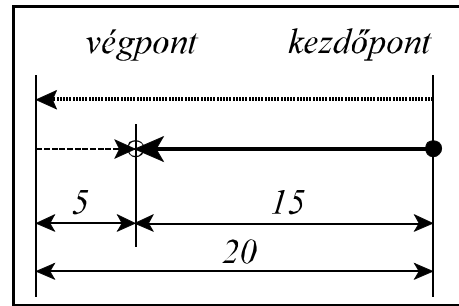
14.4-2 ábra

- c. mozgásparancs: -20  
korrekciós érték: 5



14.4-3 ábra

- d. mozgásparancs: -20  
korrekciós érték: -5

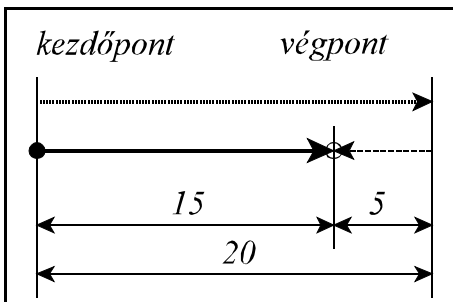


14.4-4 ábra

G46 programozása esetén:

(a korrekciós értékkel csökkenti az elmozdulást)

- a. mozgásparancs: 20  
korrekciós érték: 5



14.4-5 ábra

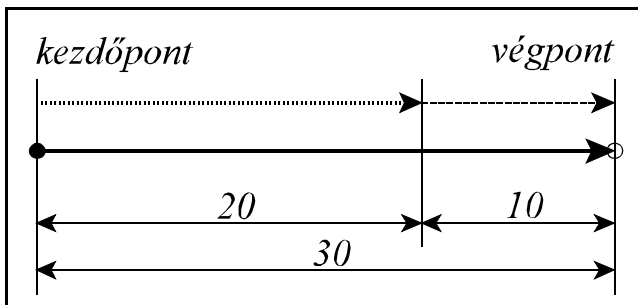
- b., c., d. eset G45-höz hasonlóan

G47 programozása esetén:

(a korrekciós érték kétszeresével növeli az elmozdulást)

- a. mozgásparancs: 20  
korrekciós érték: 5

b., c., d. eset G45-höz hasonlóan



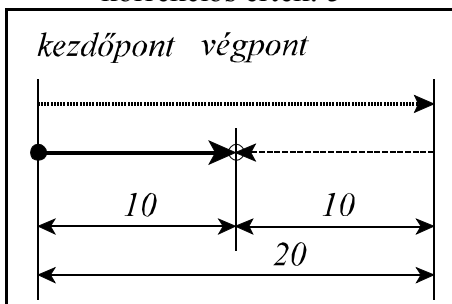
14.4-6 ábra

G48 programozása esetén:

(a korrekciós érték kétszeresével csökkenti az elmozdulást)

- a. mozgásparancs: 20  
korrekciós érték: 5

b., c., d. eset G45-höz hasonlóan



14.4-7 ábra

Ha a G45...G48 parancsot követően a mondatban egyszerre több tengelyre van kiadva mozgásparancs az így képződő korrekció az összes programozott tengelyen érvényesül, tengelyenként külön-külön, az érvényes D alatti értékkel. (Nem vektorosan képződik.)

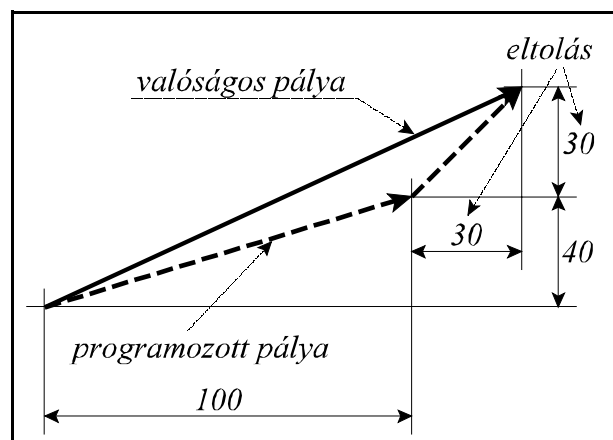
Például, ha  $D1=30$ , a  $G91 G45 G1 X100 Y40 D1$  parancsra az elmozdulásértékek:  $x=130$ ,  $y=70$ .

Az így képződő korrekciókat nem lehet törölni sem egy közös G parancsral, (mint például hosszkorrekció esetén G49), sem D00 programozásával, csak az ellenkező értelmű G45...G48 parancsral.

G45...G48 használata közben mindig csak ugyanazt a D kódot lehet használni, ellenkező esetben a vezérlés 3008 HIBÁS G45...G48 hibajelzést ad.

Abban az esetben, ha inkrementális 0 elmozdulást programozunk G45...G48 parancs kíséretében a 0 elé írt előjelet is értelmezi a vezérlés a következő módon:

Ha  $D1=12$ :



14.4-8 ábra

## 14 A szerszámkorrekció

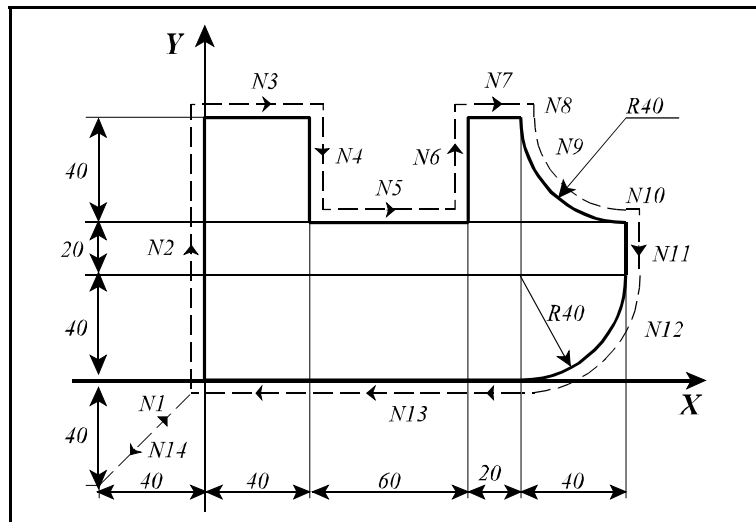
NC parancs	G45 XI0 D1	G46 XI0 D1	G45 XI-0 D1	G46 XI-0 D1
elmozdulás	x=12	x=-12	x=-12	x=12

A G45...G48 kódokkal alkalmazott szerszámugár korrekciót  $\frac{1}{4}$  és  $\frac{3}{4}$  körök esetén is lehet alkalmazni, ha a kör középpontokat I, J, vagy K címen adjuk meg.

Mintapélda: D1=10

```

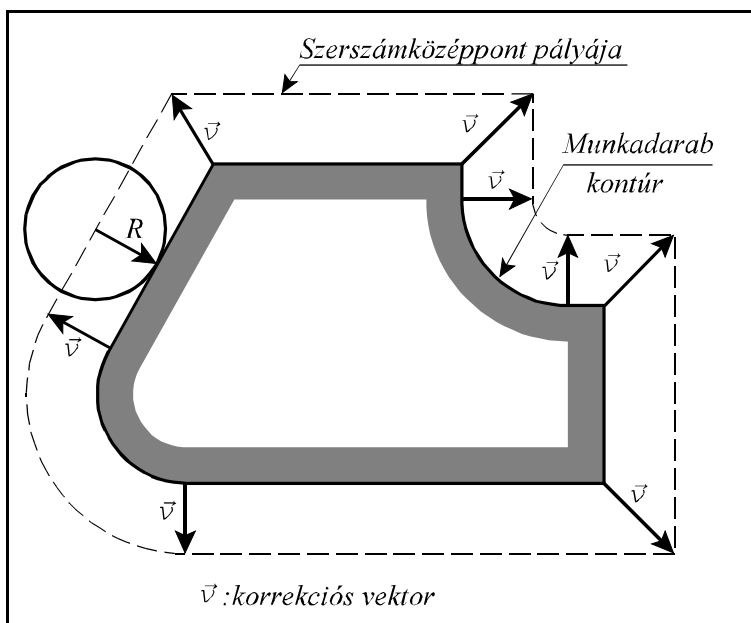
N1 G91 G46 G0 X40 Y40 D1
N2 G47 G1 Y100 F180
N3 G47 X40
N4 Y-40
N5 G48 X60
N6 Y40
N7 G47 X20
N8 G45 Y-0
N9 G46 G3 X40 Y-40 I40
N10 G45 G1 X0
N11 G45 Y-20
N12 G45 G2 X-40 Y-40 I-40
N13 G45 G1 X-120
N14 G46 G0 X-40 Y-40
    
```



14.4-9 ábra

### 14.5 A síkbeli szerszámsugár korrekció (G38, G39, G40, G41, G42)

Ahhoz, hogy egy síkbeli alakzatot körbe lehessen marni, és az alakzatnak a rajz szerinti pontjait kelljen a programban megadni, függetlenül az alkalmazott szerszám méretétől, a vezérlésnek a szerszám középpontját a programozott kontúrral párhuzamosan, attól szerszámsugárnyi távolságra kell vezetnie. A vezérlés a lehívott D korrekciószámon bejegyzett szerszámsugár korrekció értékének függvényében állapítja meg, hogy a szerszámközpont pályáját milyen távolságra vezesse a programozott kontúrtól.



14.5-1 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 D címen lehívott korrekciós értéktől és a két mondat közti átmenet geometriájától függ.

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

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

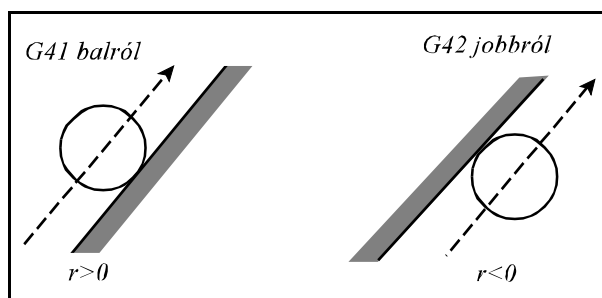
Abban az esetben, ha nem a fősíkra eső tengelyek mentén akarunk korrekciós síkot definiálni a melléktengelyeket a paramétermezőben párhuzamos tengelyekként kell definiálni. Például, ha U párhuzamos tengelynek van felvéve, és a Z, U, síkban akarjuk a szerszámsugár korrekciót alkalmazni G18 U\_\_Z\_\_ megadással lehet a síkot kijelölni.

**G40:** szerszámsugár korrekció kikapcsolása

**G41:** szerszámsugár korrekció balról

**G42:** szerszámsugár korrekció jobbról

A G41, vagy G42 parancs a korrekciószámítást bekapcsolja. G41 állapotban a programozott kontúrt a menetirány szerint balról, G42 állapotban pedig jobbról követi. Az alkalmazott szerszámsugár korrekciós értéket D címen kell megadni. D00 megadása mindig nulla sugárérték lehívásával egyenlő. A korrekciószámítás a G00, G01, G02, G03 interpolációs mozgásokra történik.



14.5-2 ábra

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: <b>pozitív</b>	sugárkorrekció: <b>negatív</b>
<b>G41</b>	balról	jobbról
<b>G42</b>	jobbról	balról

☞ *Megjegyzés:*

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

G40 vagy D00 parancs kikapcsolja a korrekciószámítást. A két parancs közti különbség, hogy a D00 utasítás csak a korrekciós vektor hosszát törli és a G41 vagy G42 állapotot változatlanul hagyja. Ha ezek után új, nullától különböző D címre történik hivatkozás a G41 vagy a G42 állapot függvényében az új szerszámsugárral a korrekciós vektor kiszámításra kerül.

Ha viszont G40 utasítást használunk, utána addig elsül a levegőben minden D címre történő hivatkozás, amíg G41-et vagy G42-t nem programoztunk.

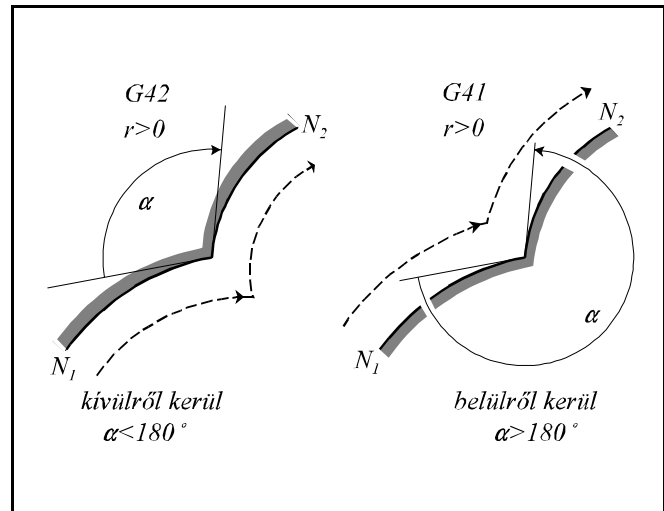
A sugárkorrekció bekapcsolásának, illetve kikapcsolásának, meghatározott szabályai vannak, amit a következő fejezetek tárgyalnak részletesen.

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 hatásos. 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:  $\alpha$ -t.  $\alpha$  iránya attól függ, hogy a kontúrt balról, vagy jobbról járjuk körül.

A vezérlés az  $\alpha$  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 kerüléshez.



14.5-3 ábra

### 14.5.1 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 szerszámközpont pályája egybeesik a programozott pályával.

G40 állapotból G41, vagy G42 utasítás hatására a vezérlő belép a sugárkorrekció számítási üzemmódba. A korrekció értékét a D címen megadott korrekciós rekeszből veszi. 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 D00–al töröljük a korrekciót és utána Dnn–nel visszakapcsoljuk (nn 0–tól különböző szám), nem a kontúrra való ráállítás stratégiáját választja a vezérlő.

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

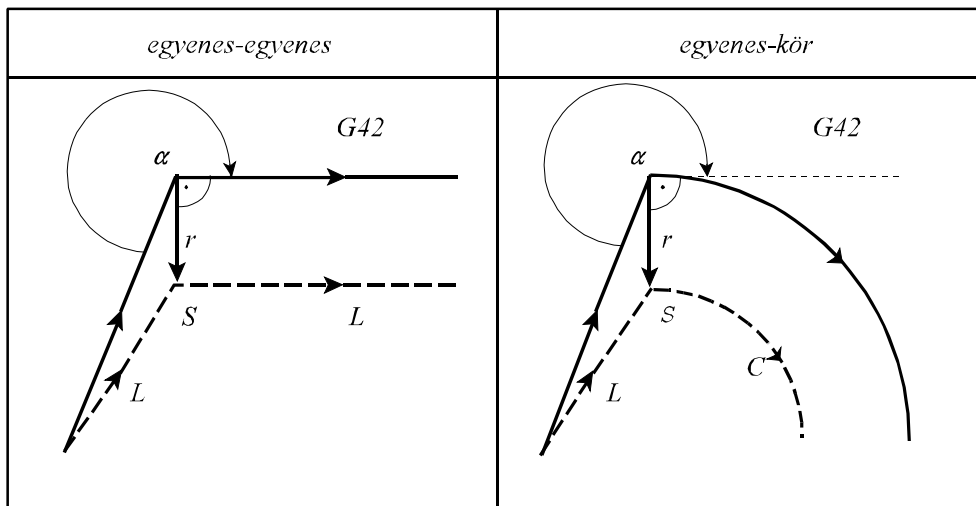
☞ *Megjegyzés:* Az ábrák jelöléseinek jelentése most, és a továbbiakban:

- r: a sugárkorrekció értéke,
- L: egyenes szakasz,
- C: körív,
- S: mondatonkénti üzemmódban a megállás helye,
- szaggatott vonal: a szerszámközpont pályája,
- folyamatos vonal: a programozott pálya.

#### A sugárkorrekció bekapcsolásának alapesetei:

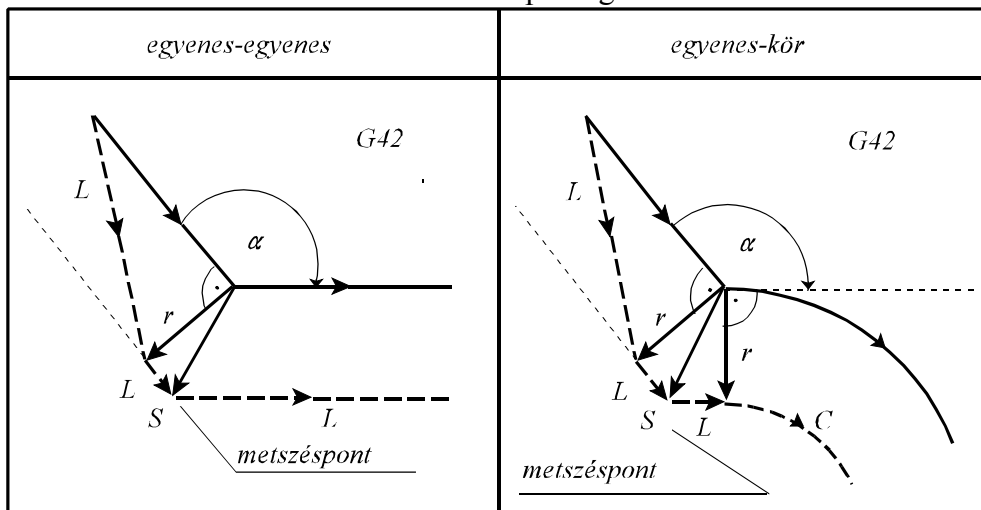
(G40)	(G40)
G42 G01 X_ Y_ D_	G42 G01 X_ Y_ D_
X_ Y_	G2 X_ Y_ R_

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

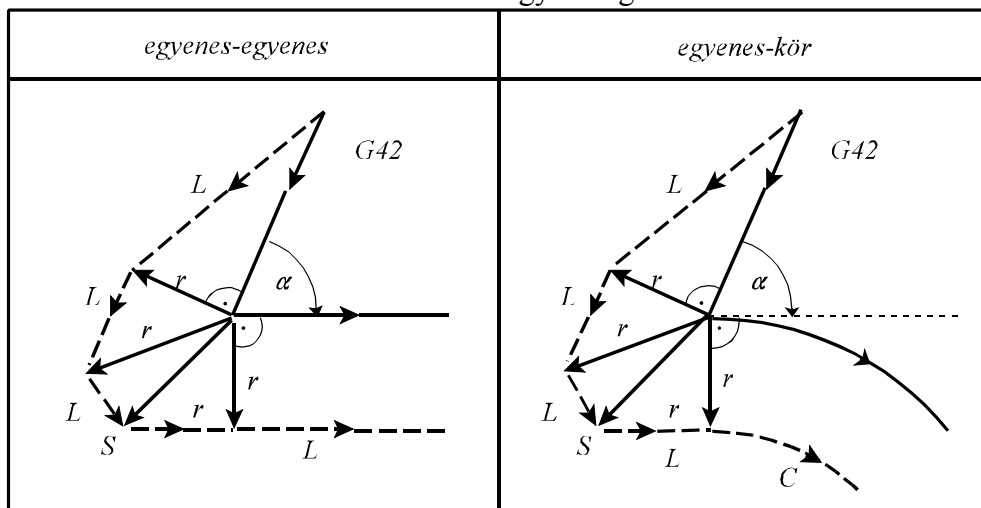


14.5.1-1 ábra



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

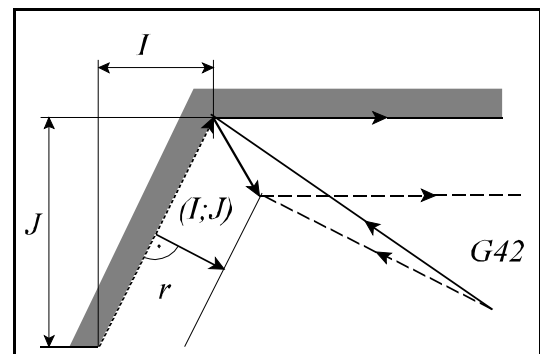
14.5.1-2 ábra

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

14.5.1-3 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: G17 esetén I, J-nek), 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, amelyben programoztuk. Ez a lehetőség például belső sarokra való ráállás esetén hasznos.



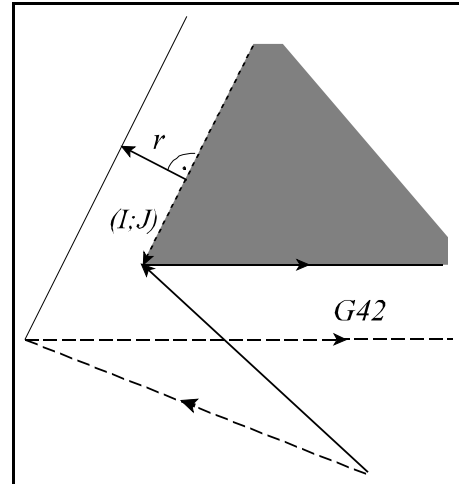
14.5.1-4 ábra

```

...
G91 G17 G40
...
N110 G42 G1 X-80 Y60 I50 J70 D1
N120 X100
...

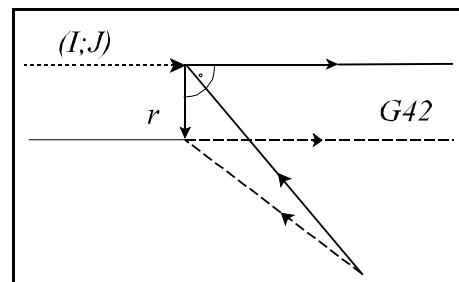
```

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



14.5.1-5 ábra

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



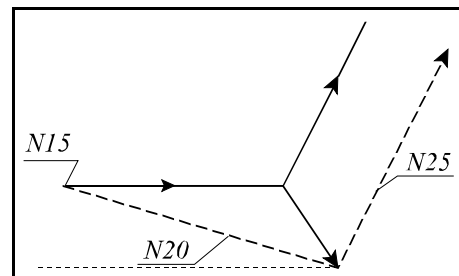
14.5.1-6 ábra

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) számítható ki:

```

...
N10 G40 G17 G0 X0 Y0
N15 G42 D1
N20 G1 X80
N25 X110 Y60
...

```



14.5.1-7 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.

```

...
N10 G40 G17 G0 X0 Y0
N15 G91 G42 D1 X0
N20 G1 X80
N25 X30 Y60
...

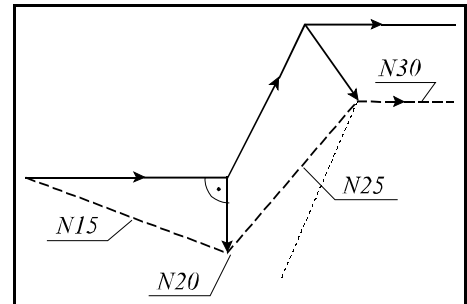
```

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 G17 G0 X0 Y0
N15 G91 G42 D1 X80
N20 G1 X0
N25 X30 Y60
N30 X60
...

```



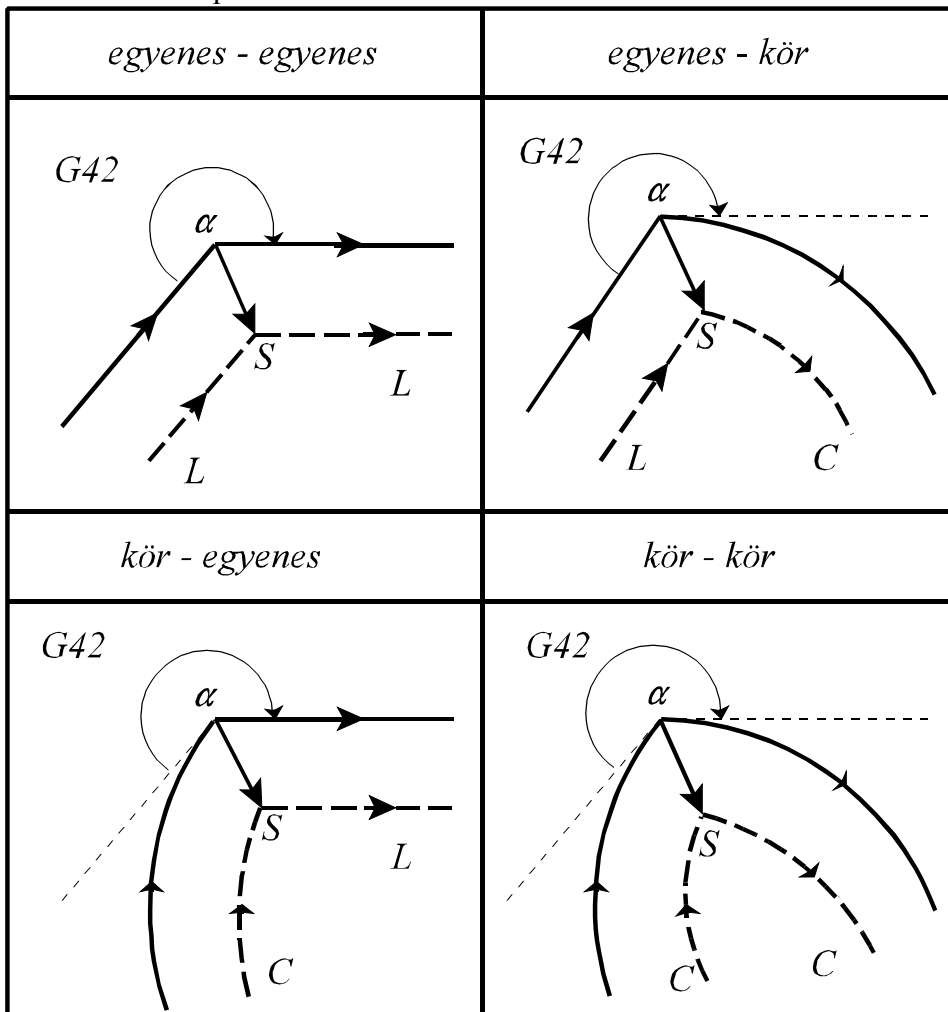
14.5.1-8 ábra

**14.5.2 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.

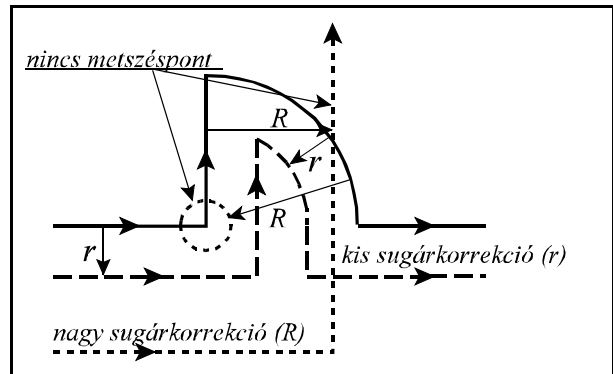
A sugárkorrekció számítás bekapcsolt állapotának alapesetei:

Metszéspontszámítás belső sarkok esetén:  $180^\circ < \alpha < 360^\circ$



14.5.2-1 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.



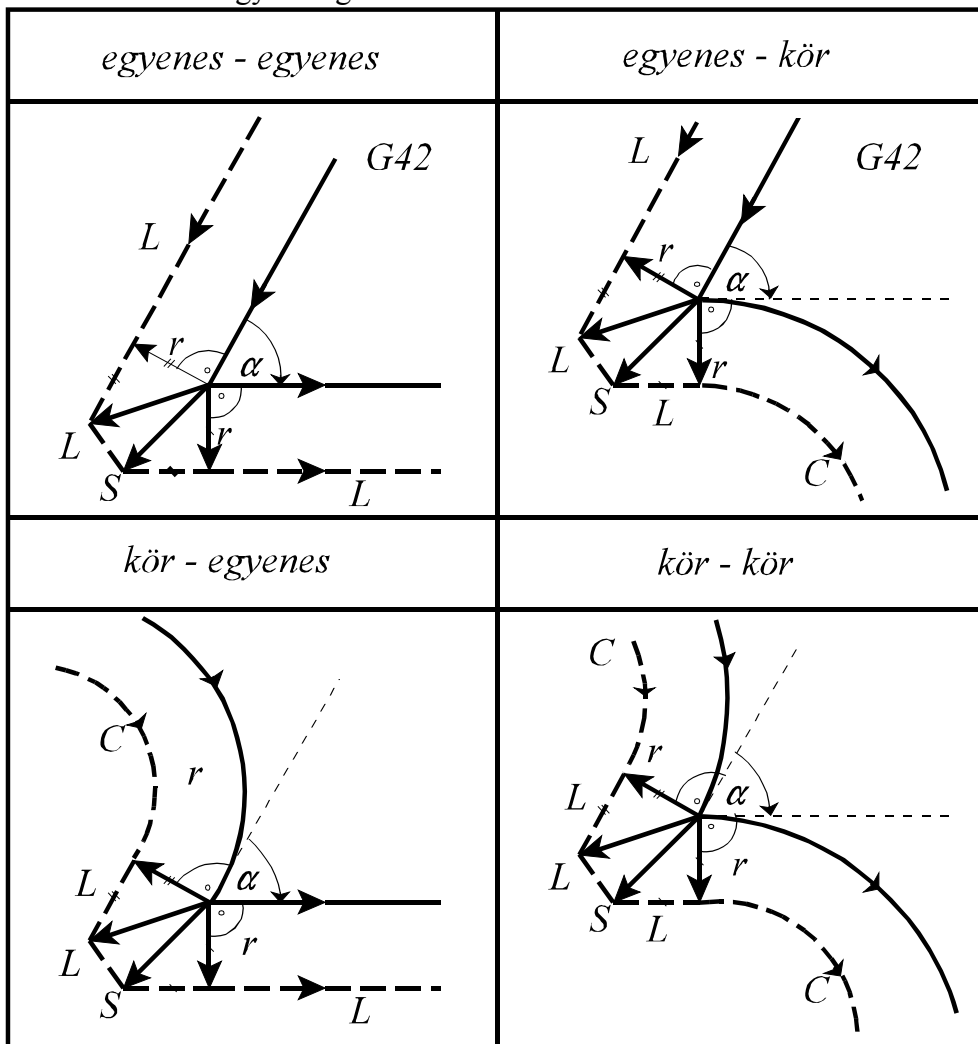
14.5.2-2 ábra

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

<i>egyenes - egyenes</i>	<i>egyenes - kör</i>
<p style="text-align: center;">G42</p>	<p style="text-align: center;">G42</p>
<i>kör - egyenes</i>	<i>kör - kör</i>
<p style="text-align: center;">G42</p>	<p style="text-align: center;">G42</p>

14.5.2-3 ábra

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



14.5.2-4 ábra

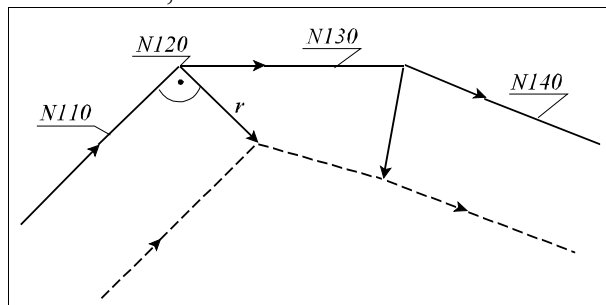
A sugárkorrekció számítás bekapcsolt állapotának speciális esetei:

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 G17 G42...
N110 G1 X40 Y50
N120 X0
N130 X90
N140 X50 Y-20
...
    
```



14.5.2-5 ábra

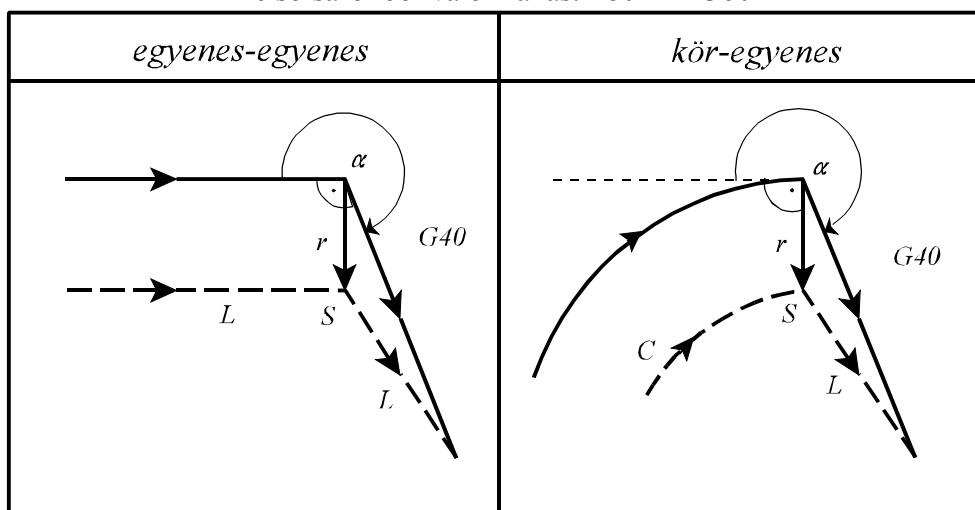
### 14.5.3 A szerszámsugár korrekciószámítás kikapcsolása. Leállítás a kontúrról.

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:

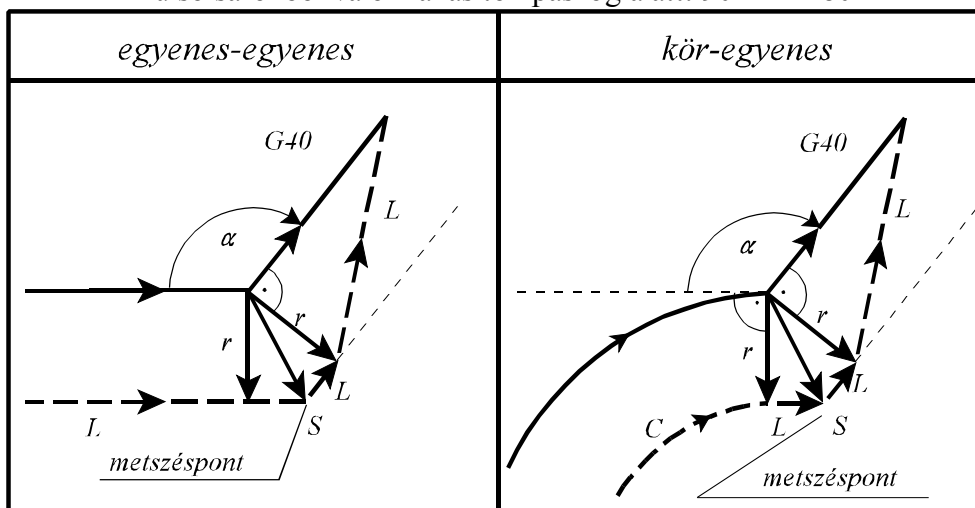
(G42)	(G42)
G01 X_ Y_	G02 X_ Y_ R_
G40 X_ Y_	G40 G1 X_ Y_

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



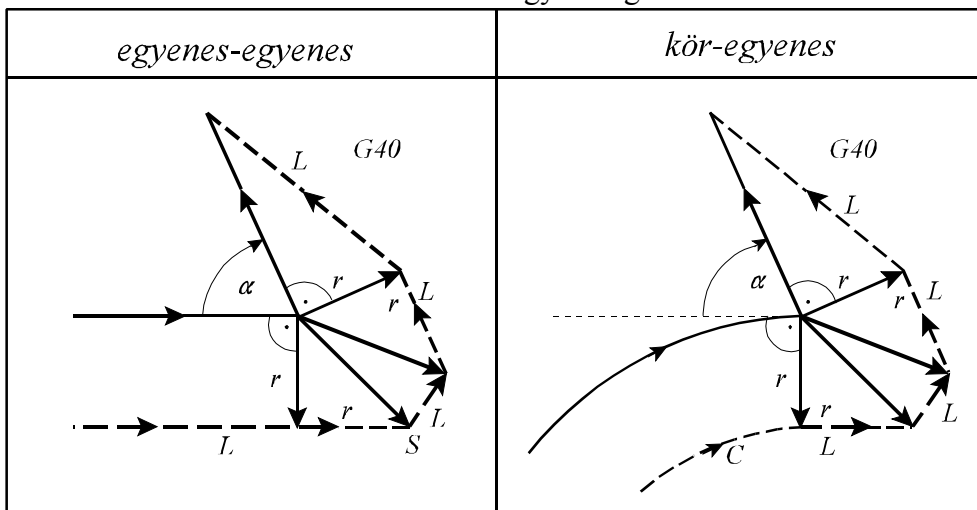
14.5.3-1 ábra

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



14.5.3-2 ábra

Külső sarokból való kiállítás hegyesszög alatt:  $0^\circ \leq \alpha < 90^\circ$



14.5.3-3 ábra

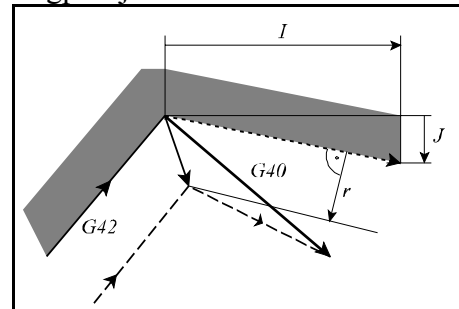
A sugárkorrekció kikapcsolásának speciális esetei:

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 kiállítás esetén hasznos.

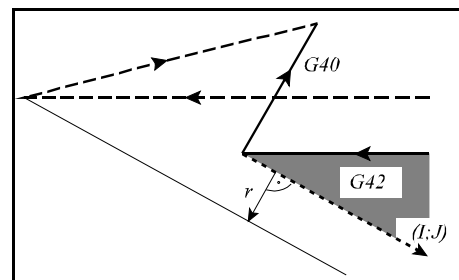
```

...
...G91 G17 G42...
N100 G1 X50 Y60
N110 G40 X70 Y-60 I100 J-20
...
    
```



14.5.3-4 ábra

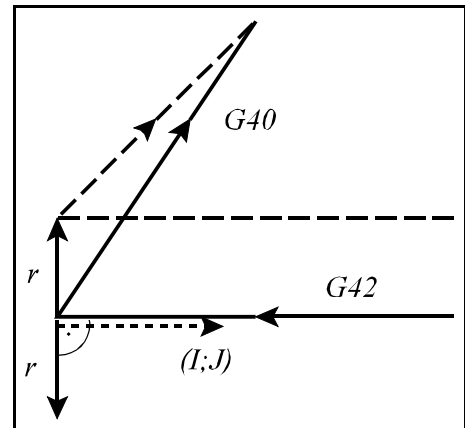
Ebben az esetben a vezérlés mindig metszéspontot számol, függetlenül attól, hogy belső, vagy külső sarkot munkálunk meg.



14.5.3-5 ábra



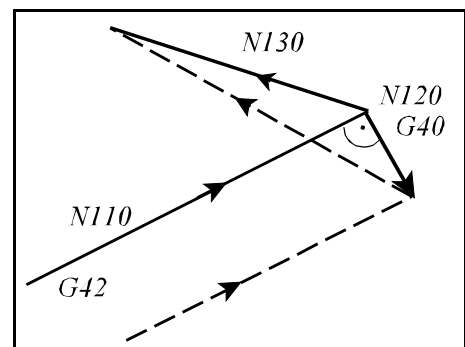
Ha nem talál metszéspontot, az előző mondat végpontjára merőlegesen áll rá.



14.5.3-6 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.

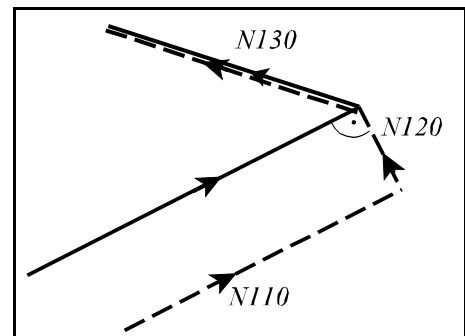
```
...G42 G17 G91...
N110 G1 X80 Y40
N120 G40
N130 X-70 Y20
...
```



14.5.3-7 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 G17 G91...
N110 G1 X80 Y40
N120 G40 X0
N130 X-70 Y20
...
```



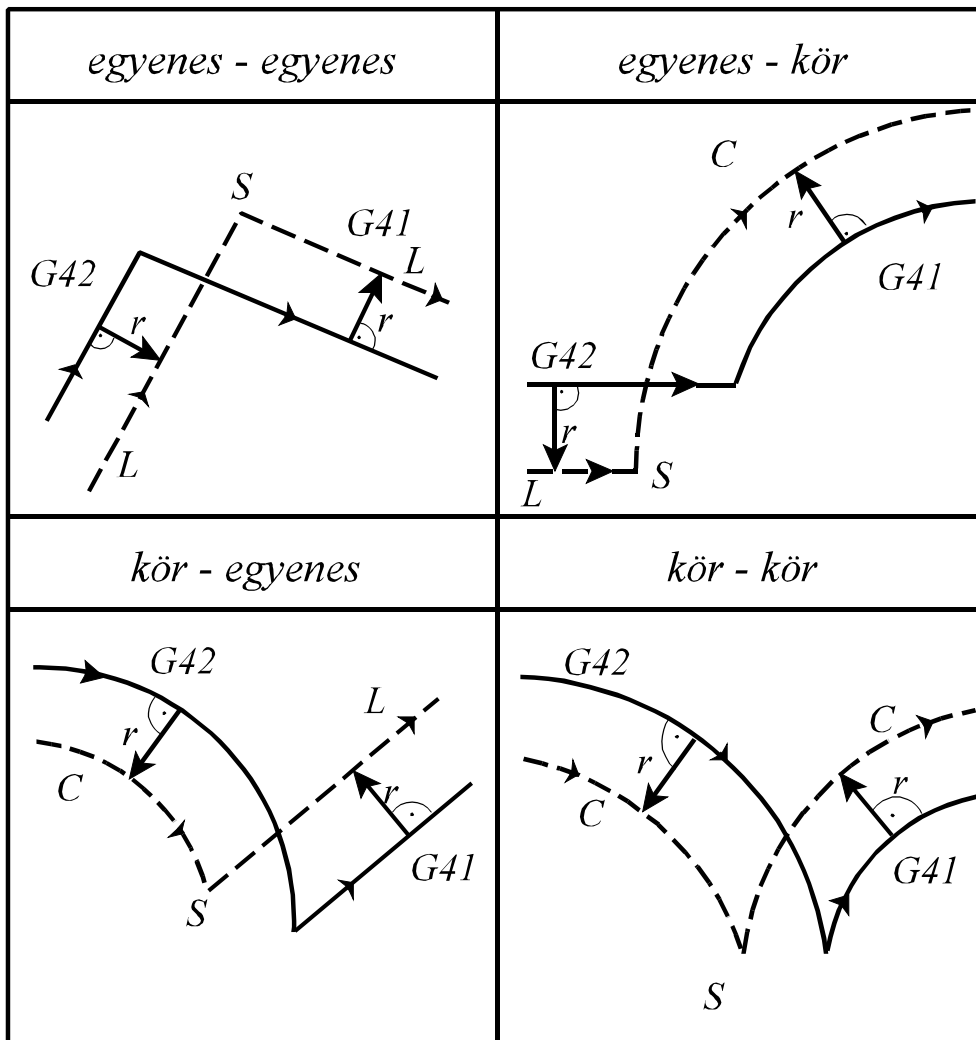
14.5.3-8 ábra

### 14.5.4 Irányváltás a sugárkorrekció számításban.

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

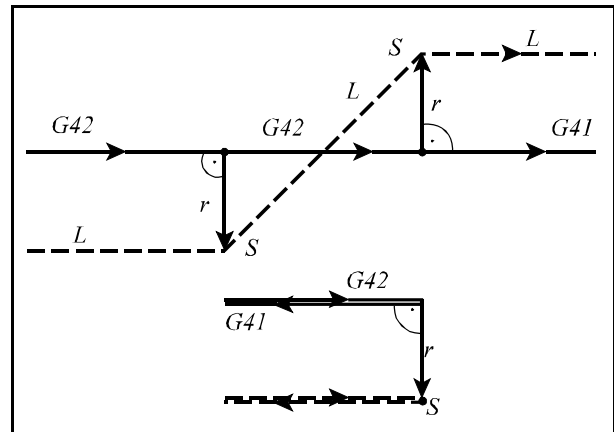
	sugárkorrekció: <b>pozitív</b>	sugárkorrekció: <b>negatív</b>
<b>G41</b>	balról	jobbról
<b>G42</b>	jobbról	balról

A kontúrkövetés iránya megfordítható a szerszám sugár korrekció számítás bekapcsolt állapotában is. Ez történhet G41, vagy G42 programozásával, vagy az eddigiekkel ellentétes előjelű szerszám sugár korrekció lehívásával D 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 szerszám sugarat tételeztünk fel és G42-ből G41-be történő kapcsolást:



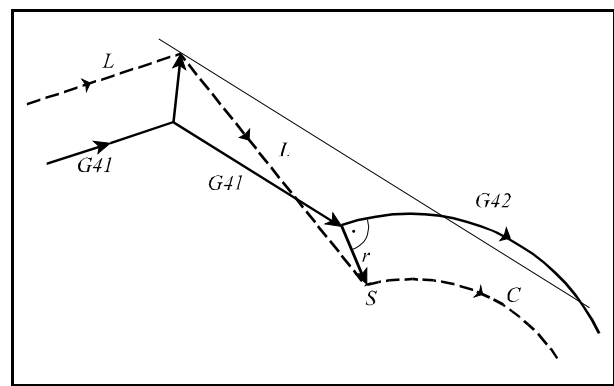
14.5.4-1 ábra

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



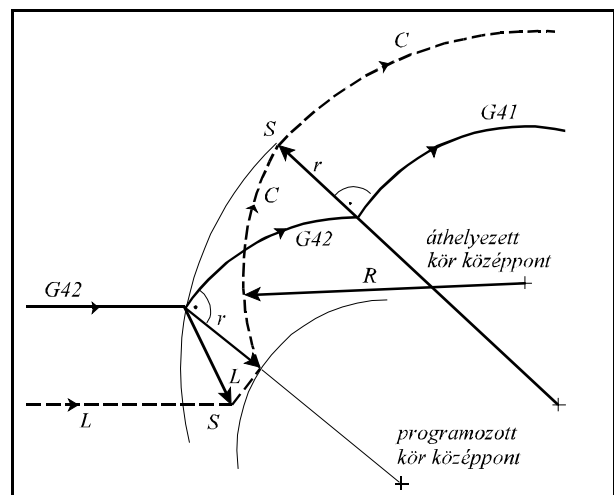
14.5.4-2 ábra

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



14.5.4-3 ábra

Ha kör–egyenes, vagy kör–kör átmenetnél nem adódik metszéspont, akkor az első körmondattal 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.



14.5.4-4 ábra

### 14.5.5 A vektor megtartás programozása (G38)

A

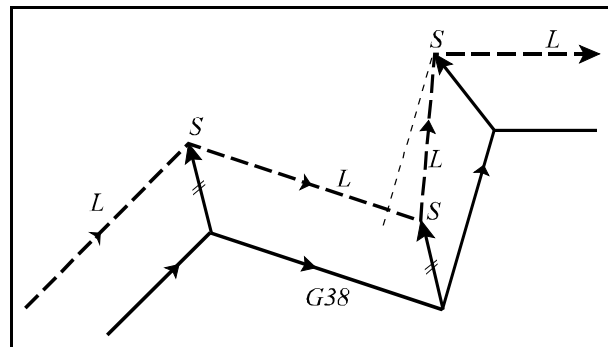
#### G38 v

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 ALATT* hibajelzést ad.

Példa a G38 működésére:

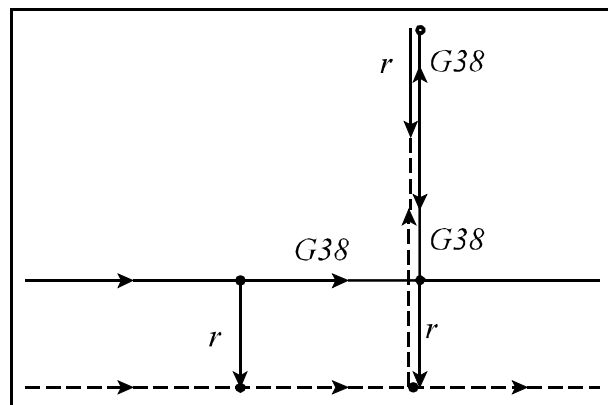
```
...G17 G41 G91...
N110 G1 X60 Y60
N120 G38 X90 Y-40
N130 X20 Y70
N140 X60
...
```



14.5.5-1 ábra

Ha beszúrást akarunk programozni a kontúr-  
követés kikapcsolása nélkül:

```
...G17 G42 G91...
N110 G1 X40
N120 G38 X50
N130 G38 Y70
N140 G38 Y-70
N150 X60
...
```



14.5.5-2 ábra

### 14.5.6 Sarokív programozása (G39)

A

#### G39 (I J K)

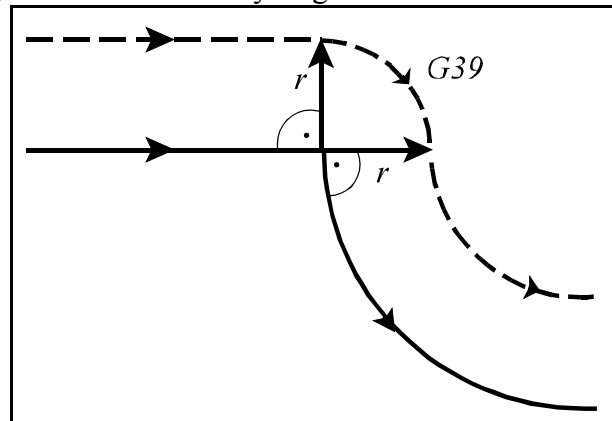
mondat programozásával a síkbeli szerszámsugár korrekciószámítás bekapcsolt állapotában elérhető, hogy külső sarkok kerülése esetén a vezérlő ne számoljon automatikusan metszéspontot, vagy ne iktasson be egyenes szakaszokat a kerüléshez, hanem egy szerszámsugárnyi körívén mozogjon a szerszám középpontja.

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

A kör kezdőpontját az előző mondat pályájának végpontjára merőleges szerszámrádiusnyi hosszúságú vektor, végpontját pedig a következő mondat kezdőpontjára merőleges szerszámrádiusnyi hosszúságú vektor adja.

G39-et külön mondatban kell programozni:

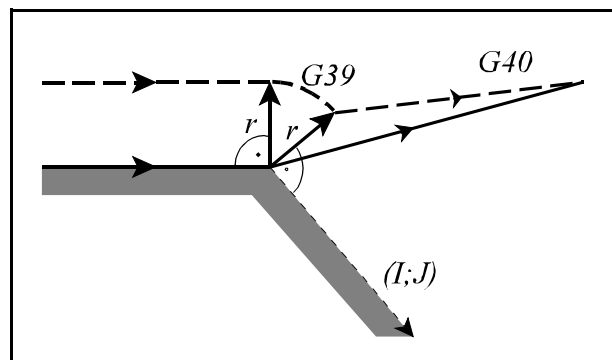
```
...G17 G91 G41...
N110 G1 X100
N120 G39
N130 G3 X80 Y-80 I80
...
```



14.5.6-1 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, szerszámrádiusnyi hosszúságú vektor adja:

```
...G17 G91 G41...
N110 G1 X100
N120 G39 I50 J-60
N130 G40 X110 Y30
...
```



14.5.6-2 ábra

Az I, J vagy K által meghatározott vektorra érvényesek az előzőleg beállított tükrözési, vagy elforgatási 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.

### 14.5.7 Általános tudnivalók a síkbeli sugárkorrekció alkalmazása esetére

A kontúrkövetés bekapcsolt állapotában (G41, G42) a vezérlés mindig két, a kiválasztott síkba eső mozgásmondat közötti korrekciósvektorokat kell, hogy számolja. A gyakorlatban szükség lehet arra, hogy két síkbeli mozgásmondat közé 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: (G17) G1 Z

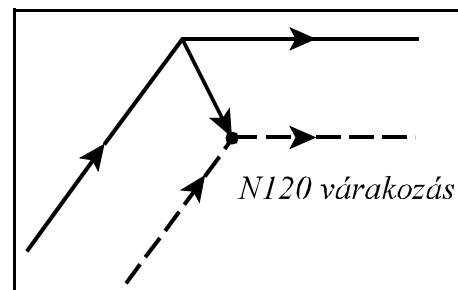
alprogram hívás: M98 P

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 mozgásmondatokkal kezdődjön, mivel ellenkező esetben a pálya torzulni fog.

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

```
...G17 G42 G91...
N110 G1 X50 Y70
N120 G4 P2
N130 X60
...
```

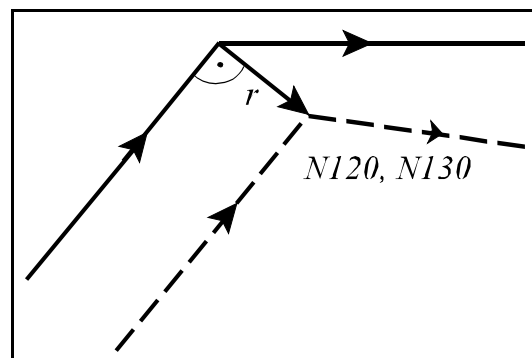


14.5.7-1 ábra

*Ha két kontúrmondat közé 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:

```
...G17 G42 G91...
N110 G1 X50 Y70
N120 G4 P2
N130 S400
N140 X60
...
```

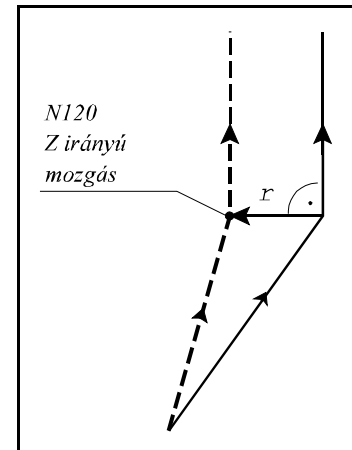


14.5.7-2 ábra

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

```
...G17 G91...
N110 G41 G0 X50 Y70 D1
N120 G1 Z-40
N130 Y40
...
```

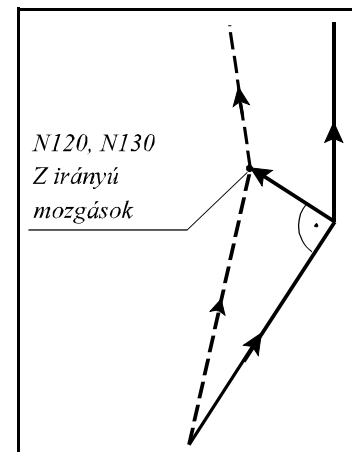
Ebben az esetben a szerszám pályája korrekt lesz, amint az ábrán látszik.



14.5.7-3 ábra

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

```
...G17 G91...
N110 G41 G0 X50 Y70 D1
N120 Z-35
N130 G1 Z-5
N140 Y40
...
```

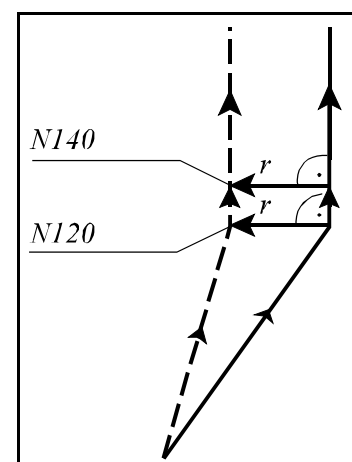


14.5.7-4 ábra

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

```
...G17 G91...
N110 G41 G0 X50 Y69 D1
N120 Z-35
N130 Y1
N140 G1 Z-5
N150 Y40
...
```

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



14.5.7-5 ábra

Ha két kontúrmondat közé:

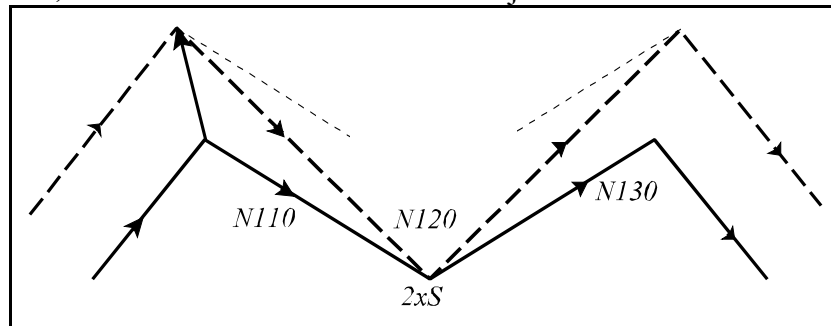
```
G22, G23, G52, G54-G59, G92
G53
G28, G29, G30
```

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

Például:

```
...G91 G17
G41...
N110 G1 X80 Y-50
N120 G92 X0 Y0
N130 X80 Y50
...
```

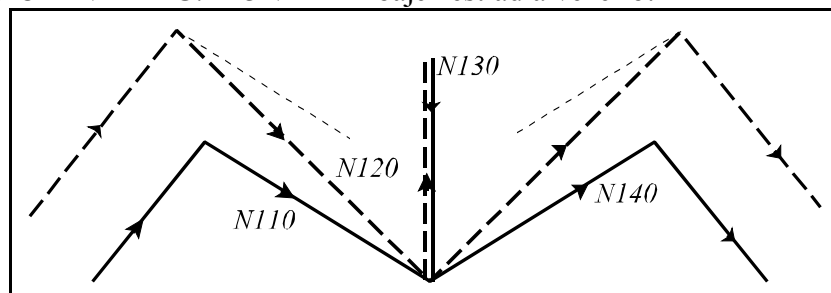


14.5.7-6 ábra

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 pozícionálást, 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 G17
G41...
N110 G1 X80 Y-50
N120 G53 Y80
N130 G53 Y0
N140 X80 Y50
...
```

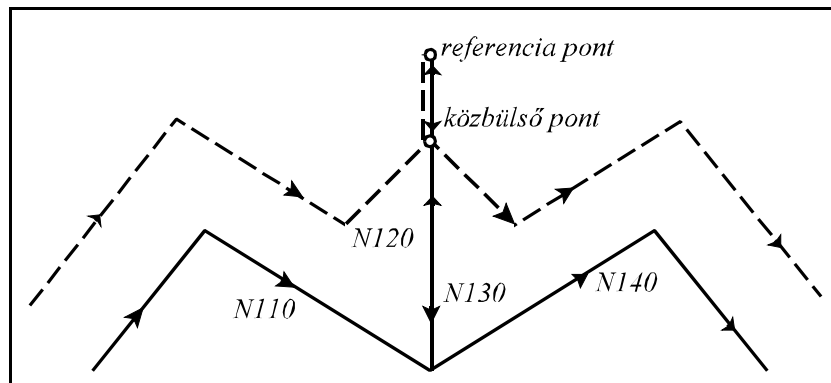


14.5.7-7 ábra

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

Például:

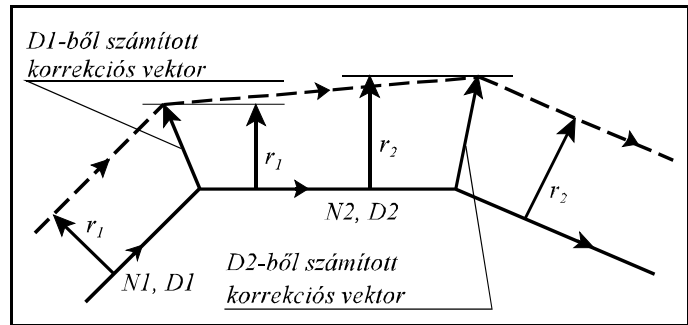
```
...G91 G17
G41...
N110 G1 X80 Y-50
N120 G28 Y80
N130 G29 Y0
N140 X80 Y50
...
```



14.5.7-8 ábra



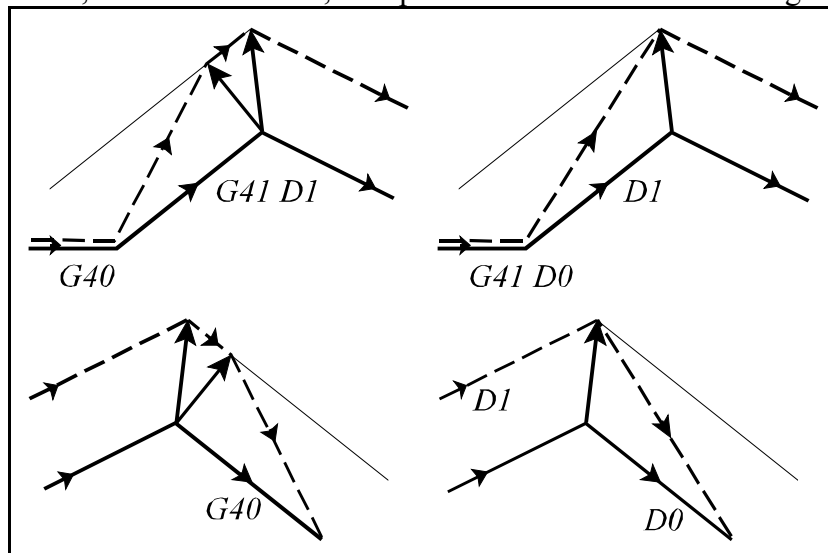
A sugárkorrekció számítás bekapcsolt állapotában (G41, G42) D címen új korrekciós sugárérték is lehívható. Ha a sugár értéke előjelet vált a kontúron való irányváltás folyamata zajlik le, amit már tárgyaltunk. Ha a sugárérték nem vált előjelet az eljárás a következő lesz. Az új sugárértékkal számolt korrekciós vektor annak a mondatnak a végpontjában kerül kiszámításra, amelyben az új D címet programoztuk.



14.5.7-9 ábra

Mivel ennek a mondatnak a kezdőpontjában az előző sugárértékkel lett a korrekciós vektor kiszámítva a szerszámközepppont pályája nem lesz párhuzamos a programozott pályával. Körmondatban is lehívható új sugárkorrekciós érték D címen, ám ekkor a szerszámközepppont egy változó sugarú köríven fog mozogni.

Speciális esete a fent elmondottaknak, ha a sugárkorrekciószámítás bekapcsolt állapotában D00-lal kikapcsoljuk a korrekciót, Dnn-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 D cím lehívásával végezzük:



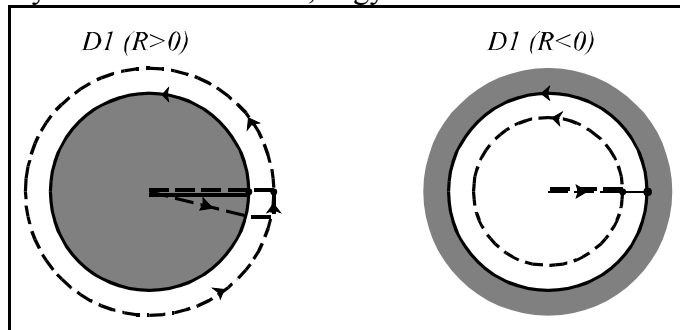
14.5.7-10 ábra

Egy adott programrészletet, vagy alprogramot használhatunk arra is, hogy pozitív sugárkorrekcióval apa, negatív sugárkorrekcióval anya darabot készítsünk, vagy fordítva.

Tekintsük az alábbi kis programrészletet:

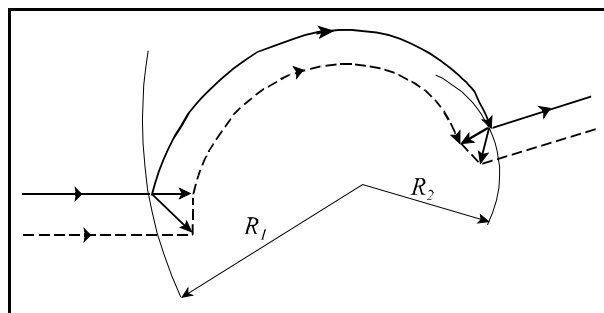
```

...
N020 G42 G1 X80 D1
N030 G1 Z-5
N040 G3 I-80
N050 G1 Z2
N060 G40 G0 X0
...
    
```



14.5.7-11 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 középponttal. 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.



14.5.7-12 ábra

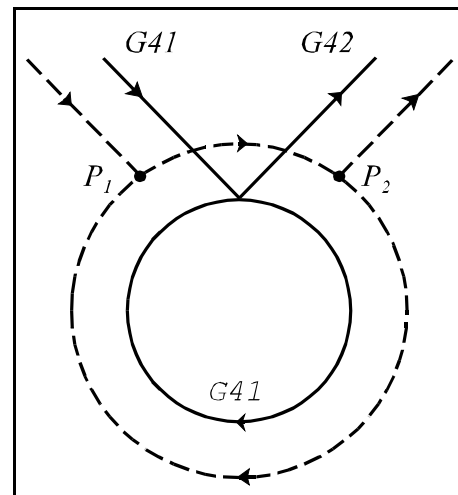
Teljes kört programozva kontúrkövetés közben számos olyan eset adódhat, amikor a szerszám pályája többet tesz meg egy teljes kör fordulatnál.

A kontúron való irányváltás programozásakor adódhat például ilyen helyzet:

```

...G17 G42 G91...
N110 G1 X30 Y-40
N120 G41 G2 J-40
N130 G42 G1 X30 Y40
...
    
```

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

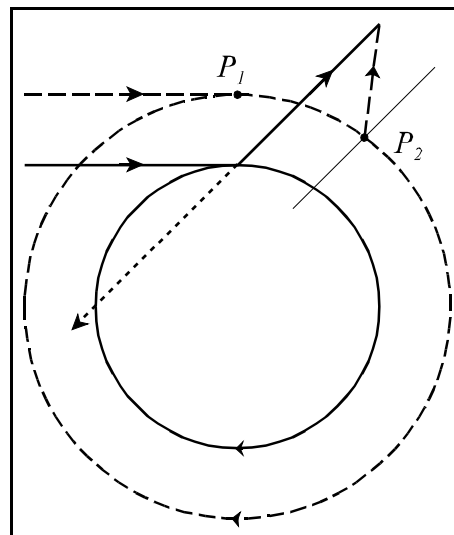


14.5.7-13 ábra

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

```
...G17 G90 G41...
N090 G1 X30
N100 G2 J-60
N110 G40 G1 X120 Y180 I-60 J-60
...
```

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

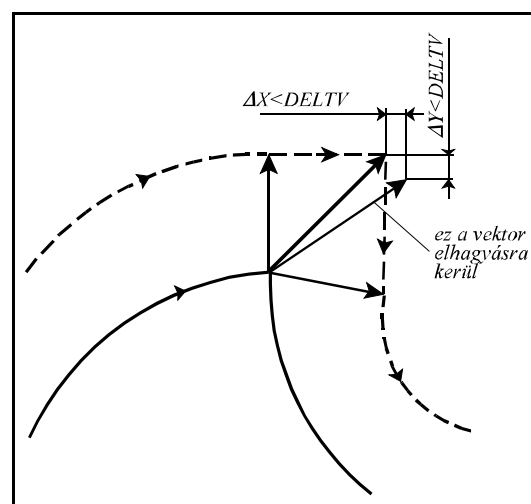


14.5.7-14 ábra

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!

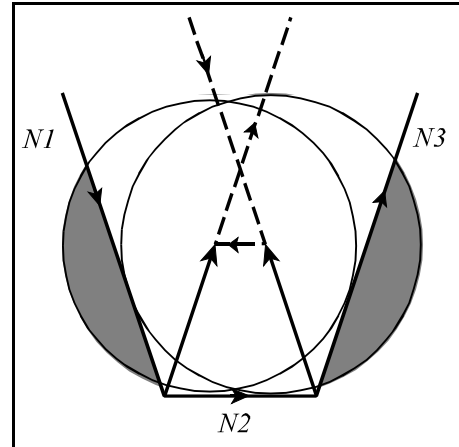


14.5.7-15 ábra

### 14.5.8 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.

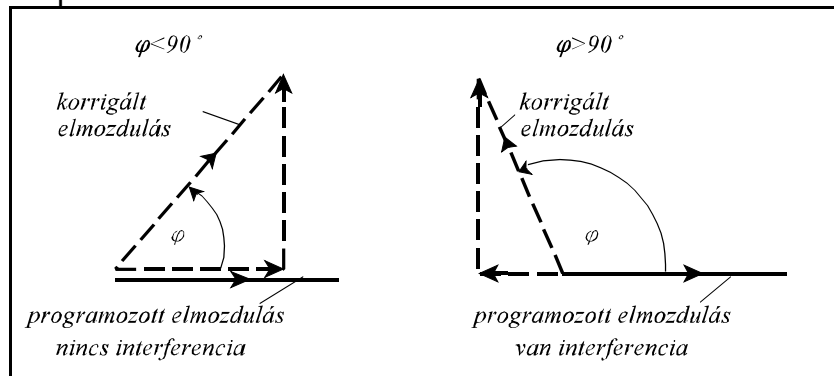
Az ábrán látható esetben a metszéspontok kiszámítása után az N2 mondat végrehajtása során a programozottal ellentétes szerszám pálya adódik. A bevonalkázott terület jelzi, hogy a szerszám belevág a munkadarabba.



14.5.8-1 ábra

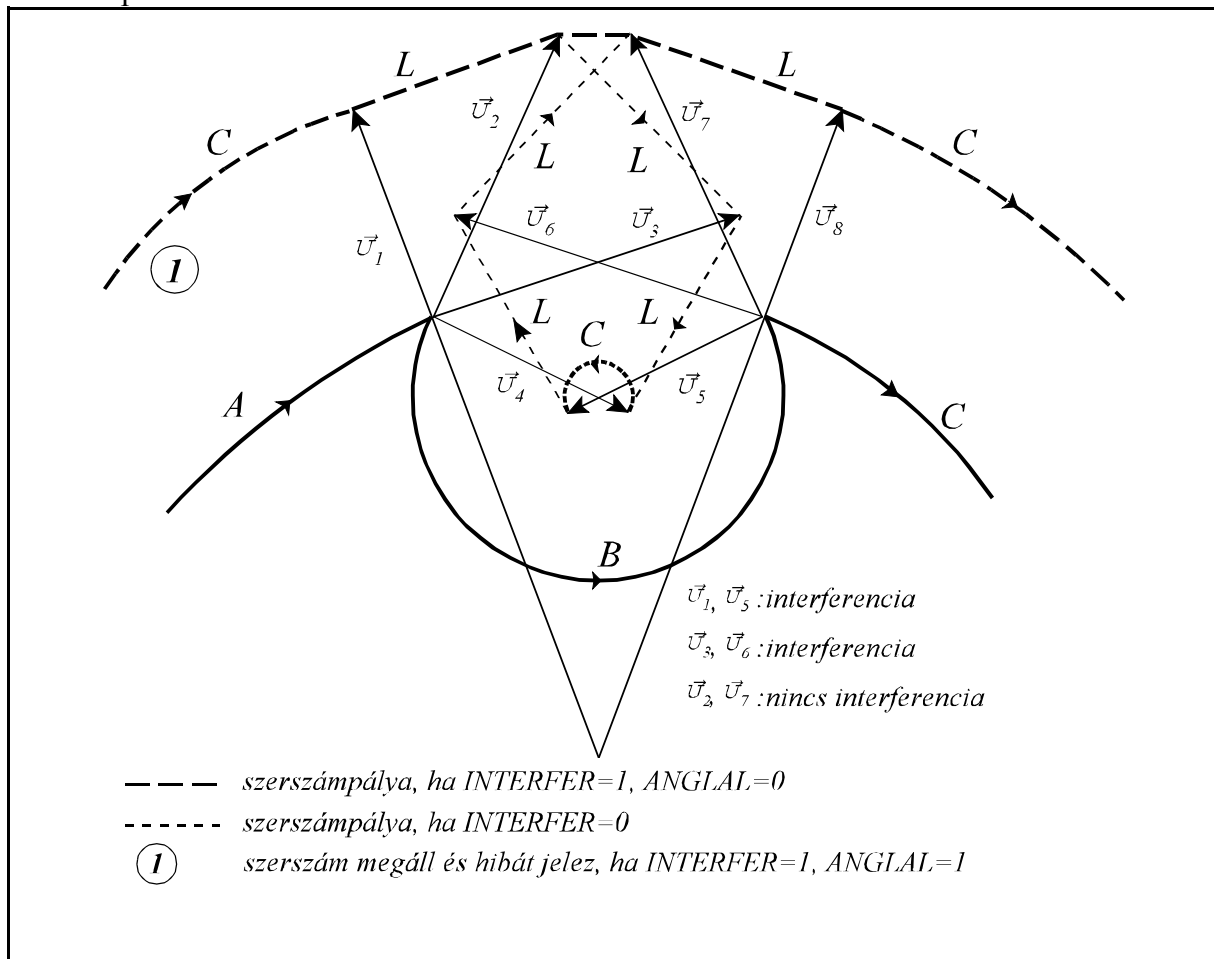
Az ilyen esetek elkerülése érdekében a vezérlés interferenciavizsgálatot végez, ha a paramétermező *INTERFER* paraméterét 1-be írjuk. Ebben az esetben a vezérlés azt vizsgálja, hogy a programozott elmozdulás és a sugárkorrekcióval korrigált elmozdulás közötti  $\varphi$  szögre teljesül-e a következő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



14.5.8-2 ábra

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



14.5.8-3 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:  $\vec{u}_1, \vec{u}_2, \vec{u}_3, \vec{u}_4$ . B és C mondat között a korrekciós vektorok pedig:  $\vec{u}_5, \vec{u}_6, \vec{u}_7, \vec{u}_8$ .

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

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

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

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

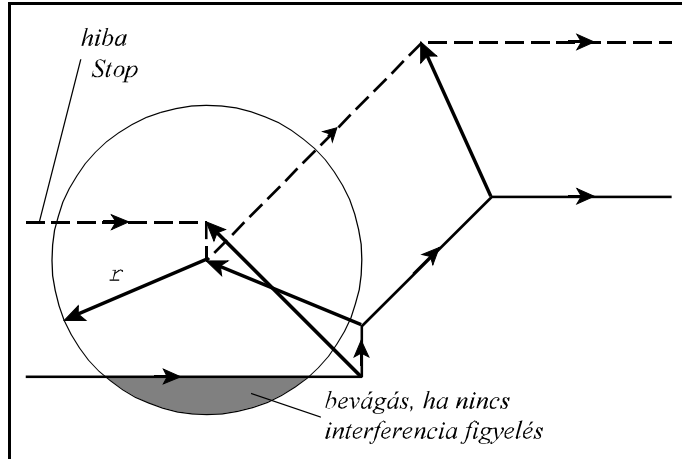
A fentiekből látható, hogy a korrekciós vektorokat a B mondat kezdőpontján és végpontján párba veszi, és párosával hagyja el azokat. Ha az egyik oldalon a korrekciós vektorok száma egy, vagy egyre csökken, akkor csak a másik oldaliakat hagyja el. Az elhagyás addig folytatódik, amíg az interferencia fennáll. 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.

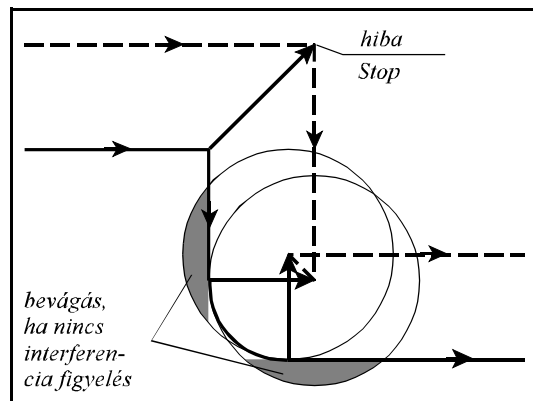
Az alábbiakban néhány tipikus esetet adunk interferenciára.

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



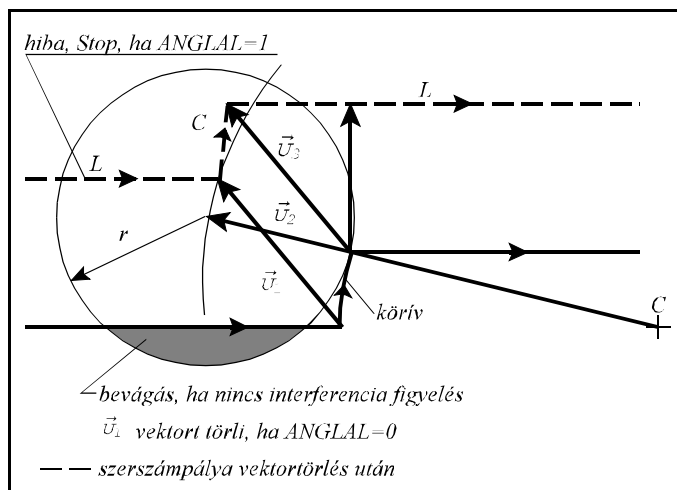
14.5.8-4 ábra

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



14.5.8-5 ábra

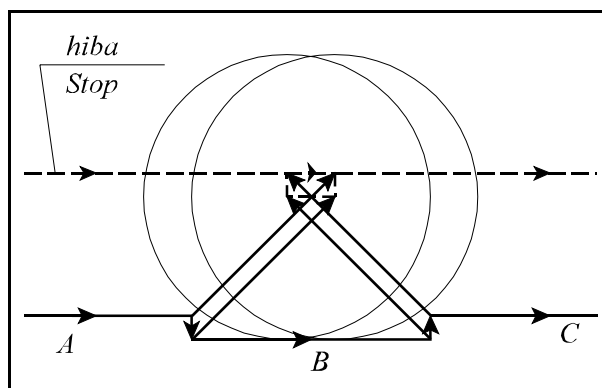
A szerszámsugárnál kisebb lépcső forgácsolása körív mentén. Ha az *ANGLAL* paraméter 0 a vezérlés a  $\vec{v}_2$  vektort törli, és a  $\vec{v}_1$   $\vec{v}_3$  vektort egy egyenessel köti össze, hogy elkerülje a bevágást. Ha az *ANGLAL* paraméter értéke 1, akkor *3048 INTERFERENCIA HIBA* hibát jelez,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



14.5.8-6 ábra

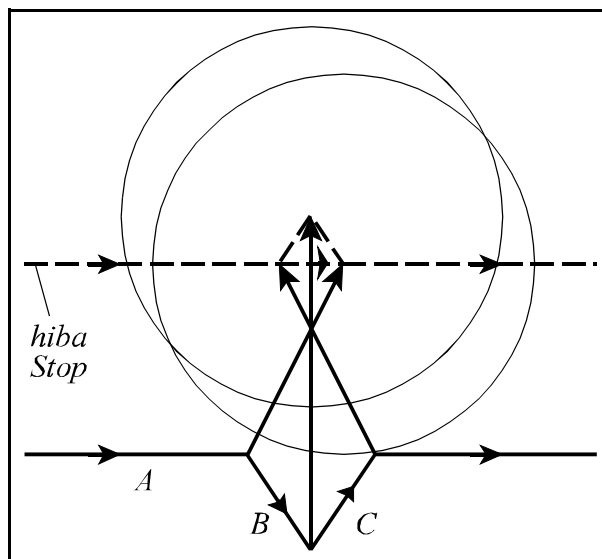
Vannak esetek, amikor valójában nem vágna bele a szerszám az anyagba, de az interferenciavizsgálat hibát jelez.

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.



14.5.8-7 ábra

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



14.5.8-8 ábra

### 14.6 A háromdimenziós szerszámkorrekció (G41, G42)

A síkbeli szerszámsugár korrekció használata a szerszámot a G17, G18, G19 parancsok által kiválasztott síkban tolja el. Lehetőség van a szerszámkorrekciót térben is figyelembe venni a háromdimenziós szerszámkorrekció alkalmazásával.

#### 14.6.1 A háromdimenziós szerszámkorrekció be-, kikapcsolása (G40, G41, G42)

A

**G41 (G42) X<sub>p</sub> Y<sub>p</sub> Z<sub>p</sub> I J K D (E)**

paranccsal a térbeli szerszámkorrekciót kapcsolhatjuk be.

X<sub>p</sub>, Y<sub>p</sub>, Z<sub>p</sub> jelentése: X, Y, Z, vagy azzal párhuzamos tengely, ha van.

Ha valamelyik koordinátairányra nem történik hivatkozás automatikusan a főirányokat veszi figyelembe. Például:

G41 X I J K utasítás az X Y Z térre vonatkozik

G41 U V Z I J K utasítás az U V Z térre vonatkozik

G41 W I J K utasítás az X Y W térre vonatkozik

I, J, K címek közül mindhármat specifikálni kell a háromdimenziós szerszámkorrekció számítás bekapcsolásakor. Ha ez nem történik meg, akkor a vezérlő a síkbeli szerszámsugár korrekció számítás állapotát veszi fel.

Az I, J, K címeiken megadott értékek a háromdimenziós korrekciósvektor komponensei. Az egyes komponensek értékei öröklődnek, vagyis, amíg nem hivatkozunk újabb I, J, vagy K értékre, az előző marad érvényben.

D címen hívható le az alkalmazni kívánt korrekciós érték.

E címen adható meg a korrekciószámítás dominátor konstans értéke.

A

**G40** vagy

**D00**

parancs kikapcsolja a háromdimenziós korrekciószámítást.

A két parancs közti különbség, hogy a D00 utasítás csak a korrekciót törli és a G41 vagy G42 állapotot változatlanul hagyja. Ha ezek után új, nullától különböző D címre történik hivatkozás a G41 vagy a G42 állapot függvényében az új szerszámkorrekció bekapcsolódik.

Ha viszont G40 utasítást használunk, utána addig elsül a levegőben minden D címre történő hivatkozás, amíg G41-et vagy G42-t nem programoztunk.

A korrekciószámítást bekapcsolni (G41, G42), vagy kikapcsolni (G40 vagy D00) csak lineáris mozgást végző mondatban (G00 vagy G01) lehet.

G40, G41, G42 parancsok öröklődnek. Bekapcsolás után a vezérlés a G40 állapotot veszi fel.



### 14.6.2 A háromdimenziós korrekciós vektor

A vezérlés a korrekciósvektor komponenseket a következő módon generálja magának:

$$v_x = \frac{I * r}{P}$$

$$v_y = \frac{J * r}{P}$$

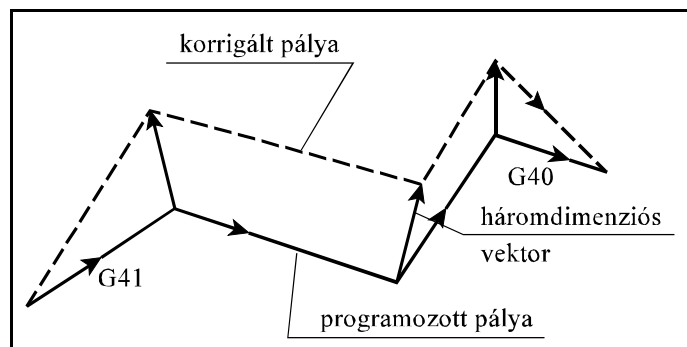
$$v_z = \frac{K * r}{P}$$

ahol  $r$ : a D címen lehívott korrekciós érték,  
 $P$ : normálószám, vagy dominátor konstans,  
 $I, J, K$ : a programban megadott értékek.

A normálószám értékét a paraméterező *DOMCONST* paraméteréről veszi, ha E címen nem adunk meg ettől eltérő értéket a programban. Ha a dominátor konstans értéke 0, és E címen sem adunk meg értéket, akkor a vezérlő a következő összefüggésből számolja ki  $P$  értékét:

$$P = \sqrt{I^2 + J^2 + K^2}$$

A vezérlés a mondatról mondatra megadott korrekciós vektorirányok alapján mondatról mondatra figyelembe veszi a korrekciót. Így a háromdimenziós megmunkálások során a CAM rendszernek nem kell egy adott szerszámhoz generálni a pályát, hanem csak a mondatok végpontjában a vektorirányokat kiszámítani, és az így generált programok különböző méretű szerszámokkal is lefuttathatók.



14.6.2-1 ábra

Körmondatban nincs lehetőség a korrekciósvektor változtatására, azaz körmondat elején és végén a korrekciósvektor megegyezik.

A G42 utasítás úgy működik, mint a G41 azzal a különbséggel, hogy a korrekciós vektort a G41-ével ellentétes irányban számítja:

$$v_x = -\frac{I * r}{P}$$

$$v_y = -\frac{J * r}{P}$$

$$v_z = -\frac{K * r}{P}$$

G41 állapotból G42 állapotba átkapcsolni, és viszont csak egyenes interpolációs módban lehet. Ha a háromdimenziós szerszámkorrekció bekapcsolt állapotában egy mondatban I is J is K is elhagyásra kerül, az előző értékek öröklődnek.

Háromdimenziós korrekció és síkbeli sugárkorrekció nem lehet egyszerre bekapcsolva.

## 15 Különleges transzformációk

### 15.1 Koordinátarendszer elforgatása (G68, G69)

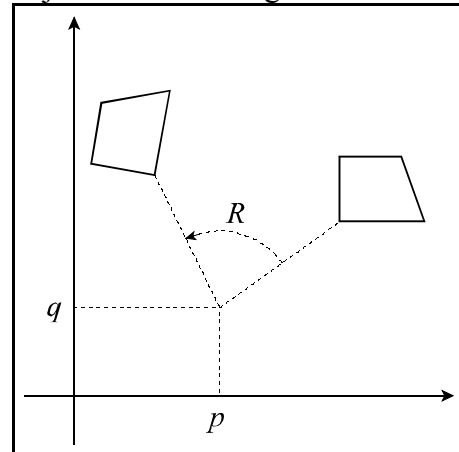
A

#### G68 p q R

paranccsal egy programozott alakzat a G17, G18, G19 által kijelölt síkban elforgatható.  $p$  és  $q$  címen adjuk meg az elforgatás középpontjának koordinátáit. Csak a kiválasztott sík  $p$  és  $q$  koordinátáira írt értékeket értelmezi.

Polárkoordinátás adatmegadás bekapcsolt állapotában is az itt beírt  $p$ ,  $q$  koordinátaadatokat derékszögű koordinátarendszerben értelmezi. Az elforgatás középpontjának  $p$ ,  $q$  koordinátáit megadhatjuk abszolút és növekményes adatként is G90, G91, vagy I operátor használatával.

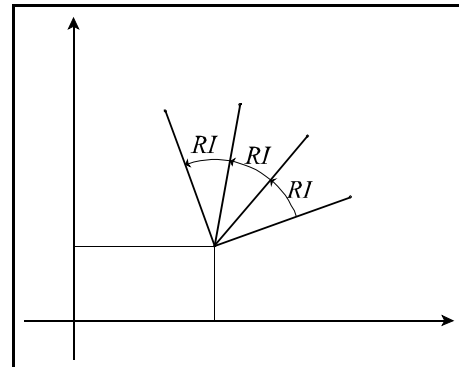
Ha  $p$  és  $q$  valamelyikének, vagy egyikének sem adunk értéket az elforgatás középpontjának a pillanatnyi tengelypozíciót veszi.



15.1-1 ábra

$R$  címen adjuk meg az elforgatás szögét. A címre írt pozitív érték az óramutató járásával ellentétes irányt, a negatív érték az óramutató járásával megegyező irányt jelent.  $R$  értékét 8 decimális számjegyen adhatjuk meg. Az elforgatás pontossága az *ANG.ACCU* paraméter alapján választható ki. Ha a paraméter értéke 0 az elforgatás számítási pontossága  $0.001^\circ$ , ha 1 a pontosság  $0.00001^\circ$ .

$R$ -re adott érték abszolút és inkrementális is lehet. Ha az elforgatás szögét inkrementálisan adjuk meg, akkor az előzőleg programozott elforgatási szögekhez hozzáadódik  $R$  értéke.



15.1-2 ábra

A

#### G69

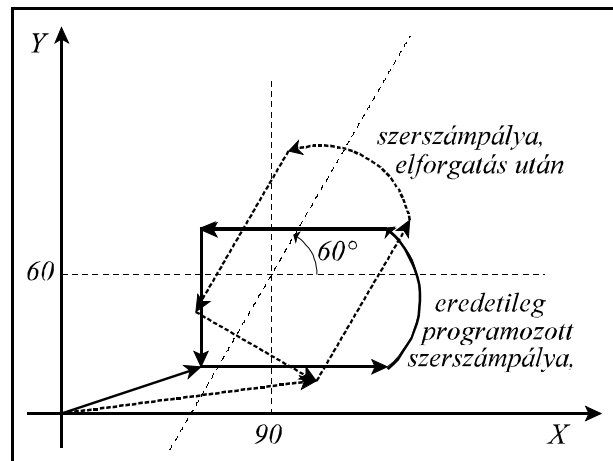
paranccsal az elforgatást ki lehet kapcsolni. Törli az elforgatás középpontjának koordinátáit, és az elforgatási szöget is. Ez az utasítás más parancsok mellett is állhat.

**Mintapélda:**

```

N1 G17 G90 G0 X0 Y0
N2 G68 X90 Y60 R60
N3 G1 X60 Y20 F150
  (G91 X60 Y20 F150)
N4 G91 X80
N5 G3 Y60 R100
N6 G1 X-80
N7 Y-60
N8 G69 G90 X0 Y0

```



15.1-3 ábra

**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. Polárkoordinátás adatmegadás bekapcsolt állapotában is az itt beírt v koordinátaadatokat derékszögű koordináta-rendszerben értelmezi.

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

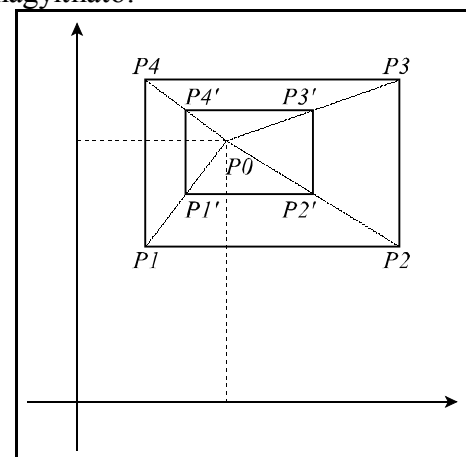
Ha valamelyik tengelycímnek, vagy egyiknek sem adunk értéket a léptékezés középpontjának a pillanatnyi tengelypozíciót veszi.

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

A

**G50**

paranccsal a léptékezésszámítás kikapcsolható.

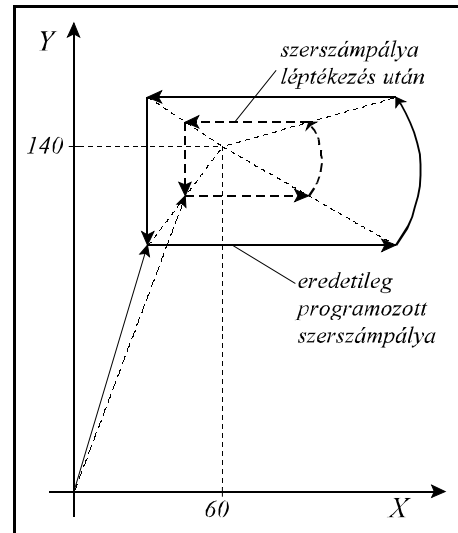


15.2-1 ábra

**Mintapélda:**

```

N1 G90 G0 X0 Y0
N2 G51 X60 Y140 P0.5
N3 G1 X30 Y100 F150
   (G91 X30 Y100 F150)
N4 G91 X100
N5 G3 Y60 R100
N6 G1 X-100
N7 Y-60
N8 G50 G90 X0 Y0
    
```



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.

Polárkoordinátás adatmegadás bekapcsolt állapotában is az itt beírt v koordinátaadatokat derékszögű koordinátarendszerben értelmezi.

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 elforgatási, sem léptékezési parancsállapot.

Ellenkező esetben 3000 TŰKRÖZÉS G51, G68 ALATT hibajelzést ad.

Ha a kiválasztott sík egyik tengelyére tükrözünk:

- a körirány automatikusan megfordul (G02 G03 csere),
- az elforgatás szöge (G68) ellentétesen értelmeződik.

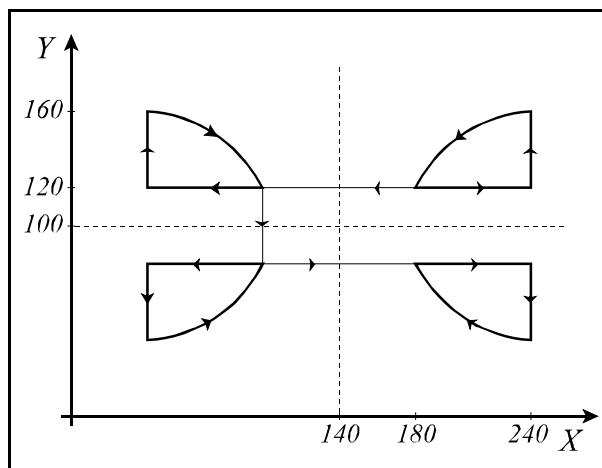
Mintapélda:

alprogram:

```
O0101
N1 G90 G0 X180 Y120 F120
N2 G1 X240
N3 Y160
N4 G3 X180 Y120 R80
N5 M99
```

főprogram:

```
O0100
N1 G90 (abszolút koordinátamegadás)
N2 M98 P101 (alprogram hívás)
N3 G51.1 X140 (tükrözés az Y-nal párhuzamos, X=140 koordinátájú
tengelyre)
N4 M98 P101 (alprogram hívás)
N5 G51.1 Y100 (tükrözés az X-szel párhuzamos, Y=100 koordinátájú
tengelyre is)
N6 M98 P101 (alprogram hívás)
N7 G50.1 X0 (tükrözés kikapcsolása az Y-nal párhuzamos tengelyen)
N8 M98 P101 (alprogram hívás)
N9 G50.1 Y0 (tükrözés kikapcsolása az X-szel párhuzamos tengelyen)
```



15.3-1 ábra

#### 15.4 A különleges transzformációk programozási szabályai

A G68 elforgatás és a G51 léptékezés utasítások sorrendje tetszőleges lehet.

Vigyázni kell viszont arra, hogy ha először elforgatás után léptékezek, akkor a léptékezés középpontjának koordinátáira is érvényes az elforgatási parancs. Ha viszont először léptékezek és utána forgatok el, az elforgatás középpontjának koordinátáira a léptékezési parancs lesz érvényes.

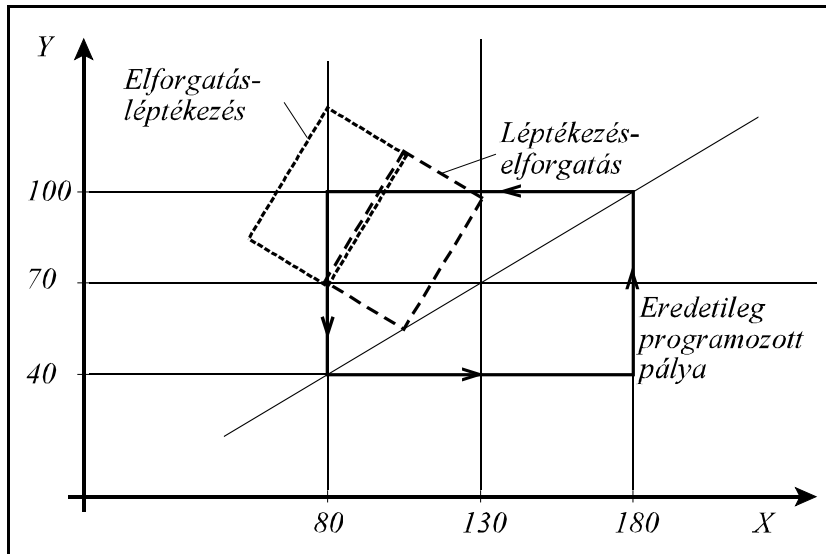
A két művelet bekapcsolási és kikapcsolási parancsainak viszont egymásba kell skatulyázódnuk, egymást nem lapolhatják át:

##### Elforgatás–léptékezés

```
N1 G90 G17 G0 X0 Y0
N2 G68 X80 Y40 R60
N3 G51 X130 Y70 P0.5
N4 X180 Y40
N5 G1 Y100 F200
N6 X80
N7 Y40
N8 X180
N9 G50
N10 G69 G0 X0 Y0
```

##### Léptékezés–elforgatás

```
N1 G90 G17 G0 X0 Y0
N2 G51 X130 Y70 P0.5
N3 G68 X80 Y40 R60
N4 X180 Y40
N5 G1 Y100 F200
N6 X80
N7 Y40
N8 X180
N9 G69
N10 G50 G0 X0 Y0
```



15.4-1 ábra

Az ábrából látható, hogy nem mindegy milyen sorrendben alkalmazzuk a különböző transzformációkat.

Más a helyzet a tükrözéssel. Tükrözést bekapcsolni csak G50 és G69 állapotban lehet, azaz ha nincs sem léptékezési sem elforgatási parancsállapot.

A tükrözés bekapcsolt állapotában viszont mind a léptékezés, mind az elforgatás bekapcsolható.

A tükrözésre is érvényes, hogy sem a léptékezési, sem az elforgatási parancsokkal nem lapoldhat át, tehát először a megfelelő sorrendben az elforgatást és a léptékezést kell kikapcsolni, és csak utána a tükrözést.

G51.1 ...	(tükrözés bekapcsolása)
G51 ...	(léptékezés bekapcsolása)
G68 ...	(elforgatás bekapcsolása)
...	
G69 ...	(elforgatás kikapcsolása)
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 X30 ,C10
N2 X10 Y40
```

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.

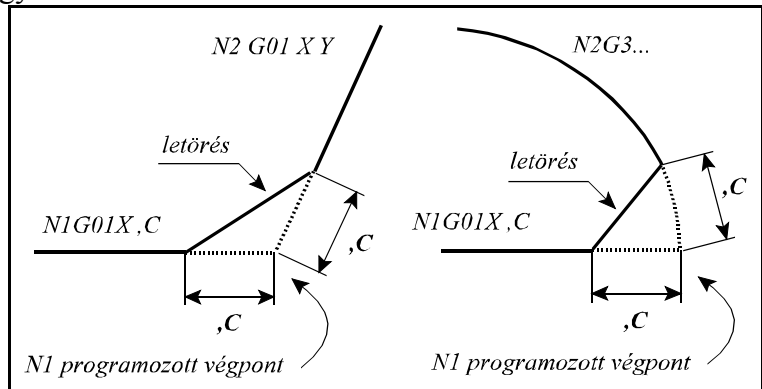
A

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

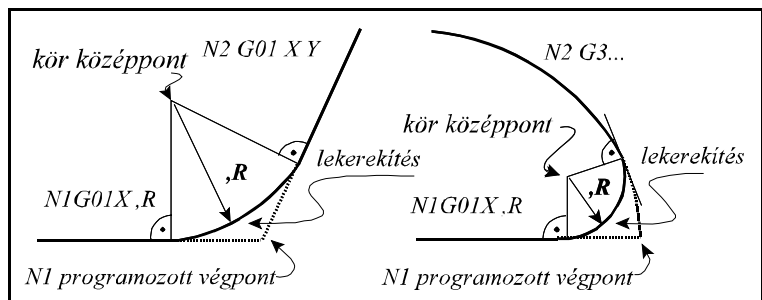
```
N1 G91 G01 X30 ,R8
N2 G03 X-30 Y30 R30
```

A ,R sugarú körívet úgy iktatja a

két mondat közé, hogy a kör mindkét pályaelemhez érintőlegesen simuljon.



16.1-1 ábra



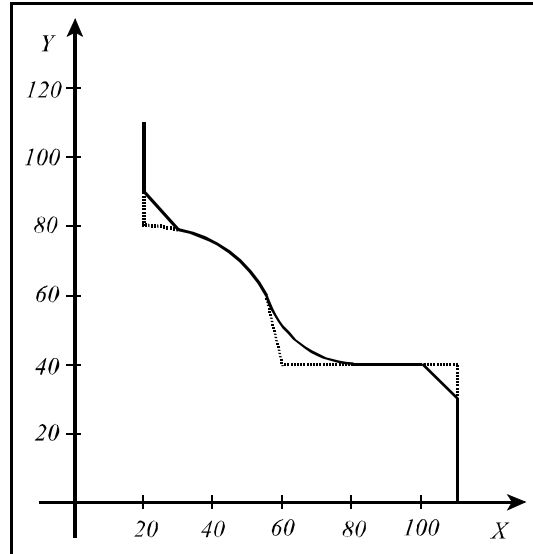
16.1-2 ábra

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

```

...
G1 Y40 ,C10
X60 ,R22
G3 X20 Y80 R40 ,C10
G1 Y110
...

```



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.

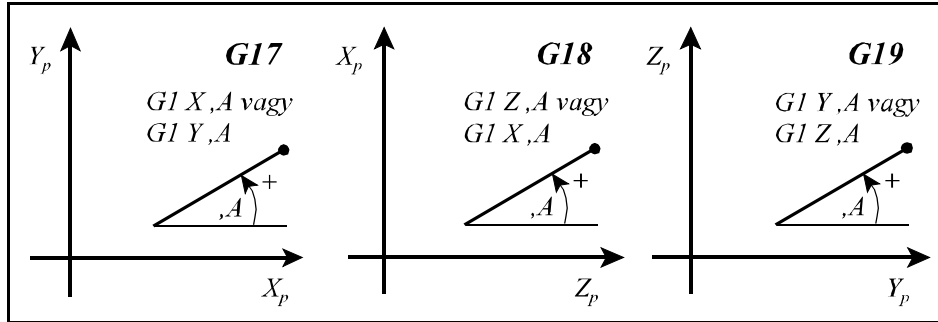
$$G17 \left\{ \begin{matrix} G00 \\ G01 \end{matrix} \right\} \left\{ \begin{matrix} X_p ,A \\ Y_p ,A \end{matrix} \right\} q F$$

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

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

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 számítódik, 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.

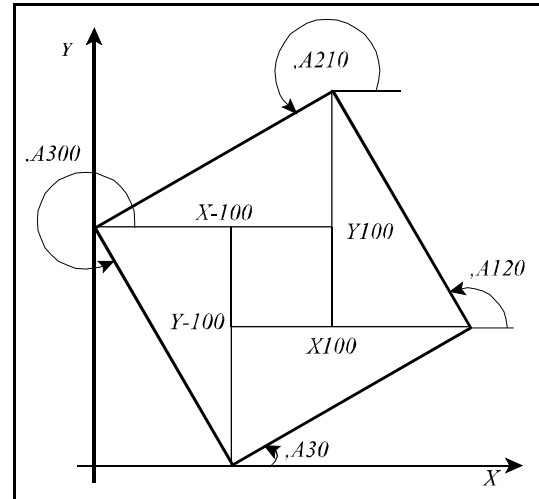




16.2-1 ábra

Például

```
G17
G90 G0 X57.735 Y0 ...
G1 G91...
X100 ,A30 (ez a megadás ekvivalens az
            X100 Y57.735 megadással,
            ahol  $7.735=100 \cdot \text{tg}30^\circ$ )
Y100 ,A120 (ez a megadás ekvivalens az
            X-57.735 Y100 megadással,
            ahol  $-57.735=100/\text{tg}120^\circ$ )
X-100 ,A210 (ez a megadás ekvivalens az
            X-100 Y-57.735 megadással,
            ahol  $-57.735=-100 \cdot \text{tg}30^\circ$ )
Y-100 ,A300 (ez a megadás ekvivalens az
            X57.735 Y-100 megadással,
            ahol  $57.735=-100/\text{tg}120^\circ$ )
```



16.2-2 ábra

**Megjegyzés:**

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

```
X100 ,A30 ,C5
Y100 ,A120 ,R10
X-100 ,A210
```

– „A címen történő irányszög megadása használható fűrciklusokban 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
```

### 16.3 Síkbeli metszéspontszámítások

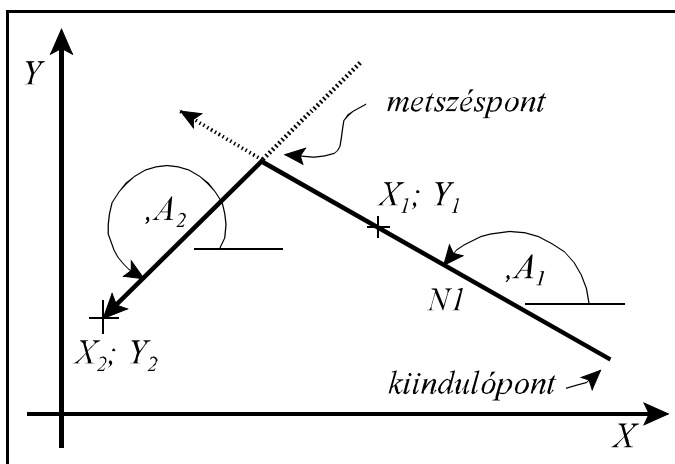
Az itt közölt metszéspontszámításokat a vezérlő csak a **szerszámsugár korrekció számítás bekapcsolt állapotában (G41, G42)** végzi el. Ha esetleg a programban nem akarunk szerszámsugár korrekciót figyelembe venni, akkor is kapcsoljuk be azt és a D00 korrekciót hívjuk le:

sugárkorrekció figyelembe vételével:  
 G41 (vagy G42) ...Dnn  
 ...  
 metszéspontszámítások  
 ...  
 G40

sugárkorrekció nélkül:  
 G41 (vagy G42) ...D00  
 ...  
 metszéspontszámítások  
 ...  
 G40

#### 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 ,A<sub>1</sub> vagy

X<sub>1</sub> Y<sub>1</sub>

N2 G1G90 X<sub>2</sub> Y<sub>2</sub> ,A<sub>2</sub>

G18 G41 (G42)

N1 G1 ,A<sub>1</sub> vagy

X<sub>1</sub> Z<sub>1</sub>

N2 G1G90 X<sub>2</sub> Z<sub>2</sub> ,A<sub>2</sub>

G19 G41 (G42)

N1 G1 ,A<sub>1</sub> vagy

Y<sub>1</sub> Z<sub>1</sub>

N2 G1G90 Y<sub>2</sub> Z<sub>2</sub> ,A<sub>2</sub>

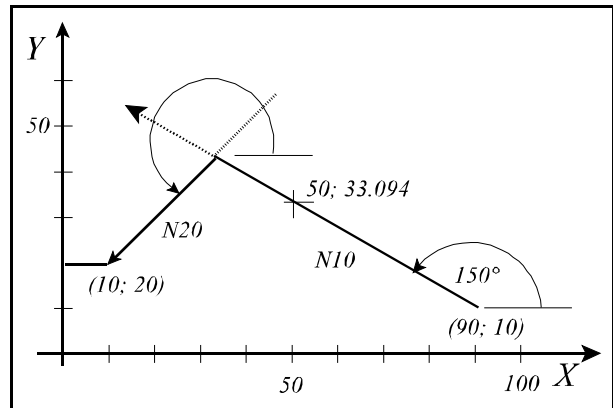
A metszéspontot mindig a G17, G18, G19 által kijelölt síkban számítja ki. Az első mondatot (N1) vagy csak irányszögével (A<sub>1</sub>) adjuk meg, és ebben az esetben a kiindulópontból a megfelelő irányszögben húz egy egyenest a metszéspontig, vagy az egyenes egy tetszőleges, a kiindulóponttól különböző pontját adjuk meg (X<sub>1</sub>, Y<sub>1</sub>; X<sub>1</sub>, Z<sub>1</sub>; vagy Y<sub>1</sub>, Z<sub>1</sub>) és ekkor a két ponton áthaladó egyenessel számítja a metszéspontot. A *második mondatban* (N2) megadott koordinátákat mindig *abszolút* (G90) adatként értelmezi a vezérlő.

Például:

```
G17 G90 G41 D0...
G0 X90 Y10
N10 G1 ,A150
N20 X10 Y20 ,A225
G0 X0 Y20
...
```

Az N10 mondatot megadhatjuk az egyenes egy pontjának koordinátaival is:

```
G17 G90 G41 D0...
G0 X90 Y10
N10 G1 X50 Y33.094
N20 X10 Y20 ,A225
G0 X0 Y20
...
```

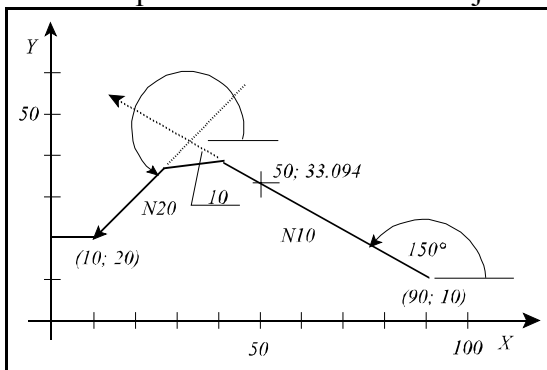


16.3.1-2 ábra

Figyeljük meg, hogy ebben az esetben az

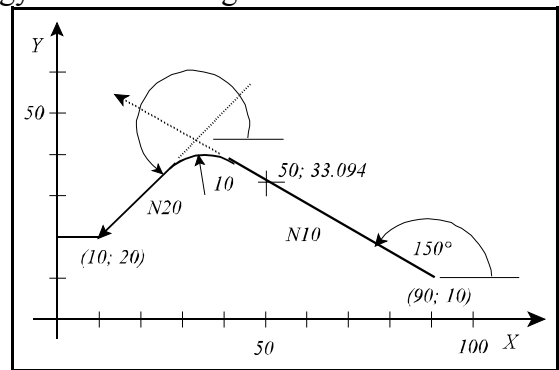
N10 mondatban megadott X, Y koordinátát (X50 Y33.094) nem végpontnak tekinti a vezérlő, hanem csak az egyenes kezdőpontját a megadott ponttal összekötő átmenő pontnak.

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



16.3.1-3 ábra

```
G17 G90 G41 D0...
G0 X90 Y10
N10 G1 X50 Y33.094 ,C10
N20 X10 Y20 ,A225
G0 X0 Y20
...
```



16.3.1-4 ábra

```
G17 G90 G41 D0...
G0 X90 Y10
N10 G1 X50 Y33.094 ,R10
N20 X10 Y20 ,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

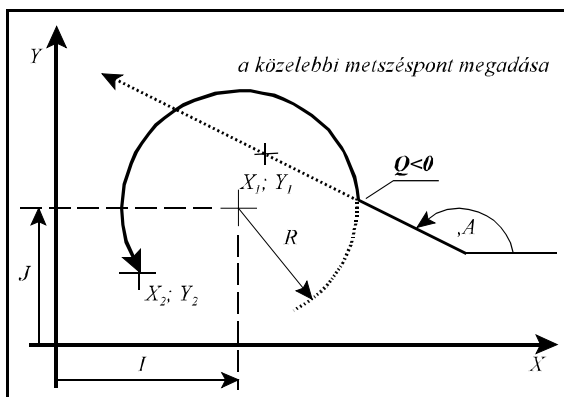
G19 G41 (G42)

N1 G1 ,A vagy

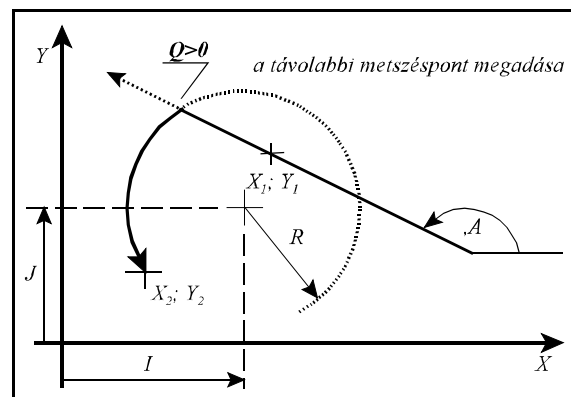
$Y_1 Z_1$

N2 G2 (G3) G90  $Y_2 Z_2 J$

K R Q



16.3.2-1 ábra

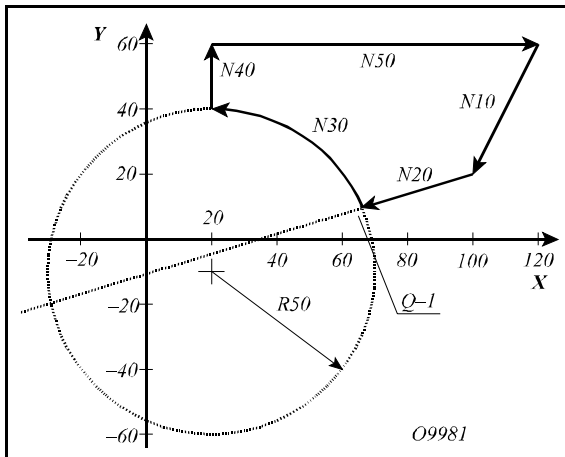


16.3.2-2 ábra

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.

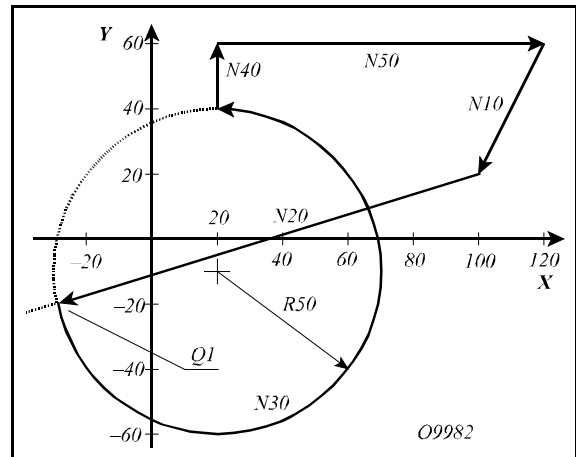
**Ha a cím értéke kisebb, mint nulla ( $Q < 0$ ) az egyenes irányába eső közelebbi, ha a cím értéke nagyobb, mint nulla ( $Q > 0$ ) az egyenes irányába eső távolabbi metszéspontot számolja ki. Az egyenesen való haladás irányát az irányszög jelöli ki.**

Nézzük a következő példát:



16.3.2-3 ábra

```
%O9981
N10 G17 G42 G0 X100 Y20 D0 S200 M3
N20 G1 X-30 Y-20
N30 G3 X20 Y40 I20 J-10 R50 Q-1
N40 G40 G0 Y60
N50 X120
N60 M30
%
```



16.3.2-4 ábra

```
%O9982
N10 G17 G42 G0 X100 Y20 D0 S200 M3
N20 G1 X-30 Y-20
N30 G3 X20 Y40 I20 J-10 R50 Q1
N40 G40 G0 Y60
N50 X120
N60 M30
%
```

Az N30 G3 körmondattól túlhatározott, mivel a középpont koordináták (I20 J-10 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 adtunk meg.

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

```
%O9983
N10 G17 G42 G0 X100 Y20 D0 S200 M3
N20 G1 X-30 Y-20 ,R15
N30 G3 X20 Y40 I20 J-10 R50 Q-1
N40 G40 G0 Y60
N50 X120
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 .

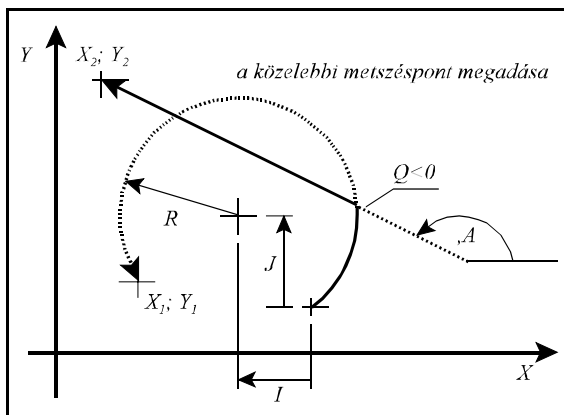
### 16.3.3 Kör és egyenes metszéspontja

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.

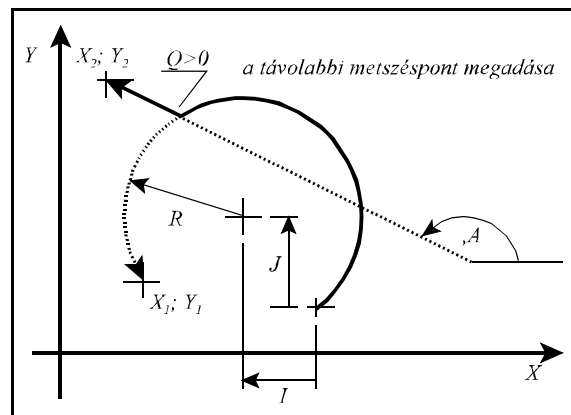
G17 G41 (G42)  
 N1 G2 (G3) X<sub>1</sub> Y<sub>1</sub> I J  
 vagy R  
 N2 G1 G90 X<sub>2</sub> Y<sub>2</sub> ,A Q

G18 G41 (G42)  
 N1 G2 (G3) X<sub>1</sub> Z<sub>1</sub> I K  
 vagy R  
 N2 G1 G90 X<sub>2</sub> Z<sub>2</sub> ,A Q

G19 G41 (G42)  
 N1 G2 (G3) Y<sub>1</sub> Z<sub>1</sub> J K  
 vagy R  
 N2 G1 G90 Y<sub>2</sub> Z<sub>2</sub> ,A Q



16.3.3-1 ábra

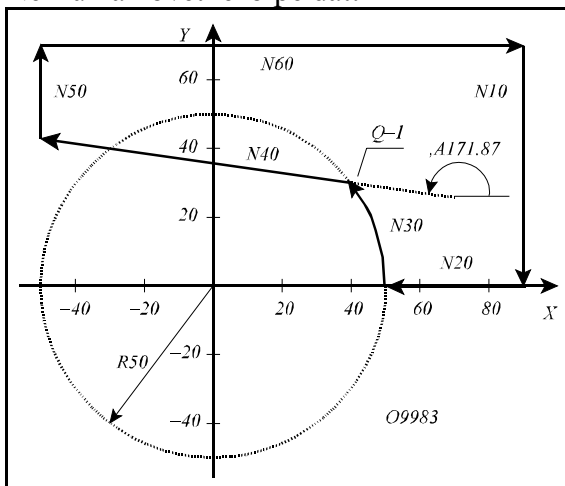


16.3.3-2 ábra

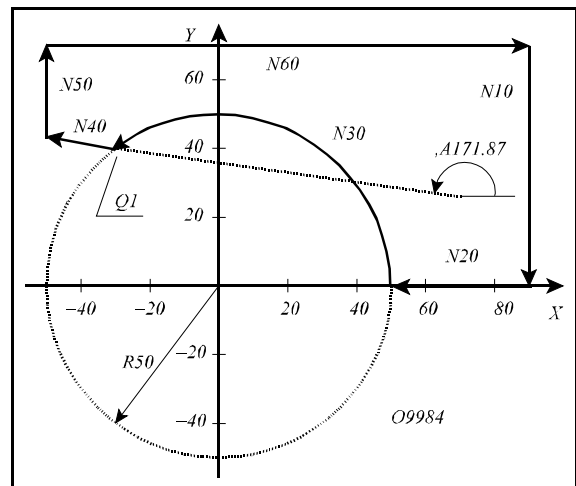
A metszéspontot mindig a G17, G18, G19 által kijelölt síkban számítja ki. Az első mondatot (N1), vagyis a kört egy tetszőleges pontjával (X<sub>1</sub>, Y<sub>1</sub>; X<sub>1</sub>, Z<sub>1</sub>; vagy Y<sub>1</sub>, Z<sub>1</sub>) és a középponti koordinátájával (I J; I K; vagy J K) adjuk meg, vagy a középponti koordináta helyett megadhatjuk a kör sugarát (R) is. A második mondatban (N2) az egyenest túlhatározzuk, vagyis megadjuk az egyenes végponti koordinátáit (X<sub>2</sub> Y<sub>2</sub>; X<sub>2</sub> Z<sub>2</sub>; vagy Y<sub>2</sub> Z<sub>2</sub>) és az egyenes irányszögét (A) is. 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 G17 G0 X90 Y0 M3 S200
N20 G42 G1 X50 D0
N30 G3 X-50 Y0 R50
N40 G1 X-50 Y42.857 ,A171.87 Q-1
N50 G40 G0 Y70
N60 X90
N70 M30
%
```

```
%O9984
N10 G17 G0 X90 Y0 M3 S200
N20 G42 G1 X50 D0
N30 G3 X-50 Y0 R50
N40 G1 X-50 Y42.857 ,A171.87 Q1
N50 G40 G0 Y70
N60 X90
N70 M30
%
```

Az N40 egyenes mondat túlhatározott, mert az egyenes végponti koordinátái (X-50 Y42.857) is és irányszöge is (A171.87) meg van adva. Ezért az előző, N30 mondatban programozott kör X-50 Y0 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 távolabbit (Q1).

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

```
%O9983
N10 G17 G0 X90 Y0 M3 S200
N20 G42 G1 X50 D0
N30 G3 X-50 Y0 R50 ,R15
N40 G1 X-50 Y42.857 ,A171.87 Q-1
N50 G40 G0 Y70
N60 X90
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)  $X_1 Y_1 I_1 J_1$   
 vagy  $X_1 Y_1 R_1$

N2 G2 (G3) G90  $X_2 Y_2 I_2$   
 $J_2 R_2 Q$

G18 G41 (G42)

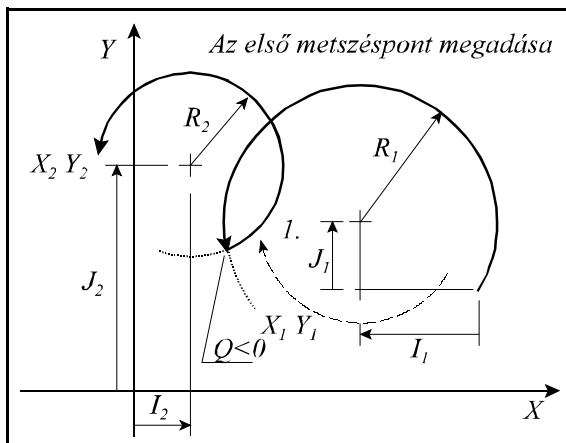
N1 G2 (G3)  $X_1 Z_1 I_1 K_1$   
 vagy  $X_1 Z_1 R_1$

N2 G2 (G3) G90  $X_2 Z_2 I_2$   
 $K_2 R_2 Q$

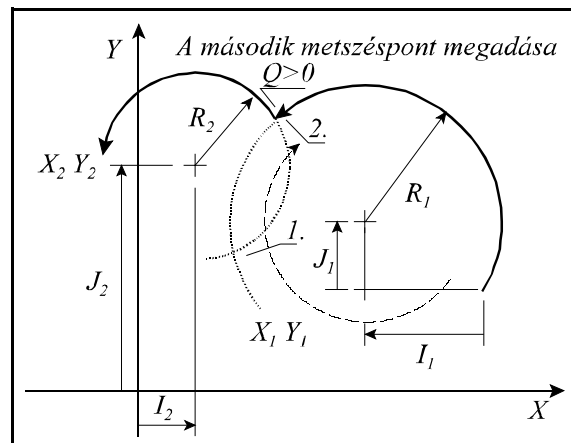
G19 G41 (G42)

N1 G2 (G3)  $Y_1 Z_1 J_1 K_1$   
 vagy  $Y_1 Z_1 R_1$

N2 G2 (G3) G90  $Y_2 Z_2 J_2$   
 $K_2 R_2 Q$



16.3.4-1 ábra



16.3.4-2 ábra

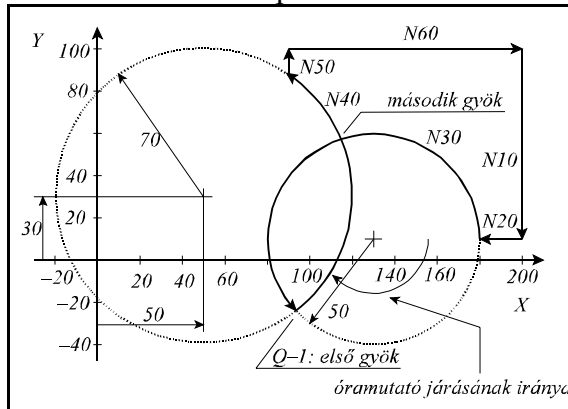
A metszéspontot mindig a G17, G18, G19 által kijelölt síkban számítja ki. Az első mondatot (N1) vagy a kör középponti koordinátájával ( $I_1 J_1$ ;  $I_1 K_1$ ;  $J_1 K_1$ ), vagy a kör sugarával ( $R_1$ ) adjuk meg. Ebben a mondatban a középponti koordináták értelmezése megegyezik a körmegadás alapértelmezésével, vagyis a kezdőponttól mért relatív távolság. A második mondatban (N2) megadott koordinátákat, így a **kör középpontját** meghatározó **I, J, K**



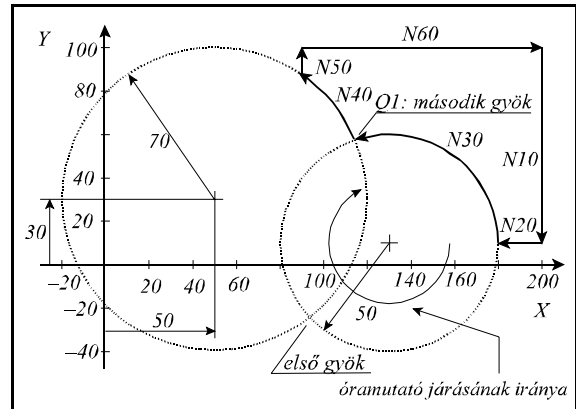
koordinátákat is mindig **abszolút** (G90) adatként értelmezi a vezérlő. Azt, hogy a kiadódó két metszéspont közül melyiket számolja ki a Q címen lehet megadni. Ha a cím értéke kisebb, mint nulla ( $Q < 0$ , pl: Q-1) az első, ha a cím értéke nagyobb, mint nulla ( $Q > 0$ , pl: Q1) a második metszéspontot 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 G17 G54 G0 X200 Y10 M3 S200
N20 G42 G1 X180 D1
N30 G3 X130 Y-40 R-50
N40 X90 Y87.446 I50 J30 R70 Q-1
N50 G40 G0 Y100
N60 X200
N70 M30
%
```

```
%O9986
N10 G17 G54 G0 X200 Y10 M3 S200
N20 G42 G1 X180 D1
N30 G3 X130 Y-40 R-50
N40 X90 Y87.446 I50 J30 R70 Q1
N50 G40 G0 Y100
N60 X200
N70 M30
%
```

Az N40 körmondattól túlhatározott, mert középpontjának koordinátái is (I50 J30 abszolút értéként), és sugara is (R70) meg vannak adva. Ezért az előző, N30 mondatban programozott kör X130 Y-40 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:

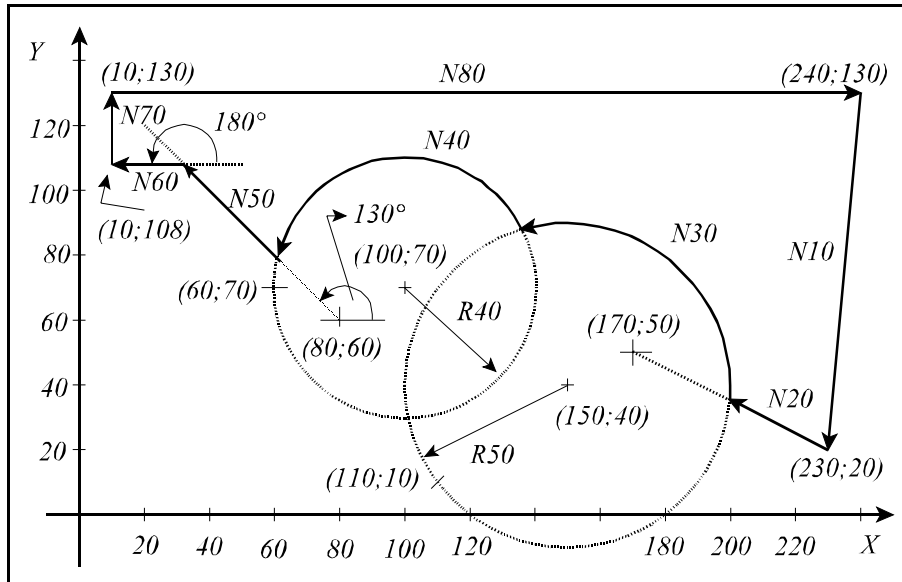
```
%O9986
N10 G17 G54 G0 X200 Y10 M3 S200
N20 G42 G1 X180 D1
N30 G3 X130 Y-40 R-50 ,R20
N40 X90 Y87.446 I50 J30 R70 Q1
N50 G40 G0 Y100
N60 X200
N70 M30
%
```

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 G17 G54 G0 G42 X230 Y20 D1 F300 S500 M3
N20 G1 X170 Y50
N30 G3 X110 Y10 I150 J40 R50 Q-1
N40 X60 Y70 I100 J70 R40 Q1
N50 G1 X80 Y60 ,A135 Q1
N60 X10 Y108 ,A180
N70 G40 G0 Y130
N80 X240
N90 M30
%
```

A fenti példában az N30, N40, N50, N60 mondat túlhatározott. Az N20 egyenest nem a programozott végpontjáig (X170 Y50) vezeti, mert az N30 körmondat túlhatározott, vagyis I J R címek mind ki vannak töltve, és Q címen megadtuk, hogy melyik metszéspontot keresse. Az N30 körmondatot sem a programozott végpontig (X110 Y10) 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 X10 Y108 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.*

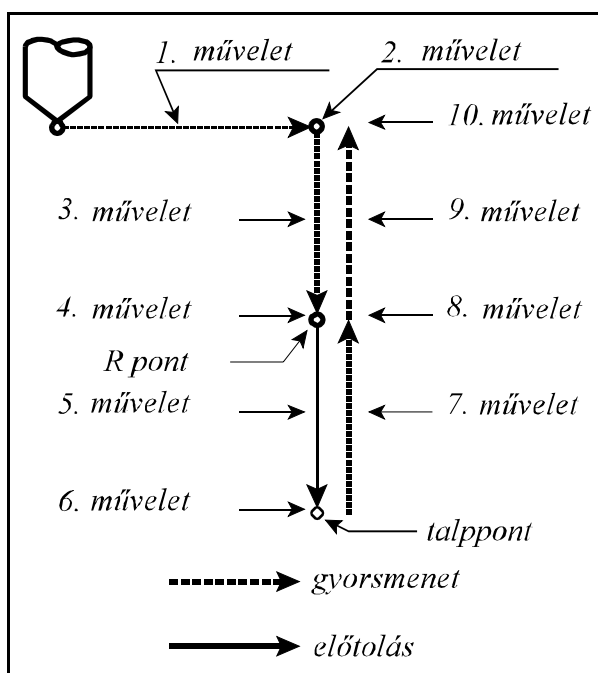
## 17 Fúróciklusok

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.



17-1 ábra

A fenti műveletek a fúróciklusok általános leírását adják, a konkrét esetekben műveletek elmaradhatnak.

A fúróciklusoknak van **pozicionálási síkja**, és **fúrótengelye**. A pozicionálási síkot és a fúrótengelyt a G17, G18, G19 síkválasztó utasítások jelölik ki.

G kód	pozicionálási sík	fúrótengely
G17	$X_p Y_p$ sík	$Z_p$
G18	$Z_p X_p$ sík	$Y_p$
G19	$Y_p Z_p$ sík	$X_p$

ahol:  $X_p$ : X, vagy azzal párhuzamos tengely

$Y_p$ : Y, vagy azzal párhuzamos tengely

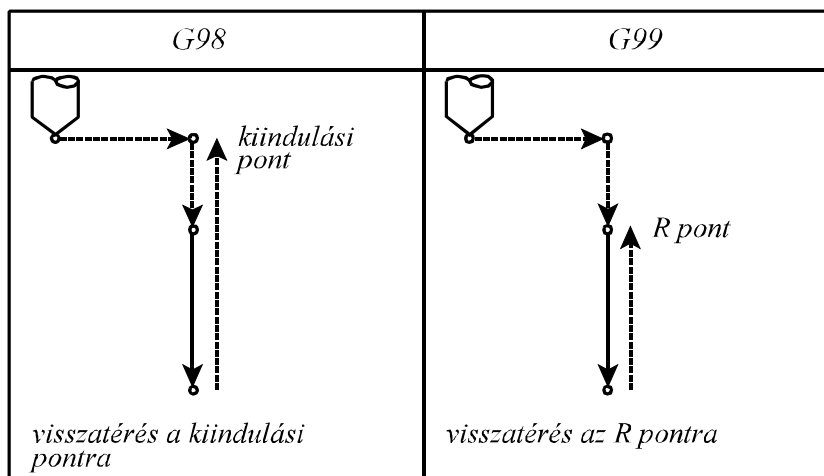
$Z_p$ : Z, vagy azzal párhuzamos tengely

U, V, W tengelyeket akkor tekinti párhuzamos tengelyeknek, ha a paramétermezőben annak vannak definiálva.

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, reset 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.



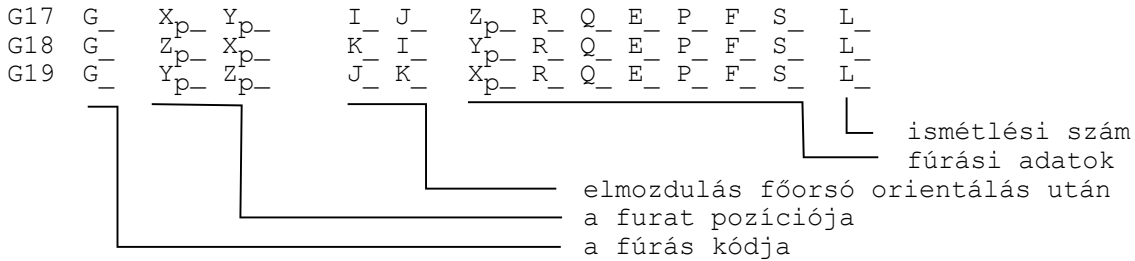
17-2 ábra

A **fúróciklusok kódjai**: G73, G74, G76, 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 **fúró ciklusok**ban használt **címek** és értelmezésük:



**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.

Amíg a ciklusállapot be van kapcsolva, a G73, G74, G76, 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 Y0 Z50 R150
N3 X100 Y30 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 Y0 Z50 R150
N3 X100 Y30 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<sub>p</sub>, Y<sub>p</sub>, Z<sub>p</sub>**

A beírt koordinátaértékek közül a kiválasztott síkban lévőket veszi a furat pozíciójának.

A beírt értékek lehetnek inkrementális, vagy abszolút, derékszögű vagy polárkoordinátában megadott értékek, dimenziójuk pedig metrikus vagy inches.

A beírt koordinátaértékekre érvényesek a tükrözési, elforgatási, és léptékezési parancsok.

A furat pozíciójára a vezérlés gyorsmeneti pozícionálással 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 ki-esztergáló 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

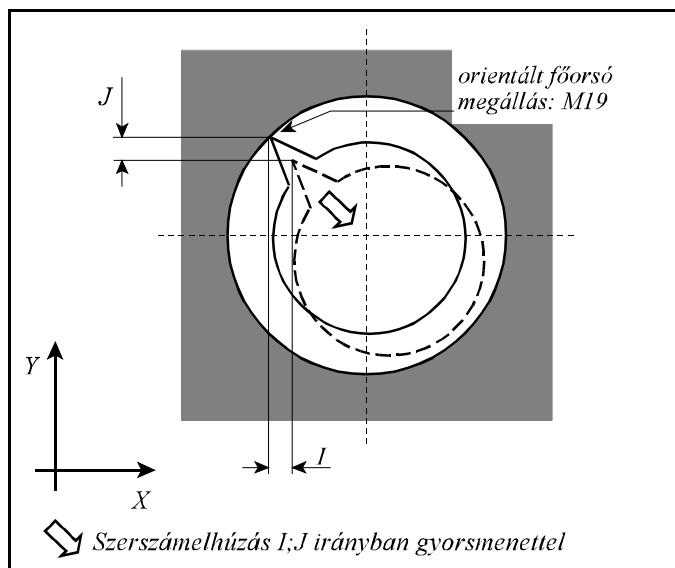
G18: K, I

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, elforgatási, vagy léptékezési parancsok. I, J és K öröklődő értékek. G80, vagy az interpolációs csoport kódjai értékeit törlik. Az elhúzás gyorsmenettel történik.

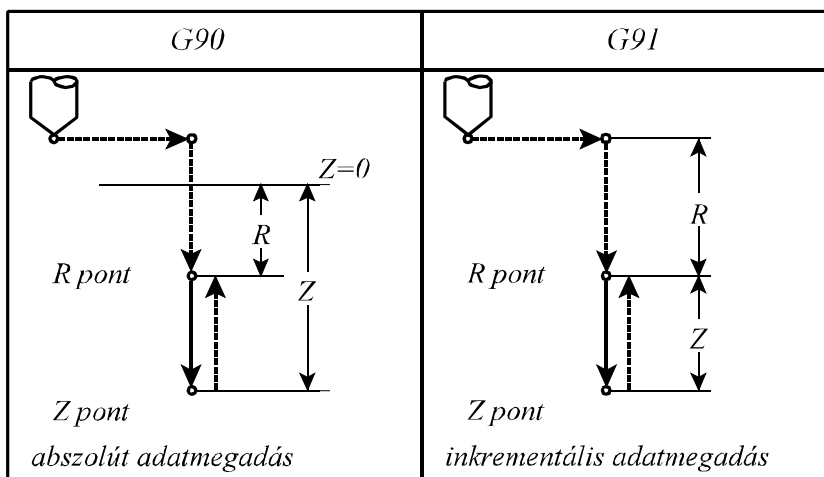


17-3 ábra

Fúrési adatok:

A furat talppontja:  $X_p, Y_p, Z_p$

A furat talppontját a fúrotengely címen 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.



17-4 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

G73-as 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

G73-as 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 fűrési kódok és a ciklusváltozók öröklődésére:**

N1 G17 G0 Z\_ M3

N2 G81 X\_ Y\_ Z\_ R\_ F\_

A ciklus üzemmód kezdetén a fűrési adatok (Z, R) meghatározása kötelező

N3 X\_

Mivel az N2 mondatban a fűrési 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 fűrő ebben az irányban mozog, majd ugyanazt a furatot fűrja, mint az N2 mondatban.

N4 G82 Y\_ Z\_ P\_

A furat pozíciója Y irányban mozog. A fűrás módsere 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 Y\_ Z\_ R\_ P\_ M3

Mivel az N5 mondatban törölődtek a fűrési adatok G80 utasítás hatására Z, R, és P értékeket újra meg kell adni.



```
N7 G0 X_ Y_
```

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.

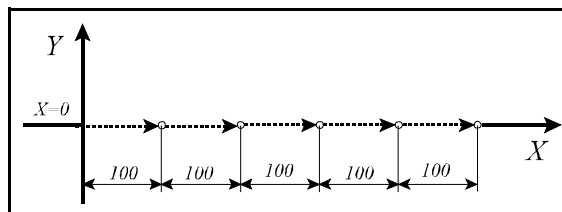
```
N1 G90 G17 G0 X0 Y0 Z100 M3
N2 G91 G81 X100 Z-40 R-97 F50 L5
```

A fenti utasítások hatására a vezérlő az X tengely mentén 100 mm távolságra egymástól 5 db egyforma furatot fúr. Az első furat pozíciója  $X=100, Y=0$ .

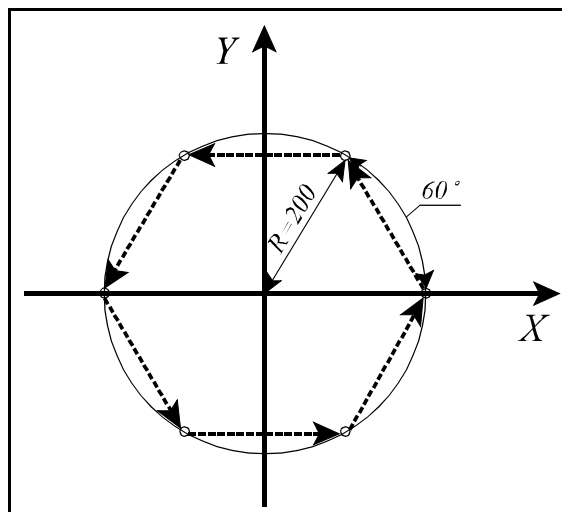
A furat pozíciója G91 hatására inkrementálisan lett megadva. Ha abszolút adatként (G90) adtuk volna meg, az X100, Y0 koordinátájú ponton ötször egymás után hajtotta volna végre a műveletet.

```
N1 G90 G17 G16 G0 X200Y-60 Z50
N2 G81 YI60 Z-40 R3 F50 L6
```

A fenti utasítások hatására a vezérlő egy 200 mm-es lyukkörön 60 fokként 6 db furatot fúr. Az első furat pozíciója az  $X=200, Y=0$  koordinátájú pontra esik.



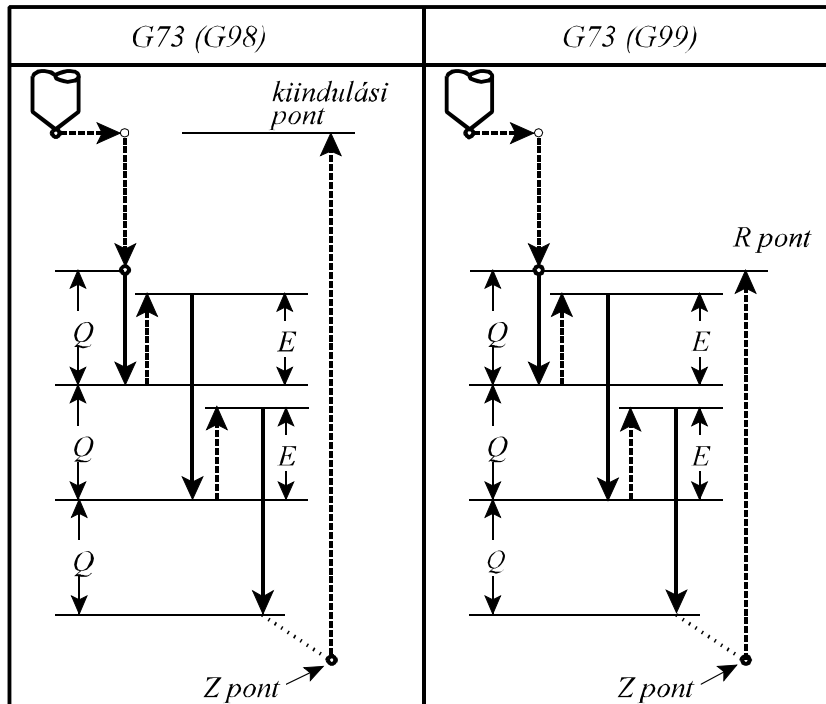
17-5 ábra



17-6 ábra

## 17.1 A fúróciklusok részletes leírása.

## 17.1.1 Nagysebességű mélyfúróciklus (G73)



17.1.1-1 ábra

A ciklusban felhasznált változók:

G17 **G73** X<sub>p</sub> Y<sub>p</sub> Z<sub>p</sub> R Q E F L

G18 **G73** Z<sub>p</sub> X<sub>p</sub> Y<sub>p</sub> R Q E F L

G19 **G73** Y<sub>p</sub> Z<sub>p</sub> X<sub>p</sub> R Q E F L

A ciklus műveletei:

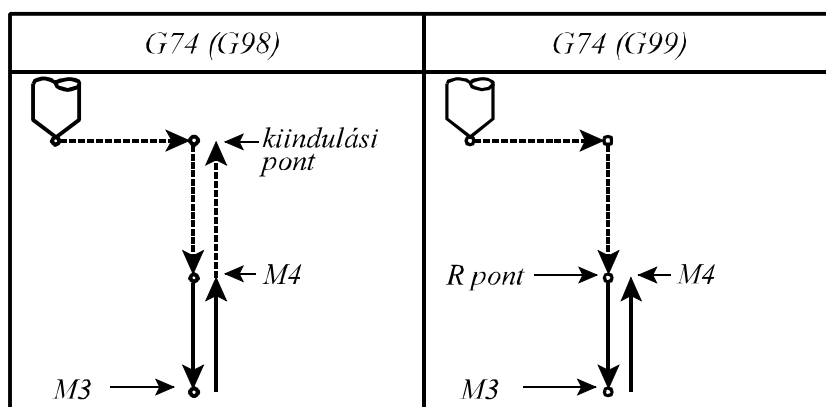
1. művelet: pozicionálás a kiválasztott síkban gyorsmenettel
2. művelet: –
3. művelet: gyorsmeneti mozgás az R pontig (megközelítési pont)
4. művelet: –
5. művelet: fúrás a talppontig F előtolással
6. művelet: –
7. művelet: G99 esetén: visszahúzás az R pontig gyorsmenettel
8. művelet: –
9. művelet: G98 esetén: visszahúzás a kiindulási pontig gyorsmenettel
10. művelet: –

Az 5. fúrási művelet leírása:

- a Q címen megadott fogásmélységet előtolással befúrja az anyagba,
- az E címen, vagy a *RETG73* paraméteren megadott értékkel gyorsmenettel visszahúz,
- az előző befúrás talppontjától számítva Q mélységet ismételten befúr,
- az E címen megadott értékkel gyorsmenettel visszahúz.

Az eljárás a Z címen megadott talppontig folytatódik.

## 17.1.2 Balmenetfúró ciklus (G74)



17.1.2-1 ábra

A ciklus csak kiegyenlítőbetéttel ellátott menetfúróval alkalmazható.

A ciklusban felhasznált változók:

G17 **G74** X<sub>p</sub> Y<sub>p</sub> Z<sub>p</sub> R (P) F L

G18 **G74** Z<sub>p</sub> X<sub>p</sub> Y<sub>p</sub> R (P) F L

G19 **G74** Y<sub>p</sub> Z<sub>p</sub> X<sub>p</sub> R (P) F L

A ciklus indítása előtt M4 (óramutató járásával ellentétes) főorsó forgásirányt kell bekapcsolni illetve programozni.

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:

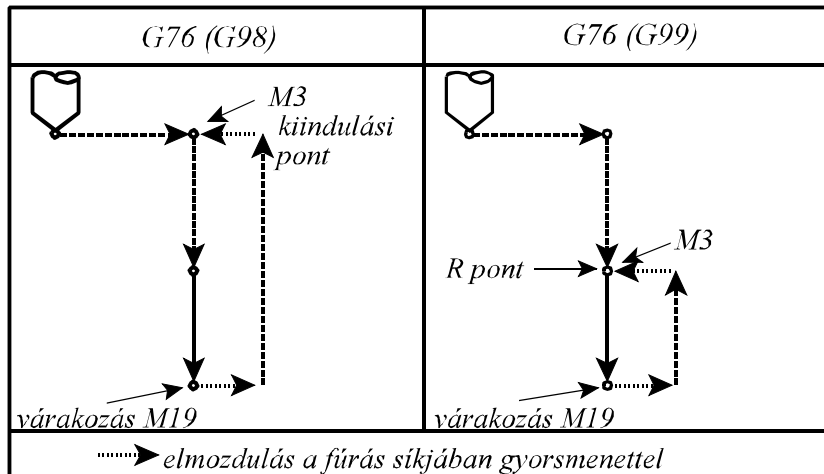
$$F=P$$

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: 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: –

## 17.1.3 Kiesztergálás automatikus szerszámelhúzással (G76)



17.1.3-1 ábra

A G76 ciklust csak akkor lehet használni, ha a főorsó orientálás be van építve a szerszámgépbe. A vezérlő számára ezt a tényét az *ORIENTI* 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 **G76** X<sub>p</sub>\_\_ Y<sub>p</sub>\_\_ I\_\_ J\_\_ Z<sub>p</sub>\_\_ R\_\_ P\_\_ F\_\_ L\_\_  
 G18 **G76** Z<sub>p</sub>\_\_ X<sub>p</sub>\_\_ K\_\_ I\_\_ Y<sub>p</sub>\_\_ R\_\_ P\_\_ F\_\_ L\_\_  
 G19 **G76** Y<sub>p</sub>\_\_ Z<sub>p</sub>\_\_ J\_\_ K\_\_ X<sub>p</sub>\_\_ 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
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

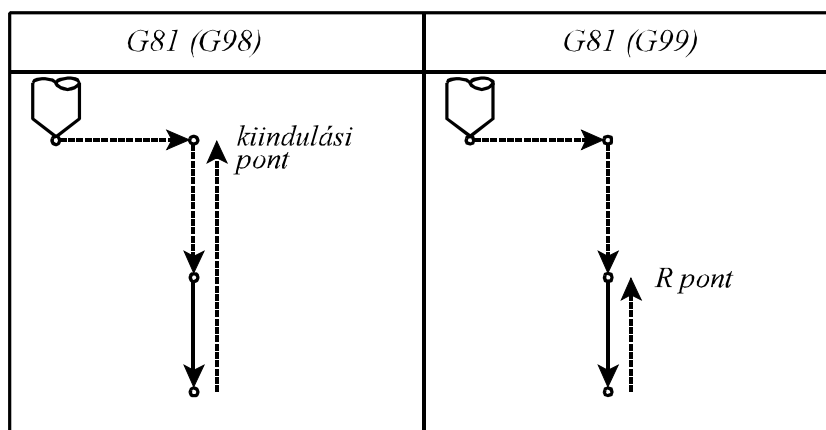
### 17.1.4 A ciklusállapot kikapcsolása (G80)

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.

### 17.1.5 Fúróciklus, kiemelés gyorsmenettel (G81)



17.1.5-1 ábra

A ciklusban felhasznált változók:

G17 **G81** X<sub>p</sub>\_\_ Y<sub>p</sub>\_\_ Z<sub>p</sub>\_\_ R\_\_ F\_\_ L\_\_

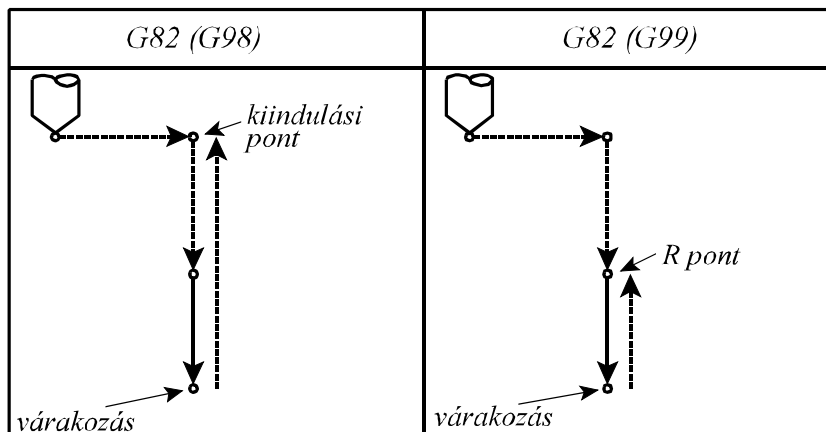
G18 **G81** Z<sub>p</sub>\_\_ X<sub>p</sub>\_\_ Y<sub>p</sub>\_\_ R\_\_ F\_\_ L\_\_

G19 **G81** Y<sub>p</sub>\_\_ Z<sub>p</sub>\_\_ X<sub>p</sub>\_\_ 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: G99 esetén: visszahúzás az R pontig, gyorsmenettel
8. művelet: –
9. művelet: G98 esetén: gyorsmeneti visszahúzás a kiindulási pontig
10. művelet: –

## 17.1.6 Fúróciklus várakozással, kiemelés gyorsmenettel (G82)



17.1.6-1 ábra

A ciklusban felhasznált változók:

G17 **G82** X<sub>p</sub>\_\_ Y<sub>p</sub>\_\_ Z<sub>p</sub>\_\_ R\_\_ P\_\_ F\_\_ L\_\_

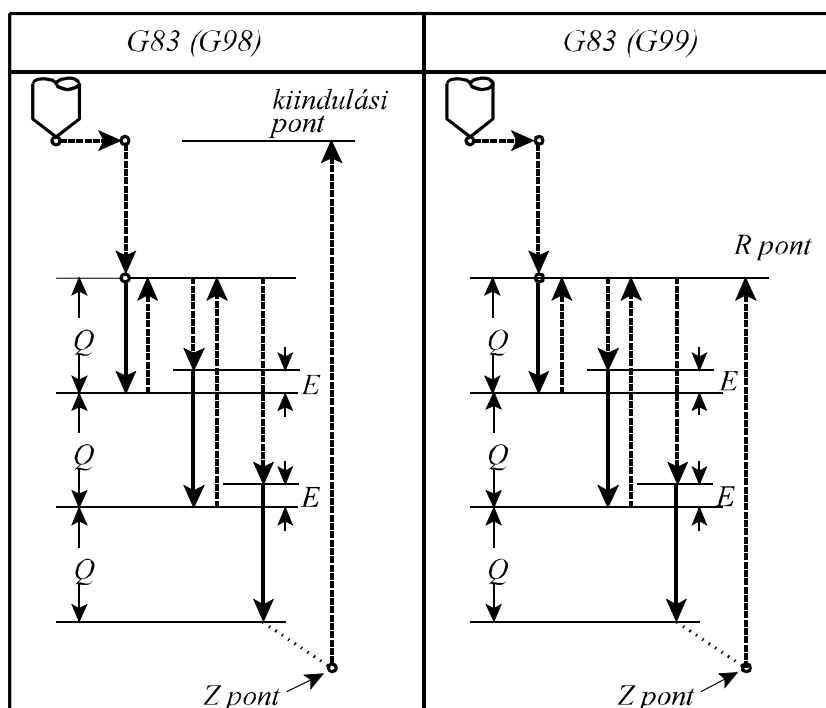
G18 **G82** Z<sub>p</sub>\_\_ X<sub>p</sub>\_\_ Y<sub>p</sub>\_\_ R\_\_ P\_\_ F\_\_ L\_\_

G19 **G82** Y<sub>p</sub>\_\_ Z<sub>p</sub>\_\_ X<sub>p</sub>\_\_ 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: –

## 17.1.7 Mélyfúróciklus (G83)



17.1.7-1 ábra

A ciklusban felhasznált változók:

G17 **G83** X<sub>p</sub>\_\_ Y<sub>p</sub>\_\_ Z<sub>p</sub>\_\_ R\_\_ Q\_\_ E\_\_ F\_\_ L\_\_  
 G18 **G83** Z<sub>p</sub>\_\_ X<sub>p</sub>\_\_ Y<sub>p</sub>\_\_ R\_\_ Q\_\_ E\_\_ F\_\_ L\_\_  
 G19 **G83** Y<sub>p</sub>\_\_ Z<sub>p</sub>\_\_ X<sub>p</sub>\_\_ R\_\_ Q\_\_ E\_\_ 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: visszahúzás a kiindulási pontig gyorsmenettel
10. művelet: –

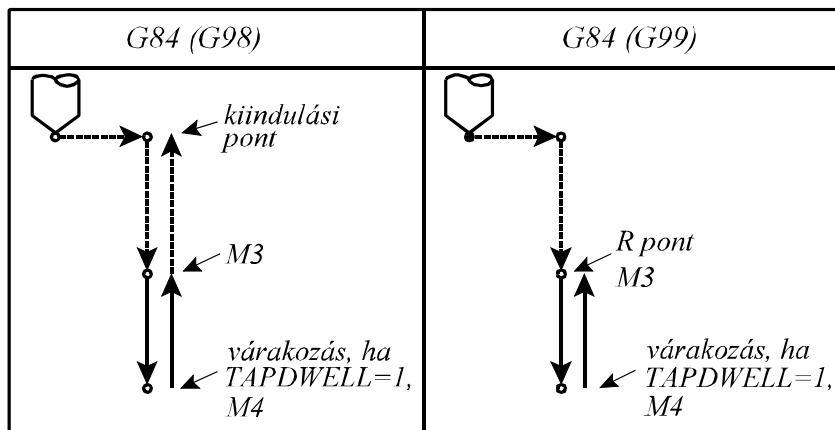
Az 5. fúrási művelet leírása:

- 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 ismételtlen 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.

## 17.1.8 Menetfúró ciklus (G84)



17.1.8-1 ábra

A ciklus csak kiegyenlítőbetéttel ellátott menetfúróval alkalmazható.

A ciklusban felhasznált változók:

G17 **G84** X<sub>p</sub> Y<sub>p</sub> Z<sub>p</sub> R (P) F L

G18 **G84** Z<sub>p</sub> X<sub>p</sub> Y<sub>p</sub> R (P) F L

G19 **G84** Y<sub>p</sub> Z<sub>p</sub> X<sub>p</sub> 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: –



### 17.1.9 Menetfúró ciklus kiegyenlítő betét nélkül (G84.2, G84.3)

Menetfúrás esetén a fúró tengely elötolása és a főorsó fordulat hányadosának egyenlőnek kell lennie a menetfúró menetemelkedésével. Másképpen fogalmazva menetfúrásnál ideális esetben

az alábbi hányadosnak pillanatról pillanatra állandónak kell lenni:  $P = \frac{F}{S}$

ahol: P: a menetemelkedés (mm/ford, vagy inch/ford),  
F: elötolás (mm/min, vagy inch/min),  
S: főorsó fordulatszám (ford/min).

A G74-es balmenet, és a G84-es jobbmenet fúró ciklusban a főorsó fordulatszáma és a fúró tengely elötolása egymástól teljesen függetlenül vezérlődik. A fenti feltétel ennek megfelelően nem teljesülhet pontosan. Különösen igaz ez a furat alján, ahol a fúró tengely elötolásának 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 menetfúró egy rugós kiegyenlítőbetéttel tesszük be a főorsóba, amely kiegyenlíti az  $\frac{F}{S}$  hányados értéké-

ben bekövetkező ingadozást.

Más a vezérlés elve a kiegyenlítőbetét kiküszöbölését lehetővé tevő G84.2, G84.3 fúró ciklusoknál. 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 fúró tengely 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 fúrása kiegyenlítő betét nélkül

**G84.3:** balmenet fúrása 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<sub>p</sub> \_ Y<sub>p</sub> \_ Z<sub>p</sub> \_ R \_ F \_ S \_ L \_

G18 **G84.** \_ Z<sub>p</sub> \_ X<sub>p</sub> \_ Y<sub>p</sub> \_ R \_ F \_ S \_ L \_

G19 **G84.** \_ Y<sub>p</sub> \_ Z<sub>p</sub> \_ X<sub>p</sub> \_ R \_ F \_ S \_ L \_

A ciklus végén a főorsó álló állapotba kerül, szükség esetén újraindításáról a programozónak kell gondoskodni.

Az elötolás és főorsó fordulatszám 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 fordulata ford/perc dimenzióban

Ebben az esetben az elmozdulás és az előtolás a fúró tengely és a főorsó mentén a következő (Z-t tételezve fel fúró tengelynek):

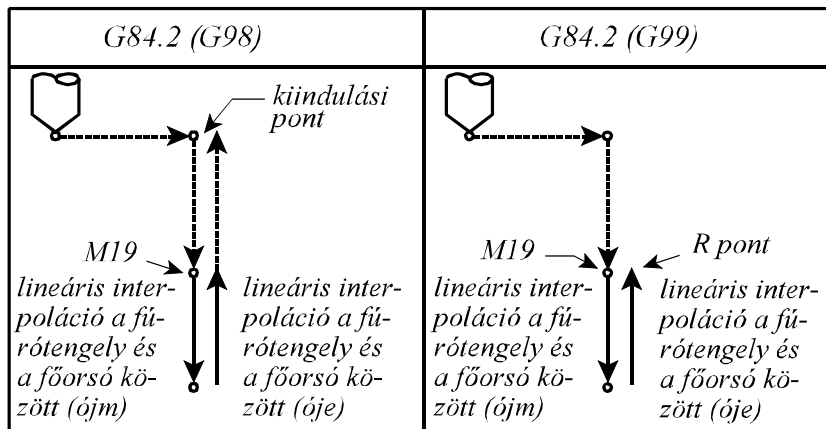
	elmozdulás	előtolás
Z	$z = R \text{ pont és talppont távolsága}$	$F_z = F \left( \frac{\text{mm}}{\text{min}} \text{ vagy } \frac{\text{inch}}{\text{min}} \right)$
S	$s = \frac{z \cdot S \cdot 360}{F} \text{ (fok)}$	$F_s = S \cdot 360 \left( \frac{\text{fok}}{\text{min}} \right)$

– 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 fúró tengely és a főorsó mentén a következő (a Z tengelyt feltételezve fúró tengelynek):

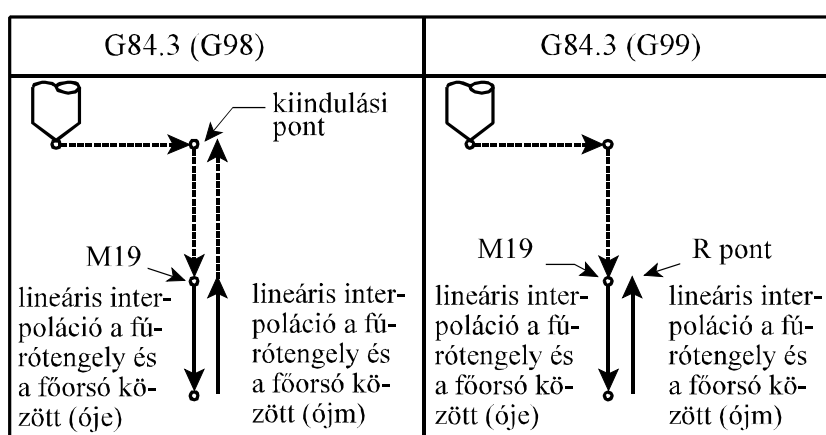
	elmozdulás	előtolás
Z	$z = R \text{ pont és talppont távolsága}$	$F_z = F \cdot S \left( \frac{\text{mm}}{\text{min}} \text{ vagy } \frac{\text{inch}}{\text{min}} \right)$
S	$s = \frac{z \cdot 360}{F} \text{ (fok)}$	$F_s = S \cdot 360 \left( \frac{\text{fok}}{\text{min}} \right)$



17.1.9-1 ábra

A ciklus műveletei G84.2 estén:

1. művelet: pozícionálás 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 fúrótengely és a főorsó között óramutató járásával megegyező (+) főorsó forgásirányban
6. művelet: -
7. művelet: lineáris interpoláció a fúrótengely és a főorsó között óramutató járásával ellentétes (-) főorsó forgásirányban
8. művelet: -
9. művelet: G98 esetén: gyorsmeneti visszahúzás a kiindulási pontig
10. művelet: -

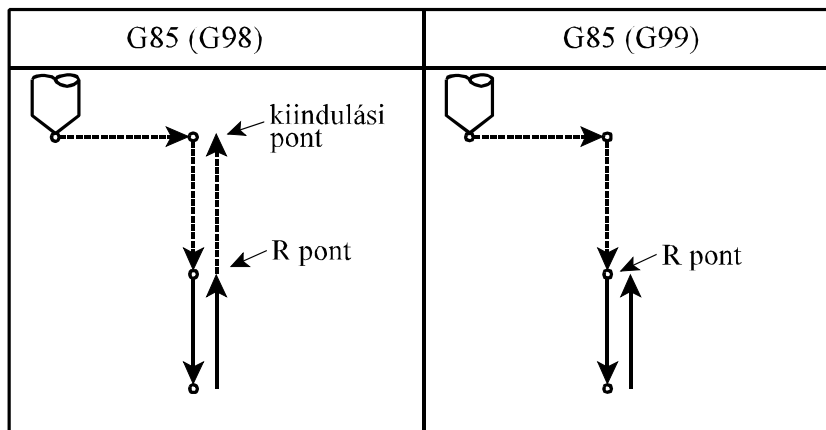


17.1.9-2 ábra

A ciklus műveletei G84.3 estén:

1. művelet: pozícionálás 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 fúrótengely és a főorsó között óramutató járásával ellentétes (-) főorsó forgásirányban
6. művelet: -
7. művelet: lineáris interpoláció a fúrótengely és a főorsó között óramutató járásával megegyező (+) főorsó forgásirányban
8. művelet: -
9. művelet: G98 esetén: gyorsmeneti visszahúzás a kiindulási pontig
10. művelet: -

## 17.1.10 Fúróciklus, kiemelés előtolással (G85)



17.1.10-1 ábra

A ciklusban felhasznált változók:

G17 **G85** X<sub>p</sub>\_\_ Y<sub>p</sub>\_\_ Z<sub>p</sub>\_\_ R\_\_ F\_\_ L\_\_

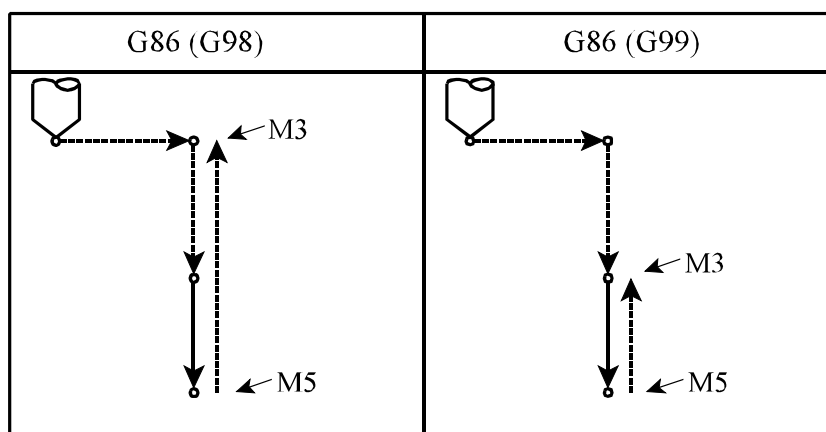
G18 **G85** Z<sub>p</sub>\_\_ X<sub>p</sub>\_\_ Y<sub>p</sub>\_\_ R\_\_ F\_\_ L\_\_

G19 **G85** Y<sub>p</sub>\_\_ Z<sub>p</sub>\_\_ X<sub>p</sub>\_\_ 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: –

## 17.1.11 Fúróciklus, gyorsmeneti kiemelés álló főorsóval (G86)



17.1.11-1 ábra

A ciklusban felhasznált változók:

G17 **G86** X<sub>p</sub>\_\_ Y<sub>p</sub>\_\_ Z<sub>p</sub>\_\_ R\_\_ F\_\_ L\_\_

G18 **G86** Z<sub>p</sub>\_\_ X<sub>p</sub>\_\_ Y<sub>p</sub>\_\_ R\_\_ F\_\_ L\_\_

G19 **G86** Y<sub>p</sub>\_\_ Z<sub>p</sub>\_\_ X<sub>p</sub>\_\_ R\_\_ F\_\_ L\_\_

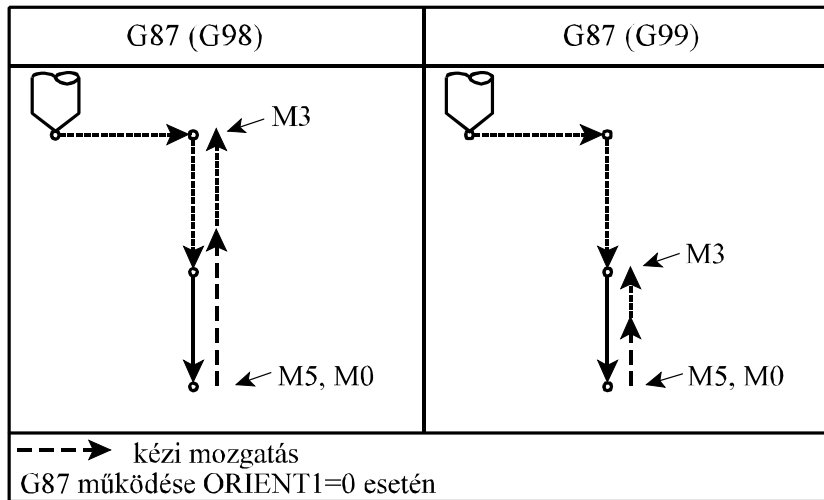
A ciklus indításakor M3-as forgásirányt kell adni a főorsónak.

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

### 17.1.12 Fúróciklus, kézi működtetés a talpponton/ Kiesztergálás visszafelé, automatikus szerszámelhúzással (G87)

A ciklust a vezérlő kétféleképp hajtja végre:



17.1.12-1 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.

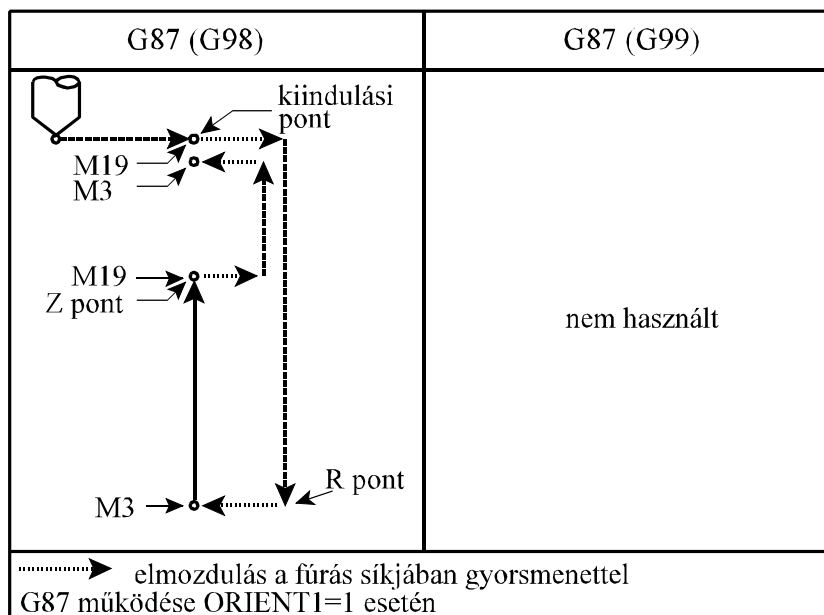
A ciklusban felhasznált változók:

G17 **G87** X<sub>p</sub> Y<sub>p</sub> Z<sub>p</sub> R F L  
 G18 **G87** Z<sub>p</sub> X<sub>p</sub> Y<sub>p</sub> R F L  
 G19 **G87** Y<sub>p</sub> Z<sub>p</sub> X<sub>p</sub> 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 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: 98 esetén: START után gyorsmeneti visszahúzás a kiindulási pontig
10. művelet: G98 esetén: főorsó újraindítás: M3



17.1.12-2 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<sub>p</sub>\_\_ Y<sub>p</sub>\_\_ I\_\_ J\_\_ Z<sub>p</sub>\_\_ R\_\_ F\_\_ L\_\_

G18 **G87** Z<sub>p</sub>\_\_ X<sub>p</sub>\_\_ K\_\_ I\_\_ Y<sub>p</sub>\_\_ R\_\_ F\_\_ L\_\_

G19 **G87** Y<sub>p</sub>\_\_ Z<sub>p</sub>\_\_ J\_\_ K\_\_ X<sub>p</sub>\_\_ R\_\_ F\_\_ L\_\_

A ciklus indításakor M3-as forgásirányt kell adni a főorsónak.

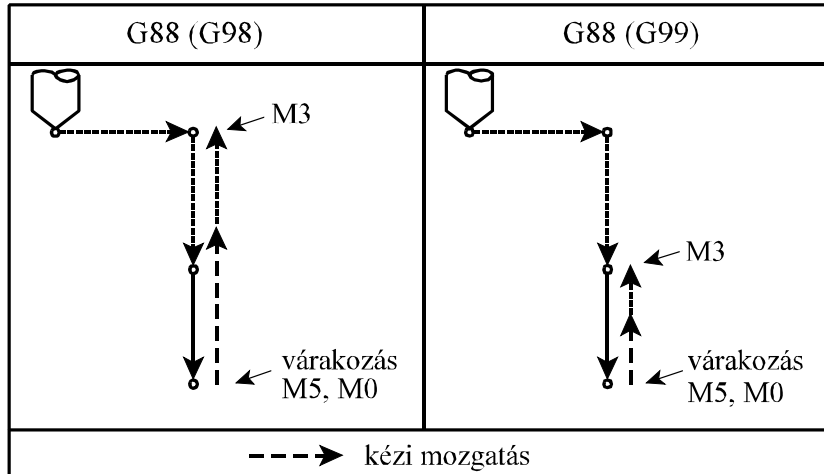
A ciklus műveletei:

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.

### 17.1.13 Fúróciklus, várakozás után kézi működtetés a talpponton (G88)



17.1.13-1 ábra

A ciklusban felhasznált változók:

G17 **G88** X<sub>p</sub> Y<sub>p</sub> Z<sub>p</sub> R P F L

G18 **G88** Z<sub>p</sub> X<sub>p</sub> Y<sub>p</sub> R P F L

G19 **G88** Y<sub>p</sub> Z<sub>p</sub> X<sub>p</sub> R P F L

A ciklus indításakor M3-as forgásirányt kell adni a főorsónak.

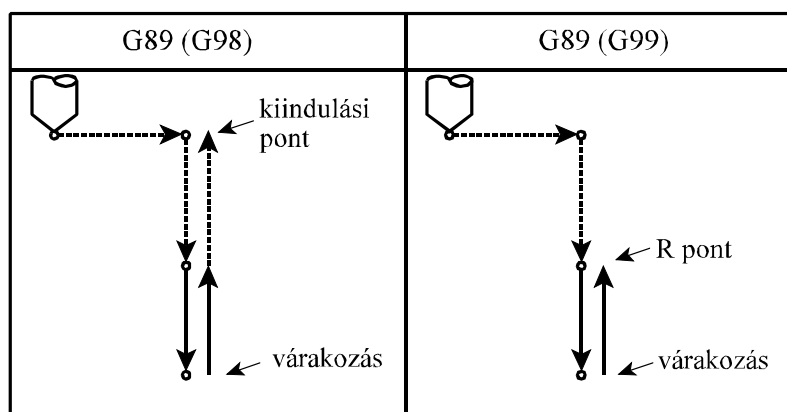
A ciklus műveletei:

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 leállítas: 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.



## 17.1.14 Fúróciklus, talpponton várakozás, kiemelés előtolással (G89)



17.1.14-1 ábra

A ciklusban felhasznált változók:

G17 **G89** X<sub>p</sub> Y<sub>p</sub> Z<sub>p</sub> R P F L

G18 **G89** Z<sub>p</sub> X<sub>p</sub> Y<sub>p</sub> R P F L

G19 **G89** Y<sub>p</sub> Z<sub>p</sub> X<sub>p</sub> 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.

## 17.2 Megjegyzések a fúróciklusok használatához

- Ciklusüzem módban ha egy G kód nélküli mondat a következő címek valamelyikét tartalmazza, a fúróciklus végrehajtásra kerül:  
X<sub>p</sub>, Y<sub>p</sub>, Z<sub>p</sub>, vagy R  
Ellenkező esetben a fúróciklus nem hajtódik végre.
- Ciklusüzem módban G04 P várakozási mondatot programozva a parancs végrehajtódik a programozott P szerint, de a várakozásra vonatkozó ciklusváltozó **nem** törlődik, és **nem** íródik át.
- I, J, K, Q, E, P értékét olyan mondatban kell megadni, ahol fúrás is történik, különben **nem** tárolódnak el az értékek.

A fentiek illusztrálására tekintsük a következő példát:

G81	X_ Y_ 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 G74, G84 ciklusok 5., 6., és 7. műveletében.
- Ha ciklusmondatban G43, G44, G49 kerül programozásra, vagy új H értéket adunk meg, a hosszkorrekciót a 3. műveletben, mindig a fúrótengely mentén veszi figyelembe.
- G45, ..., G48 utasítások a fúróciklusban nem kerülnek végrehajtásra.

## 18 MÉRŐFUNKCIÓK

### 18.1 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 törlőjel (pl. egy mérő-tapintó jele) be nem érkezik, vagy a v koordinátákon megadott végponti pozíciót el nem érte a vezérlés. A törlőjel 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.

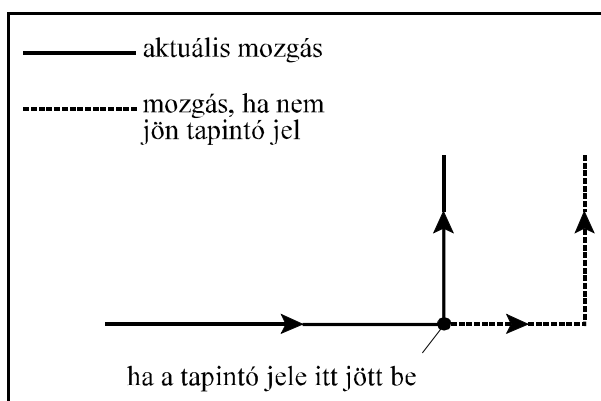
Ha a G31 utasításban szintaktikai hibát talál a vezérlés 3051 G22, G28, ... G31, G37 hibajelzést ad.

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



18.1-1 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 (G43, G44) és,
- az aktuális szerszámeltolás (G45...G48) 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.

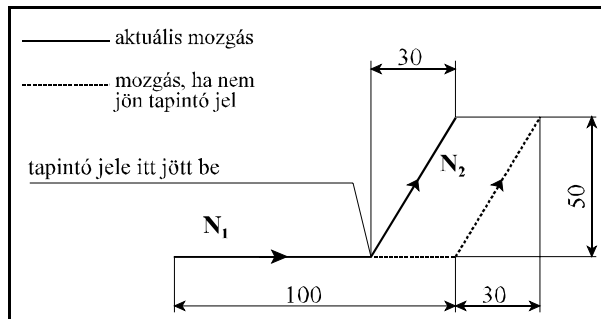
A mondat végrehajtása csak G40 állapotban lehetséges. G41 vagy G42 állapot esetén G31 programozása 3054 G31 ROSSZ ÁLLAPOTBAN hibajelzést vált ki. Szintén ezt a hibajelzést

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 X100
N2 X30 Y50
```

N1 mondatban elindít egy inkrementális mozgást X irányban. Ha a külső jel bejövetele után a vezérlő az X=86.7 koordinátájú ponton áll meg, ettől a ponttól számítva lép inkrementálisan 30-at x irányban, 50-et az Y irányban az N2 mondatban.

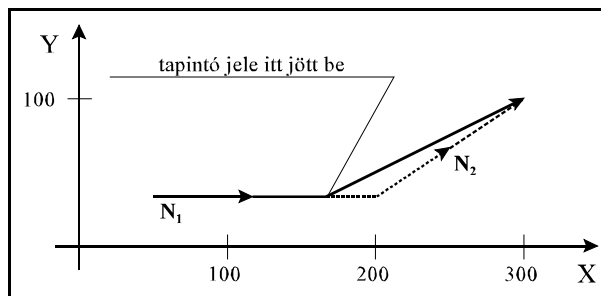


18.1-2 ábra

Ha abszolút adatmegadást programoztunk a mozgás a következőképp alakul:

```
N1 G31 G90 X200
N2 X300 Y100
```

N1 mondat elindít X irányban egy mozgást az X=200 koordinátájú pontra. Ha a külső jel bejövetele után a vezérlő az X=167 koordinátájú ponton áll meg, az N2 mondatban az X irányú elmozdulás X=300-167, azaz X=133 lesz.



18.1-3 ábra

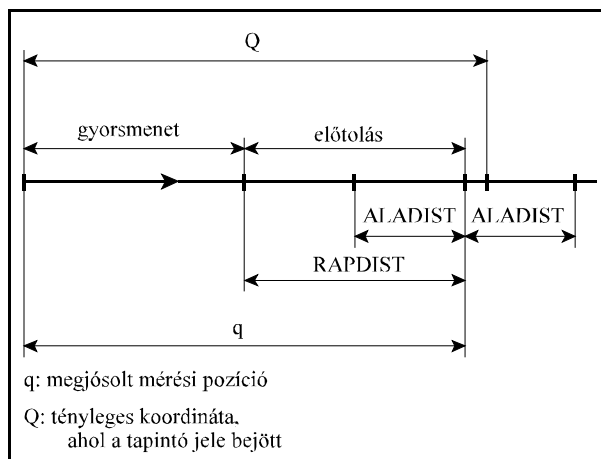
## 18.2 Automatikus szerszámhossz mérés (G37)

A

### G37 q

utasítás hatására a q koordinátán megadott irányban gyorsmenettel elindul a mozgás. q értéke mindig abszolút adatként kerül értelmezésre.

A mozgás a q - *RAPDIST* pozícióig megy gyorsmenettel, ahol *RAPDIST* paraméteren beállított érték. 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 a q címen programozott pozíció (megjósolt mérési pozíció) *ALADIST* sugarú környezetén kívül jön meg a tapintó jele.



18.2-1 ábra

Ha a mérés sikerrel járt, és a tapintó jele a Q koordinátájú ponton jött be, a vezérlés az érvényben lévő

– H korrekciós regiszter kopás értékéhez

- hozzáadja (ha az *ADD* paraméter =1) a Q-q különbséget,
- vagy kivonja abból (ha az *ADD* paraméter =0).

A mérés megkezdése előtt a megfelelő H értéket és a hosszkorrekciót be kell kapcsolni.

- A G37 utasítás egylovétű.
- A G37 ciklus mindig az aktuális munkadarab koordináta-rendszerben 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 hosszkorrekcióra utaló G kódot (G43, G44, G49) a G37 mondatban nem lehet megadni. Ellenkező esetben 3055 G37 ROSSZ ÁLLAPOTBAN hibajelzést ad.
- Szintén a fenti hibajelzést adja, ha G51, G51.1, G68 állapot, vagy, ha G16 állapot van érvényben.

A G37 funkció végrehajtása közben előjövő hibaüzenetek a következők:

- 3103 HATÁRON KÍVÜL üzenet, ha a G37 mondatban programozott végpozíció *ALADIST* sugarú környezetén belül nem jön meg a tapintó jele.

## 19 Biztonsági funkciók

### 19.1 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,

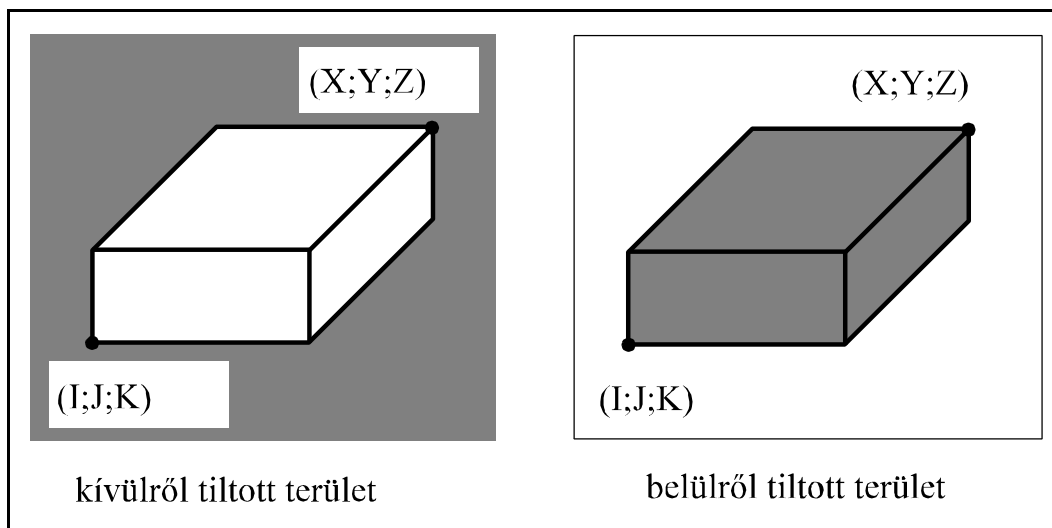
A következő feltételeknek kell teljesülni a megadott adatokra:

$$X \geq I, Y \geq J, Z \geq K$$

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.



19.1-1 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 számítás-

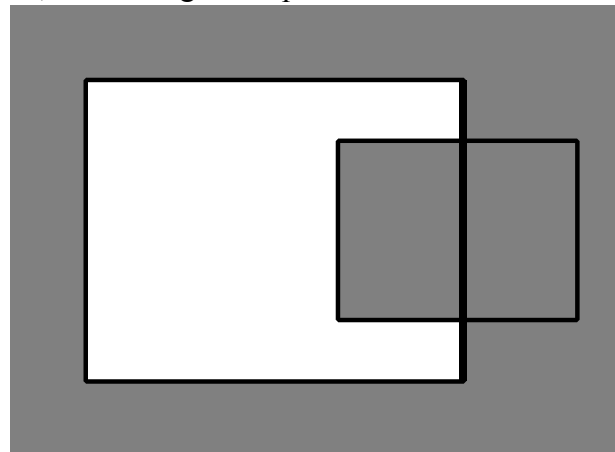
ra kerülnek a gép koordinátarendszerére úgy, hogy a bekapcsolt korrekciós eltolásokat is tartalmazzák. Tehát például, ha a G22 utasítás megadásakor be volt kapcsolva a hosszkorrekció Z irányban, az erre a tengelyre megadott koordináta határadatok úgy határolják be a mozgást, hogy a szerszám hegyét nem engedik a határon túl. Ha viszont nincs korrekció bekapcsolva a szerszám tartó 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 munkaterület belülről van tiltva és a tiltott zónába, vagy annak határára jutnak a tengelyek *1400 BELÜLRŐL TILTOTT ZÓNA* hibaüzenetet ad a vezérlő.
- Ha a munkaterület kívül van tiltva *130n TILTOTT ZÓNA t+*, vagy *132n TILTOTT ZÓNA t-* hibajelzést ad a vezérlő, ahol "t" a tiltott zónára futó tengely neve.

## 19.2 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 referenciapont felvétel után végzi.
- A paraméteres végállásfigyelés mindig külső teret tilt le.
- A paraméteres végállásfigyelés és a programozott munkatérbehatárolás által kijelölt tartományok átfedhetnek egymást.



19.2-1 ábra

### 19.3 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. Mozdulás 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.

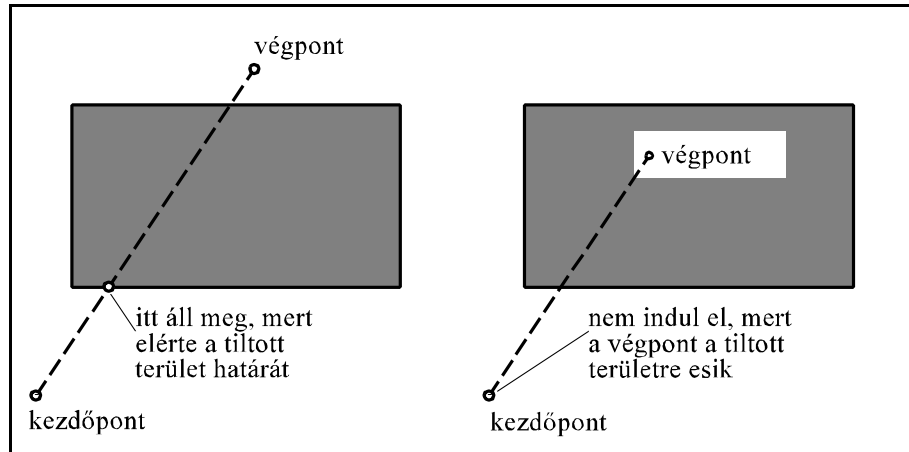
Ha a paramétermezőben a *CHBFMOVE* paraméter 1-be van

állítva a vezérlés mielőtt a mondatvégrehajtás során elindítaná a tengelyeket ellenőrzést végez, hogy az adott mondat programozott végpontja nem esik-e valamelyik tiltott területbe.

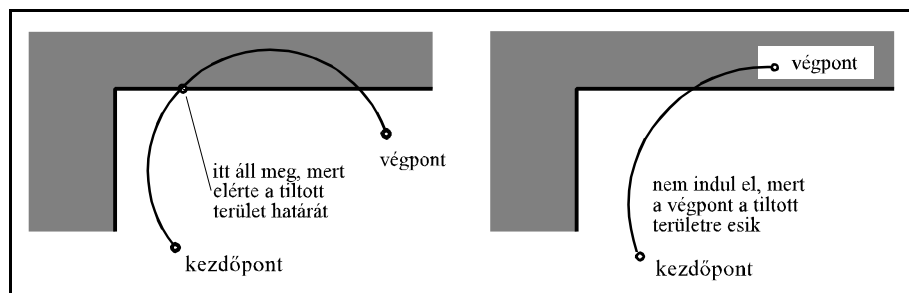
Ha az adott mondat végpontja a végállástartományon kívülre esik **3056 VÉGÁLLÁS** hibajelzést ad, ha a programozott tiltott munkatérbe esik **3057 TILTOTT TERÜLET** hibajelzést ad. Így

gyakorlatilag a mozgás el sem indul.

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.



19.3-1 ábra



19.3-2 ábra



## 20 A felhasználói makró

### 20.1 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

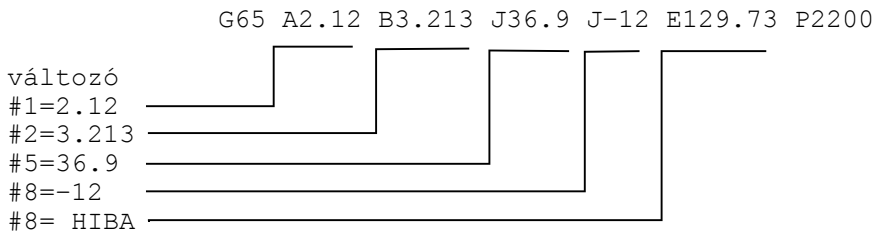
<i>lv</i>	<i>1.ak</i>	<i>2.ak</i>
#23	W	J7
#24	X	K7
#25	Y	I8
#26	Z	J8
#27	–	K8
#28	–	I9
#29	–	J9
#30	–	K9
#31	–	I10
#32	–	J10
#33	–	K10

– **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 elfo-

gadja 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. Tizedespont, előjel, is átadható a címeken.

## 20.2 Az öröklődő makróhívás

### 20.2.1 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 ismételten 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
Y30
X150
...
G67
    
```

a fúrást minden pozicionálás után végrehajtja

makróprogram

```

%01250
G0 Z#18                          (gyorsmeneti pozicionálás z irányban az R-1 címen
                                  megadott pontra)
G1 Z#26 F#9                       (az F130 címen megadott előtolással fúrás a Z-100
                                  címen megadott talppontig)
G4 P#24                            (várakozás a furat alján az X2 címen megadott ér-
                                  tékkel)
G0 Z-[#18+#26]                    (a szerszám visszahúzása a kiindulási pontra)
M99                                (visszatérés a hívó programba)
%
    
```

## 20.2.2 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 makrót hívja be, azt hajtja végre az L címen megadott számú ismétléssel.

A parancs hatása ugyanaz, mintha minden mondat G65-ös makróhívás lenne:

```
G66.1 P L
X Y Z _____
M S _____] = [_____ G65 P L X Y Z
X _____]   [_____ G65 P L M S
G67                [_____ G65 P L X
```

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 (legfőjebb a / feltételes mondat címe előzi meg) a második N címet argumentumnak veszi:

```
/N130 X12.3 Y32.6 N250
```

```
mondatszám _____
#24=12.3 _____
#25=32.6 _____
#14=250 _____
```

ha az N cím a mondat közepén található (bármely /-től különböző cím megelőzi) az N cím argumentumként kerül értelmezésre:

```
X34.236 N320
#24=34.236 _____
#14=320 _____
```

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 mondattól 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.

### 20.3 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>

utasítássor helyett az

Nn **Gg** <argumentum kijelölés>

utasítássort 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 kódoként 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.

### 20.4 Felhasználói makróhívás M kódra

A paramétermezőben ki lehet jelölni legfeljebb 10 különböző M kódot, amelyre makróhívást akarunk kezdeményezni. Ekkor az

Nn **Mm** <argumentum kijelölés>

utasítássort 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.

## 20.5 Felhasználói alprogramhívás M kódra

A paramétermezőben ki lehet jelölni legfeljebb 10 darab M kódot amire alprogramhívást lehet kezdeményezni. Ekkor az

Nn Gg Xx Yy M98 Pp

utasítás helyett a következő utasítás adható meg:

Nn Gg Xx Yy **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.

## 20.6 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 Yy **Tt**

mondat ekvivalens az alábbi két mondattal:

**#199=t**

Gg Xx Yy 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.

### 20.7 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 Yy Ss

mondat ekvivalens az alábbi két mondattal:

#198=s

Gg Xx Yy 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.

### 20.8 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, vagy az interpolátornak átadásra, hanem az A, B, vagy C kód az O9030, O9031, vagy az O9032 alprogram hívását kezdeményezi. Ekkor pl. a

Gg Xx Yy Bb

mondat ekvivalens az alábbi két mondattal:

#196=b

Gg Xx Yy M98 P9031

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 vagy az interpolátornak 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.

### 20.9 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.

### 20.9.1 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.

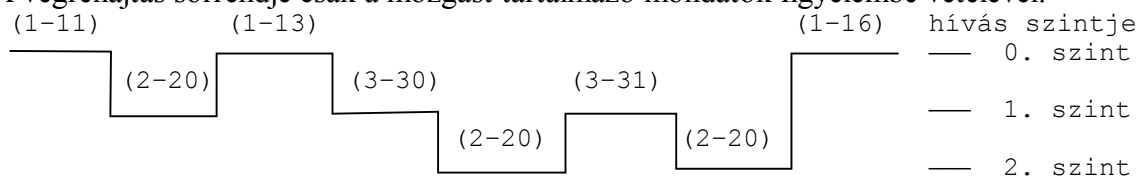
Öröklődő, G66 típusú makrók többszörös hívása esetén minden mozgásmondat végrehajtása után először a később megadott makró hívódik, és ebből hívódnak meg visszafelé haladva az előbb megadott makrók. Lássuk a következő példát:

```
%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 (O0003) törli, az N15 mondatban megadott az N10 mondatban hívottat (O0002).

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 szint-ről 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.

## 20.10 A felhasználói makró formátuma

A felhasználói makró programformátuma megegyezik az alprogramokéval:

```
O (programszám)
:
parancsok
:
M99
```

A programszám tetszőleges lehet, de az O9000–tól O9034–ig terjedő programszámok speciális hívásokra vannak fenntartva.

## 20.11 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 paramétereket átadni, addig a makrónak a lokális változókön keresztül argumentumok, vagy paraméterek átadhatók.



### 20.11.1 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 hivatkozunk.

#[#120-4] jelentése: a 120-as változóban található számból 4-et levonva kapjuk a hivatkozott változó számát.

### 20.11.2 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.

### 20.11.3 Üres változók

Az olyan változó értéke amelyekre még nem hivatkoztunk üres. A #0 változó értéke mindig üres:

#0=<üres>

### 20.11.4 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

## 20.12 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.

### 20.12.1 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 változó 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.

### 20.12.2 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 törlődnek kikapcsolásra.

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.

### 20.12.3 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 kiadáshoz 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–#13999**

A szerszámkorrekciós értékeket ki lehet olvasni a #10001–#13999 változókon, illetve értéket lehet nekik adni.

Korrekció száma	<b>H</b>		<b>D</b>	
	<i>geometriai</i>	<i>kopás</i>	<i>geometriai</i>	<i>kopás</i>
<b>1</b>	#10001	#11001	#12001	#13001
<b>2</b>	#10002	#11002	#12002	#13002
<b>:</b>	<b>:</b>	<b>:</b>	<b>:</b>	<b>:</b>
<b>999</b>	#10999	#11999	#12999	#13999

**Munkadarab nullponteltolások: #5201–#5328**

A munkadarab nullpont eltolások értékeit ki lehet olvasni a #5201–#5328 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 *AXIS* paramétercsoportban. Általában az 1. tengely az X, a 2. tengely az Y, a 3. tengely a Z 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 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	0
1	0	0	0	1
2	0	1	1	0
3	0	1	1	1
4	1	0	0	0
5	1	0	0	1
6	1	1	1	0
7	1	1	1	1

0: a funkció hatásos

1: a funkció el van nyomva





**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

**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)	
:	:	nem lehetséges
#5028	8. tengely pillanatnyi pozíciója (G53)	

**A pillanatnyi pozíció (G53)**

- a gép koordinátarendszerében,
- az összes korrekció (hossz, sugár, szerszámeltolás) figyelembe vételével kerül a változóba.

### Pillanatnyi pozíciók a munkadarab koordinátarendszerében

rendszer változó	pozíció információ jellege	beolvasás mozgás közben
#5041	1. tengely pillanatnyi pozíciója	nem lehetséges
#5042	2. tengely pillanatnyi pozíciója	
:		
#5046	8. tengely pillanatnyi pozíciója	

#### A pillanatnyi pozíció

- az aktuális munkadarab koordinátarendszerben,
- a koordinátaeltolások figyelembe vételével,
- derékszögű koordinátákban,
- az összes korrekció (hossz, sugár, szerszámeltolás) figyelembe vételével kerül a változóba.

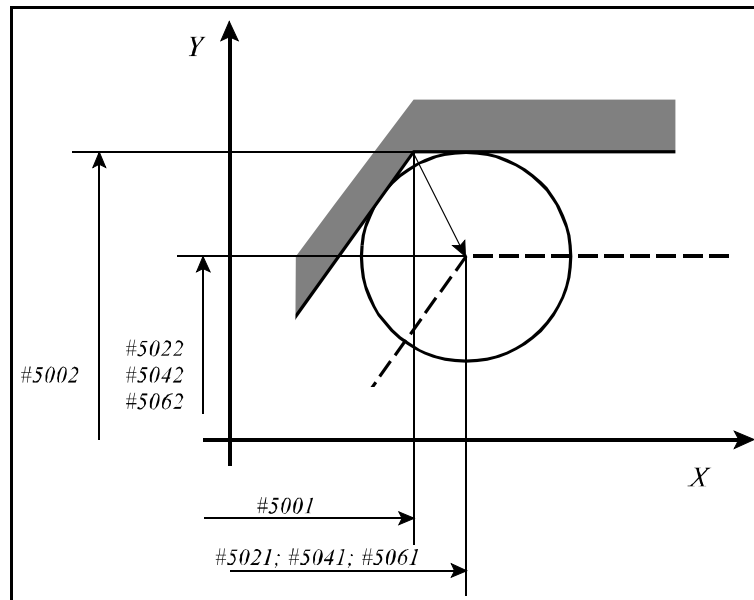
### Tapintási pozíciók

rendszer változó	pozíció információ jellege	beolvasás mozgás közben
#5061	1. tengely tapintási pozíciója (G31)	lehetséges
#5062	2. tengely tapintási pozíciója (G31)	
:		
#5068	8. tengely tapintási pozíciója (G31)	

#### 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 vesznek fel.

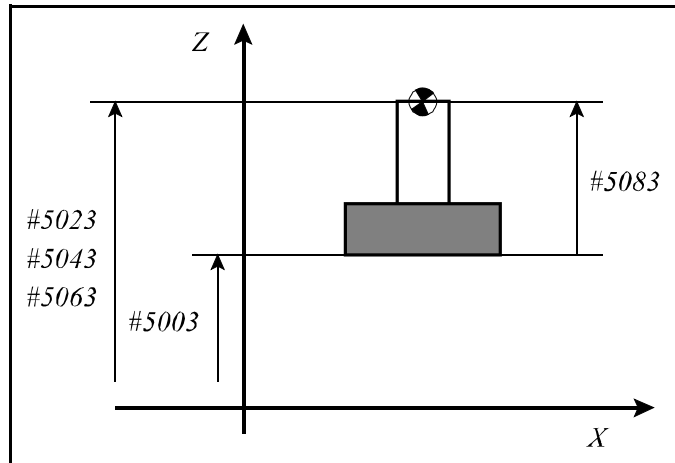


20.12.3-1 ábra

**Szerszám–hosszkorrekciók**

rendszer változó	pozíció információ jellege	beolvasás mozgás közben
#5081	1. tengelyen bejegyzett hosszkorrekció	
#5082	2. tengelyen bejegyzett hosszkorrekció	
:		
#5088	8. tengelyen bejegyzett hosszkorrekció	nem lehetséges

A kiolvasható szerszám–hosszkorrekció a végrehajtás alatt álló mondatban érvényes hosszkorrekció.



20.12.3-2 ábra

**Szervo lemaradás**

rendszer változó	pozíció információ jellege	beolvasás mozgás közben
#5101	1. tengelyen a szervo lemaradás értéke	
#5102	2. tengelyen a szervo lemaradás értéke	
:		
#5108	8. tengelyen a szervo lemaradás értéke	nem lehetséges

A kiolvasható lemaradás előjelhelyes érték, milliméterben megadva.

**20.13 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.

**20.13.1 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.

### 20.13.2 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.**Szinusz:**  $\#i = \text{SIN } \#j$ A függvény kódja: **SIN**A művelet hatására  $\#i$  változó  $\#j$  változó szinuszát veszi fel.  $\#j$  értéke mindig fokban értendő.**Koszinusz:**  $\#i = \text{COS } \#j$ A függvény kódja: **COS**A művelet hatására  $\#i$  változó  $\#j$  változó koszinuszát 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  $(2n+1) \cdot 90^\circ$ , ahol  $n=0, \pm 1, \pm 2, \dots$ **Arkuszszinusz:**  $\#i = \text{ASIN } \#j$ 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  $-90^\circ$  közé esik.**Arkuszkoszinusz:**  $\#i = \text{ACOS } \#j$ 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  $0^\circ$  és  $180^\circ$  közé esik.**Arkusztangens:**  $\#i = \text{ATAN } \#j$ A függvény kódja: **ATAN**A művelet hatására  $\#i$  változó  $\#j$  változó arkusztangensét veszi fel. Az eredmény, azaz  $\#i$  értéke  $+90^\circ$  és  $-90^\circ$  közé esik.**Exponenciális:**  $\#i = \text{EXP } \#j$ A függvény kódja: **EXP**A művelet hatására a  $\#i$  változó a természetes szám (e)  $\#j$ -edik hatványát veszi fel.

**Természetes alapú logaritmus:** #i = LN #jA 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 #jA függvény kódja: **ABS**

A művelet hatására a #i változó a #j változó abszolút értékét veszi fel.

**Bináris szám binárisan kódolt decimálissá átalakítása:** #i = BCD #jA függvény kódja: **BCD**A művelet hatására a #i változó a #j változó BCD értékét veszi fel. #j változó értékter-  
tartománya: 0 – 99999999.**Binárisan kódolt decimális szám binárisra alakítása:** #i = BIN #jA függvény kódja: **BIN**A művelet hatására a #i változó a #j változó bináris értékét veszi fel. #j változó érték-  
tartománya: 0 – 99999999. A #j változóban az egyes tetrádokon lévő számjegyek  
nagysága nem lehet 9-nél nagyobb.**Kerekítés abszolút értékben lefelé:** #i = FIX #jA függvény kódja: **FIX**

A művelet a #j változó törtrészét eldobja és ez az érték kerül #i változóba.

Például:

$$\begin{aligned} \#130 &= \text{FIX } 4.8 = 4 \\ \#131 &= \text{FIX } -6.7 = -6 \end{aligned}$$
**Kerekítés abszolút értékben felfelé:** #i = FUP #jA függvény kódja: **FUP**

A művelet a #j változó törtrészét eldobja és abszolút értékben 1-et ad hozzá.

Például:

$$\begin{aligned} \#130 &= \text{FUP } 12.1 = 13 \\ \#131 &= \text{FUP } -7.3 = -8 \end{aligned}$$
**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:

$$\#110 = \#111 + \#112 * \text{COS } \#113$$

\_\_\_\_\_ 1 \_\_\_\_\_  
\_\_\_\_\_ 2 \_\_\_\_\_ a műveletek sorrendje  
\_\_\_\_\_ 3 \_\_\_\_\_

### A műveletek végrehajtási sorrendjének módosítása

[ és ] zárójelek használatával a műveletek végrehajtási sorrendje módosítható. A zárójelezés öt-szörös mélységig megengedett. Ha ennél nagyobb mélység van a programban a vezérlés *3064 HIBÁS MAKRÓ KIFEJEZÉS* hibajelzést ad.

Példa háromszoros mélységű zárójelezésre:

$$\#120 = \text{COS} \left[ \left[ \left[ \#121 - \#122 \right] * \#123 + \#125 \right] * \#126 \right]$$

$\begin{array}{c} \text{-----} \quad 1 \quad \text{-----} \\ \text{-----} \quad 2 \quad \text{-----} \\ \text{-----} \quad 3 \quad \text{-----} \\ \text{-----} \quad 4 \quad \text{-----} \\ \text{-----} \quad 5 \quad \text{-----} \end{array}$

A számok a műveletek végrehajtási sorrendjét mutatják. Látható, hogy az azonos szintű zárójeleken belül a műveletek végrehajtási sorrendjére a fent említett precedenciaszabály érvényes.

### 20.13.3 Feltételes kifejezések

A programnyelv a következő feltételes kifejezéseket ismeri:

egyenlő:	#i <b>EQ</b> #j
nem egyenlő:	#i <b>NE</b> #j
nagyobb, mint:	#i <b>GT</b> #j
kisebb, mint:	#i <b>LT</b> #j
nagyobb, vagy egyenlő:	#i <b>GE</b> #j
kisebb, vagy egyenlő:	#i <b>LE</b> #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!

### 20.13.4 Feltétel nélküli elágazás: GOTO<sub>n</sub> (MENJ)

A **GOTO<sub>n</sub>** 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.

### 20.13.5 Feltételes elágazás: IF[<feltételes kifejezés>] GOTO<sub>n</sub> (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.

### 20.13.6 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.

### 20.13.7 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 ismételten 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
:

```

– DOm ... 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 ágazni:

```

:
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
:

```

### 20.13.8 Adatkiadási parancsok

A vezérlés a következő adatkiadási parancsokat ismeri:

<b>POPEN</b>	periféria megnyitása
<b>BPRNT</b>	bináris adatkiadás
<b>DPRNT</b>	decimális adatkiadás
<b>PCLOS</b>	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**

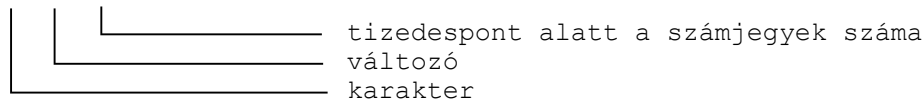
Mielőtt adatkiadó parancsot adunk, meg kell nyitni a megfelelő perifériát, amin keresztül az adatkiadás történni fog. A megfelelő periféria kiválasztása az n számmal történik:

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:

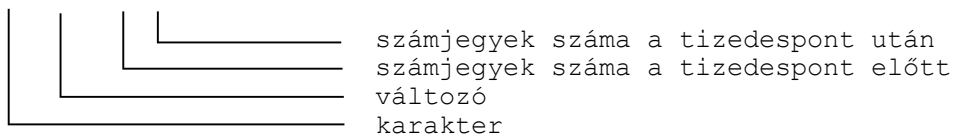
```
BPRNT [ C*/ X#110 [3] Y#120 [3] M#112 [0] ]
#110=318.49362        318494=0004DC1Eh
#120=0.723415        723=000002D3h
#112=23.9            24=00000018h
```

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 helyiértéküktől 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.

### 20.14 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.

### 20.15 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**

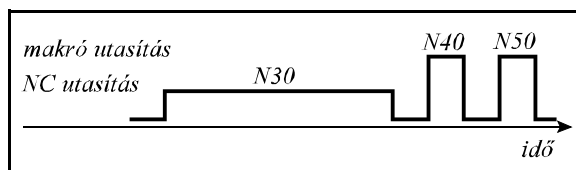
%O1000

```

N10 #100=50
N20 #101=100
N30 G1 X#100 Y#101
N40 #100=60 (értékadás N30 után)
N50 #101=120 (értékadás N30 után)
N60 G1 X#100 Y#101

```

Az N40 és N50 mondatokban leírt értékadást az N30 mondat végrehajtása után végzi el.



20.15-1 ábra

**SBSTM=1**

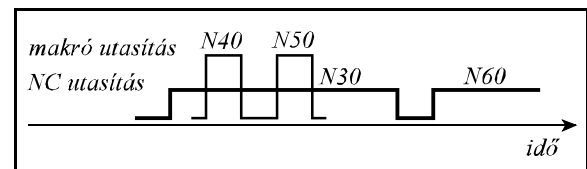
%O1000

```

N10 #100=50
N20 #101=100
N30 G1 X#100 Y#101
N40 #100=60 (értékadás N30 közben)
N50 #101=120 (értékadás N30 közben)
N60 G1 X#100 Y#101

```

Az N40 és N50 mondatokban leírt értékadást az N30 mondatban folyó mozgás közben végzi el.



20.15-2 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.

### 20.16 Makrók és alprogramok kijelzése automata üzemmódban.

A makrók és alprogramok mondatait a vezérlés kijelzi 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.

### 20.17 STOP gomb használata makróutasítás végrehajtása közben.

A STOP gomb megnyomása, vagyis a programvégrehajtás felfüggesztése mindig a végrehajtása alatt álló makróutasítás befejezése után lesz hatásos.

### 20.18 Üregmaró makróciklus

A

#### G65 P9999 X Y Z I J K R F D E Q M S T

utasítás üregmaró ciklust indít. A ciklus végrehajtásához be kell tölteni a tárba az O9999 programszámú makró, amely a vezérlés PROM tárában található.

A ciklus hívása előtt az üreg geometriai közepe fölé kell állni a kiválasztott síkban, az anyag felületétől biztonsági távolságnyra. A ciklus végén ugyanerre a pontra húzza vissza a szerszámot.

A mondat címeinek értelmezése:

**X:** az üreg X irányú mérete

**Y:** az üreg Y irányú mérete

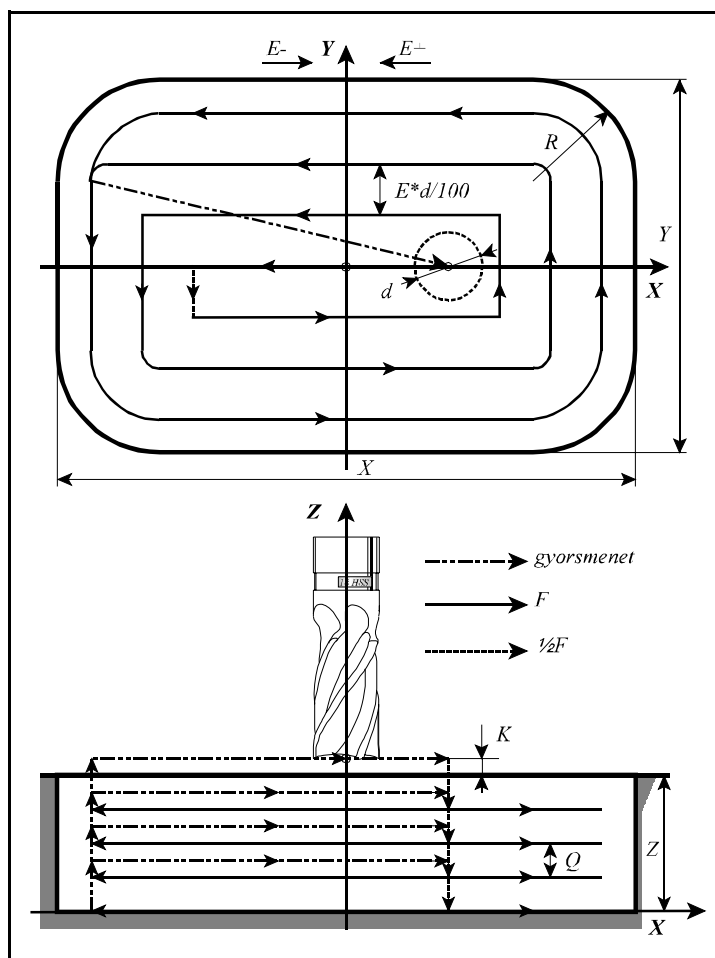
**Z:** az üreg Z irányú mérete

A G17, G18, G19 utasítás dönti el, hogy a három koordináta közül melyik az üreg hosszúsága, szélessége illetve mélysége. Például: G17 esetén Z az üreg mélysége, X és Y közül a hosszabb koordináta az üreg hosszúsága, a rövidebb a szélessége. Ezeket az értékeket abszolút értékben pozitív számként kell beadni.

**R:** az üreg sarkainak sugara  
R címen kell megadni az üreg sarkainak esetleges lekerekítését. Ha az R címet nem töltjük ki az üreg sarkainak lekerekítése a szerszám sugarával lesz egyenlő.

**I:** biztonsági távolság mélységi irányban G19 esetén

**J:** biztonsági távolság mélységi irányban G18 esetén



20.18-1 ábra



**K:** biztonsági távolság mélységi irányban G17 esetén

A kiválasztott sík függvényében I (G19), J (G18), vagy K (G17) címen kell megadni a szerszámirányú biztonsági ráhagyást a mondatban. A vezérlés a ciklus indulásakor feltételezi, hogy a szerszám hegye a darab felületétől ilyen távolságra áll. Az üreg kimarása közben pedig, amikor egy szint kiszedésével végzett, ekkora távolságra emeli ki a szerszámot, hogy a következő szint kiszedéséhez a kezdőpontra álljon.

**D:** a szerszám sugárkorrekcióját tartalmazó rekesz címe

A programban használt szerszám sugárkorrekciós regiszterének számát D címen kötelezően meg kell adni. Az üregmarást egyébként G40 állapotban kell végezni.

**E:** fogásszélesség a maróátmérő százalékában

+ előjellel: megmunkálás az óramutatóval ellentétes

– előjellel: megmunkálás az óramutatóval egyező irányban

E címen két információt közölhetünk a vezérlővel. E értéke azt adja meg, hogy a fogásszélesség mekkora legyen a maróátmérő százalékában. Ha nincs megadva, a vezérlés automatikusan +83%-ot tételez föl. A vezérlés az üreg szélességének függvényében az E címen megadott adatot módosíthatja úgy, hogy egy szint kiszedésénél a fogásvétel értéke egyenletes legyen. A módosítás azonban csak csökkentés lehet. Az E cím előjele a marás irányát mondja meg. Ha E+, azaz pozitív, a megmunkálás az óramutató járásával ellentétes, ha E-, azaz negatív, a megmunkálás az óramutató járásával megegyező irányban történik.

**Q:** fogásmélység

Q címen adhatjuk meg a fogásmélységet az alkalmazott mértékrendszerben azaz, mm-ben, vagy inch-ben. A vezérlés az üreg mélységének függvényében a programozott értéket az egyenletes fogásfelosztás érdekében felülbírállhatja. A módosítás azonban csak csökkentés lehet.

**F:** előtolás

F címen adhatjuk meg a ciklus során alkalmazott előtolás nagyságát. Ha F címnek nem adunk értéket az öröklött F értéket veszi figyelembe. Az F érték 50%-t alkalmazza az alábbi esetekben:

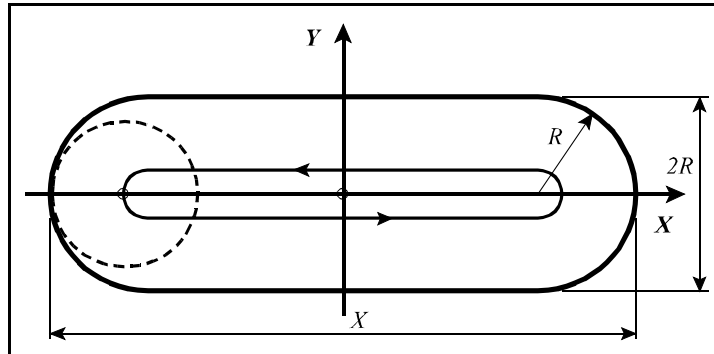
- Amikor egy szintet elkezd kibontani és a szerszám irányában Q mélységet lefűr,
- Az üreg hosszirányban történő marásánál mindaddig, amíg a szerszám mindkét oldalon terhelve van.

**M S T:** funkció

Az üregmarást hívó mondatban egy db. M, ill. S, T funkciót lehet megadni, amit a vezérlő a marás megkezdése előtt hajt végre.

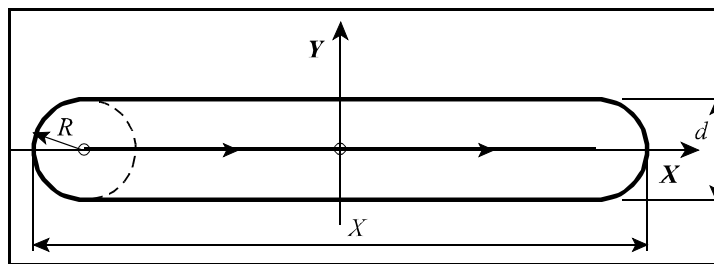
*Az üregmarás elfajuló esetei:*

Ha az üreg szélessége nem lett megadva az üreg sarkainak sugarát kétszer veszi és ez lesz az üreg szélessége.



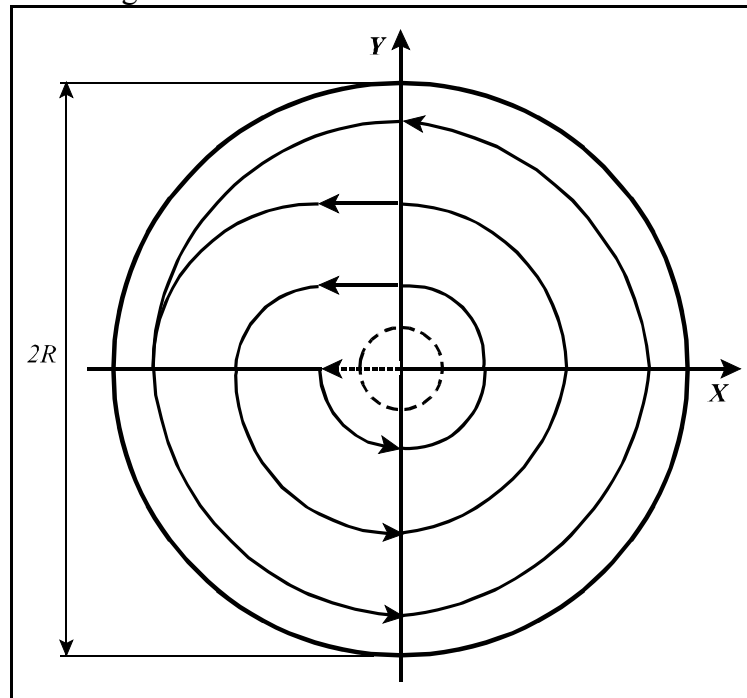
20.18-2 ábra

Ha sem az üreg szélessége, sem a sarok lekerekítési sugara nem lett megadva az alkalmazott szerszám átmérőjét veszi az üreg szélességének (horony).



20.18-3 ábra

Ha sem az üreg hosszúsága, sem a szélessége nem lett megadva, csak R címet programoztunk, akkor egy R sugarú körüregot szed ki.



20.18-4 ábra

Ha sem hosszúság, sem szélesség, sem sugár nem lett megadva akkor a ciklus fúrássá fajul.

*Az ürgmarás végrehajtása során előforduló hibajelzések:*

**MACRO ERROR 1:** mondatkitöltési hiba. Lehetséges okai:

- Az üregmélység nincs megadva,
- A szerszámsugár nics megadva,
- A fogásmélység nincs megadva.

**MACRO ERROR 2:** méretmegadási hiba. Lehetséges okai:

- Ha az üreg hosszúságának, vagy szélességének megadott méret kisebb, mint az üreg sugarának kétszerese,
- Ha az üreg hosszúsági, vagy szélességi mérete kisebb, mint a D címen lehívott szerszámátmérő,
- Ha a fogásszélességre megadott érték 0, vagy a lehívott szerszámátmérő 0,
- Ha a fogásmélység értéke 0, azaz Q címre 0 lett programozva.

## Jegyzetek

**Betűrendes index:**

#0	165	Főorsó	14, 15, 44, 62, 70, 71
#10001–#13999	169	jeladó	29
#1000–#1015	167	orientálás	64
#1032	167, 168	override	30, 48
#1100–#1115	168	tartományváltás	70
#1132	168	Főprogram	10
#195	162	Forgatás	38, 111
#196	162	Forgó tengely	23
#197	162	Formátum	10
#198	162	Fősík	60
#199	161	Főtengely	61, 155
#1nn	167	Furat pozíciója	130
#3000	170	Fúróciklus címei	129
#3001	171	Fúróciklusok	128
#3002	171	Fúróciklusok kódjai	129
#3003	171	Fúróciklusok konfigurálása	129
#3004	171	Fúrótengely	128
#3006	172	Gyorsítás	46
#3007	172	Háromdimenziós szerszámkorrekció	109
#3901	172		170
#3902	172	Hibajelzés	170
#4001–#4130	173	Hosszkorrekció	15, 57, 76, 77, 81, 150, 151, 153, 155, 175
#4201–#4330	173	Hűtővíz	15, 70, 71
#5201–#5326	170	Ikremensrendszer	17
#5nn	167	Inch	39
Abszolút koordinátamegadás	13	Inkremensrendszer	39, 40, 45, 76
Alkatrészprogram	9	bemeneti	18
Alprogram	10, 70, 72, 73, 100	kimeneti	18
Belső körív	49	Inkrementális koordinátamegadás	14
Belső sarok	48, 86, 87, 90, 93, 94, 107	Interface	167, 168
Bemeneti inkremensrendszer	18	Interferenciavizsgálat	105
Bemeneti mértékrendszer	17	Interpoláció	12
Biztonsági funkciók	154	Kezdőpont	130
Címlánc	9	Kicsinyítés	112
decimal point	185	Kiesztergálás	130, 136, 146, 147
DNC üzemmód	11	Kiindulási pont	128
Dominátor konstans	110	Kimeneti inkremensrendszer	18
Elágazás	179	Kimeneti mértékrendszer	17
Előkészítő funkciók	12	Kontúrkövetés	100
Előtolás	12, 171	iránya	96
Előtoláscsökkentés	48, 49	zavarproblémái	105
Értékhatár	9, 39, 45, 50, 73, 74, 76	Koordinátaadatok	39
Feltételes kifejezések	179	Koordinátamegadás	13
Feltételes stop	70	abszolút	13
Feltételvizsgálat	166, 179		

növekményes .....	<u>14</u>	AXIS .....	<u>170</u>
Koordinátarendszer .....	<u>13, 54</u>	B(9031) .....	<u>162</u>
eltolása .....	<u>57</u>	C(9032) .....	<u>162</u>
gépi .....	<u>54</u>	CDIR6 .....	<u>65</u>
létrehozása .....	<u>57</u>	CHBFMOVE .....	<u>156</u>
lokális .....	<u>58</u>	CIRCOVER .....	<u>49</u>
munkadarab .....	<u>55, 170</u>	CLEG83 .....	<u>131, 139</u>
transzformációi .....	<u>111</u>	CODES .....	<u>22, 23, 37, 44, 61, 79</u>
Kopáskorrekció .....	<u>16</u>	CORNANGL .....	<u>48</u>
Kör .....	<u>48, 49, 86, 92, 93, 99</u>	CORNOVER .....	<u>49</u>
adatmegadása .....	<u>25, 38, 82</u>	CUTTING2 .....	<u>171</u>
interpolálása .....	<u>12, 24, 43, 46, 60</u>	DECDIST .....	<u>49</u>
iránya .....	<u>113</u>	DELTV .....	<u>105</u>
Körasztal .....	<u>58</u>	DOMCONST .....	<u>110</u>
Korrekció .....	<u>52, 60, 76, 83</u>	EXTER .....	<u>154</u>
módosítása .....	<u>77, 169</u>	FEED .....	<u>43</u>
Követő nullák .....	<u>39</u>	G(901n) .....	<u>160</u>
Közbülső pont .....	<u>51, 52, 102</u>	G31FD .....	<u>151</u>
Lassítás .....	<u>46</u>	G37FD .....	<u>152</u>
Lemaradás .....	<u>175</u>	HELICALF .....	<u>28</u>
Léptékezés .....	<u>38, 112</u>	INDEX_C1 .....	<u>65</u>
M kódok .....	<u>70</u>	INDEX1 .....	<u>64, 141</u>
Megközelítési pont .....	<u>128</u>	INPOS .....	<u>22</u>
Mérés .....	<u>151</u>	INTERFER .....	<u>105</u>
Mérőfunkciók .....	<u>151</u>	LIMP2n .....	<u>154</u>
Mértékrendszer .....	<u>39, 40, 45, 76</u>	M(900n) .....	<u>161</u>
Metrikus .....	<u>39</u>	M(902n) .....	<u>161</u>
Mondat .....	<u>10</u>	M_NUMB1 .....	<u>65</u>
Mondatonkénti végrehajtás .....	<u>171</u>	MD8 .....	<u>187</u>
Mondatszám .....	<u>173</u>	MD9 .....	<u>187</u>
Munkadarab koordinátarendszerek .....	<u>55</u>	MULBUF .....	<u>21</u>
Munkatér behatárolás .....	<u>154</u>	O_LINE .....	<u>168</u>
Nagyítás .....	<u>38, 112</u>	ORIENT1 .....	<u>64, 136, 146, 147</u>
Normálószám .....	<u>110</u>	POSCHECK .....	<u>22</u>
Növekményes koordinátamegadás .....	<u>14</u>	PRNT .....	<u>184</u>
Öröklődő funkciók .....	<u>14, 173</u>	PRTCNTM .....	<u>70, 172</u>
Override .....	<u>30, 43, 47, 48, 135, 140, 150,</u>	PRTREQRD .....	<u>172</u>
tiltás .....	<u>171</u>	PRTTOTAL .....	<u>172</u>
tiltás .....	<u>20</u>	RAD .....	<u>77</u>
Paraméter		RADDIF .....	<u>26</u>
A(9030) .....	<u>162</u>	RAPDIST .....	<u>152, 153</u>
ACC0 .....	<u>46</u>	RAPID6 .....	<u>65</u>
ACCDIST .....	<u>49</u>	REFPOS .....	<u>52</u>
ADD .....	<u>152</u>	RETG73 .....	<u>131, 134</u>
ALADIST .....	<u>152, 153</u>	S(9033) .....	<u>162</u>
ANG.ACCU .....	<u>111</u>	SECOND .....	<u>50</u>
ANGLAL .....	<u>106, 108</u>	SERIAL .....	<u>186</u>

SKIPF	<u>151</u>	Szó	<u>9</u>
STRKEN	<u>154</u>	Tárolt pozíció	<u>151</u>
T(9034)	<u>161</u>	Teljes kör	<u>26, 104</u>
TAPDWELL	<u>135, 140</u>	Tengelyek	
WRPROT	<u>167</u>	elnevezése	<u>17</u>
Pontos megállás	<u>47, 171</u>	inkrementsrendszere	<u>17</u>
Pozícióinformációk	<u>173</u>	pozíciója	<u>151</u>
Pozíciókijelzés	<u>57-59</u>	száma	<u>17</u>
Pozicionálási sík	<u>128</u>	Tiltott tartomány	<u>156</u>
Pozícióvisszacsatolás	<u>64</u>	Tizedespont	<u>39, 158, 183, 184</u>
Program vége	<u>70</u>	kiadása	<u>184</u>
Programformátum	<u>10, 164</u>	Transzformációk	<u>100, 111</u>
Programkezdet	<u>10</u>	programozási szabályai	<u>114</u>
Programnév	<u>10</u>	Tükrözés	<u>38, 99, 113, 115, 130, 131, 172</u>
Programozott stop	<u>70</u>	Ugrás	<u>75</u>
Programszám	<u>10, 173</u>	Üregmaró ciklus	<u>188</u>
Programvég	<u>10</u>	Változó	<u>164</u>
PROM	<u>188</u>	0	<u>165</u>
R pont	<u>128</u>	üres	<u>165</u>
Referenciapont	<u>12, 51, 52</u>	Változó sugarú körív	<u>26, 28, 103, 104</u>
Rendszerváltozó	<u>151, 167</u>	Változók	<u>166</u>
RS232	<u>11</u>	globális	<u>161, 162, 164, 167</u>
Sarkok kerülése	<u>91, 92, 99, 105</u>	lokális	<u>157, 162, 164, 166</u>
Sarkok megmunkálása	<u>47, 48</u>	Várakozás	<u>30, 50, 90, 100, 131</u>
Sarokív	<u>99</u>	Végállás	<u>155, 156</u>
Segédfunkció	<u>71</u>	paraméteres	<u>51, 155</u>
Sík	<u>60</u>	Végrehajtási sorrend	<u>71</u>
váltás	<u>61, 83</u>	Végtelen ciklus	<u>75</u>
Síkválasztás	<u>54, 60</u>	Vektor megtartás	<u>98</u>
STOP	<u>150, 171, 188</u>	Vezérelt tengelyek	<u>17</u>
feltételes	<u>70</u>	Vezető nullák	<u>39</u>
kapcsolók	<u>47</u>	Visszahúzás	<u>128, 131, 139</u>
programozott	<u>70</u>		
tiltás	<u>43, 48, 135, 140</u>		
STOP állapot	<u>146, 148</u>		
Sugárkorrekció	<u>16</u>		
Számábrázolás	<u>165</u>		
Szerszámhossz mérés	<u>152</u>		
Szerszámkezelés	<u>68</u>		
Szerszámsugár korrekció	<u>28, 76</u>		
irányváltás	<u>96</u>		
módosítása	<u>77</u>		
síkbeli	<u>53, 60, 83</u>		
térbeli	<u>52, 109</u>		
vektor megtartás	<u>98</u>		
Szerszámszám	<u>14, 68</u>		
Szerszámváltás	<u>71</u>		

