

Alprogramok használata Mach3-ban

A Mach3 G-kód értelmezője két formában támogatja alprogramok hívását:

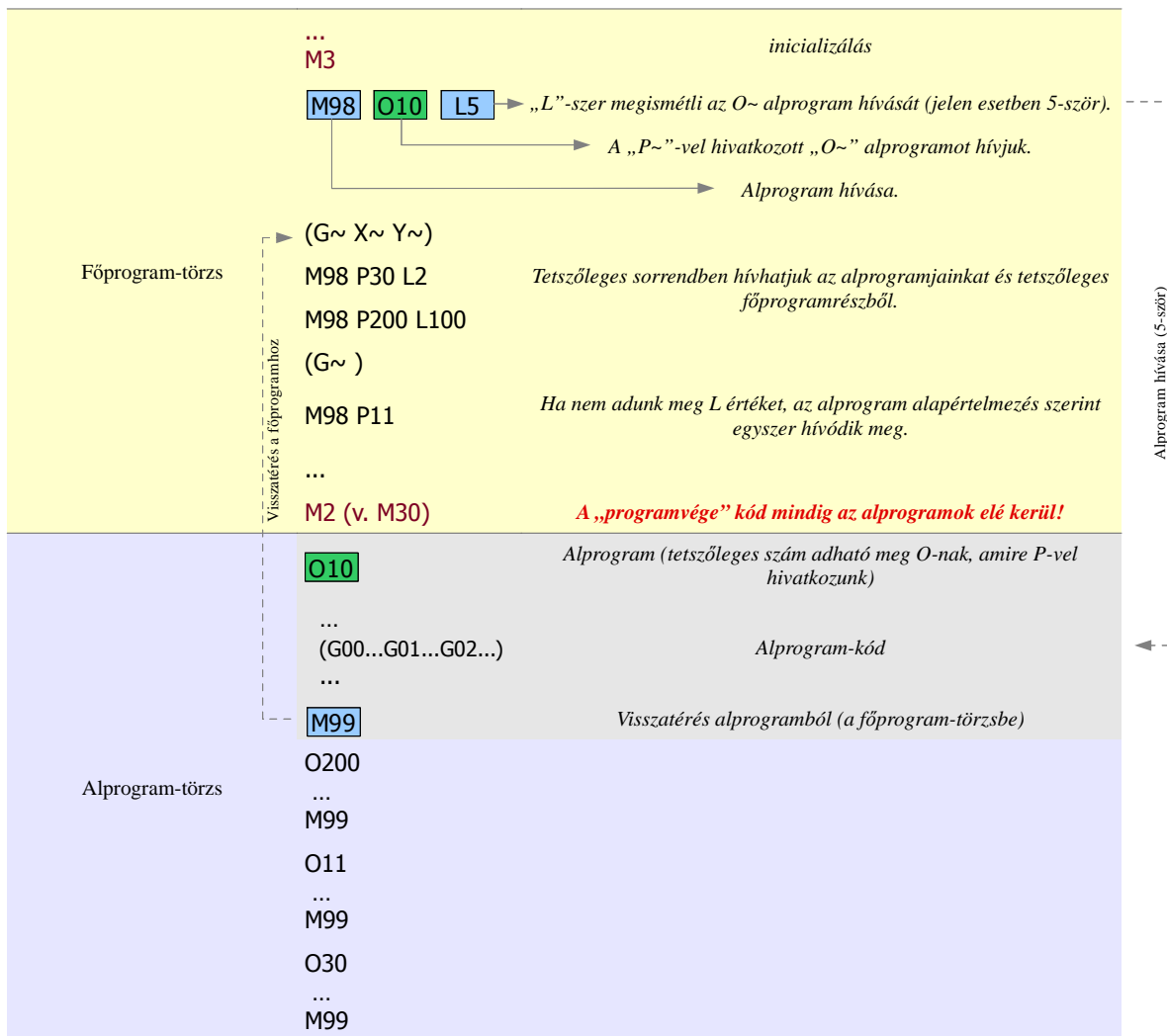
- Ha az alprogram az érvényes alkatrészprogramban helyezkedik el, akkor az

M98 P~ L~ kód segítségével;

- Ha az alprogram különálló fájlban található, akkor az

M98(fájlnév) L~ kód segítségével.

Jelen esetben csak azzal a formával foglalkozom, amikor az alprogramunk az éppen érvényes programban helyezkedik el. Alprogramot tartalmazó g-kódok általános szerkezete:



A fenti vázlat szemlélteti, hogy a G-kód két programtörzshez tartozhat.

A főprogram-törzs:

Tulajdonképpen az inicializációs és a programvége M kódok között helyezkedik el, tetszőleges G-kódokat és alprogram hívásokat tartalmazhat. Ez a programrész minden esetben végrehajtódik a kód futtatása során. M98 alprogram hívást tartalmazó G-kód esetén **az inicializációs rész (M3 és társai) elhagyhatóak, de a programvége (M2, M30) kód nem!** Utóbbi hiányában a Mach3 a kódot hibásnak értelmezi!

Az alprogramok törzse:

Itt helyezkednek el az O~ és M99 kódok által közrefogott alprogramkódok. Az O-nak tetszőleges számot írhatunk be, amely számra az M98 kód P paraméterével hivatkozunk majd.

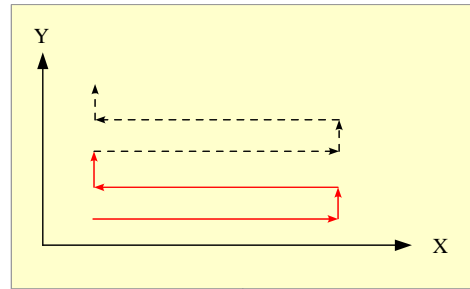
Az alprogramjainkat (ahogy az a fenti vázlatból is látható) tetszőleges sorrendben írhatjuk a kódunkba, és tetszőleges pontból is hívhatjuk meg azokat.

Mindig csak azok az alprogramok lesznek végrehajtva, amelyeket a főprogram törzséből M98 kódokkal hívunk. Itt jegyzem meg, hogy alprogram(ok) akkor is szerepelhet(nek) a G-kódban, ha soha nem hívjuk meg azokat. Fontos még azt is megemlíteni, hogy alprogramot egy másik alprogramból is meghívhatunk.

Egy egyszerű példa

Az ábra szerinti „kígyóvonalban” szeretnénk a marónkkal simítást végezni Y=5 mm lépésenként, X=120 mm hosszokban egymás után 20-szor megismételve a lépéseket. (A ciklust az X=10, Y=5 mm pontból kezdjük.) Ehhez egyesével kellene beprogramozni minden egyes X-Y lépést, azonban a program néhány sorra rövidíthető, ha csak a piros színnel kiemelt szerszámpályát programozzuk be egyszer, majd ezt egy ciklusba szervezzük. A programunk, mint látható még nem tökéletes, mert az utolsó Y lépés szükségtelen lehet a megmunkálás során.

G00 Z5	(Felemeljük a Z-t a biztonság kedvéért :)
G00 X10 Y5	(A kezdőpontra állunk)
M3	(Elindítjuk a főorsót – elhagyható kód)
G01 Z-0.1	(-0.1 mm fogást veszünk)
M98 P10 L20	(Meghívjuk a „kígyóvonal” alprogramot 20-szor)
G00 Z5	(Ezzel folytatódik a programunk az alprogramból való visszatérés után)
M5	(Főorsó ki)
G00 X0 Z0	(nullpontra pozicionálunk)
M2	(Programvége)
O10	(a „kígyóvonal” alprogram)
G91	(Átváltunk relatív távolság-módba!)
G01 X120	(X+120 mm-re pozicionálunk)
G01 Y5	(Y+5 mm-re...)
G01 X-120	(És nem X10, mert még relatív módban vagyunk!)
G01 Y5	(Ismét Y+5 mm-t pozicionálunk)
G90	(Visszaállunk abszolút távolság-módba)
M99	(Alprogram vége – visszatérés a főprogramhoz)
%*	



*: A %-jel a Mach3 értelmezőjében a kommentezést jelöli. Elfogadott szokás ezzel vagy üres sorral a programot zárni a programot, mert a Mach3 a szövegfájlok utolsó sorát nem értelmezi!

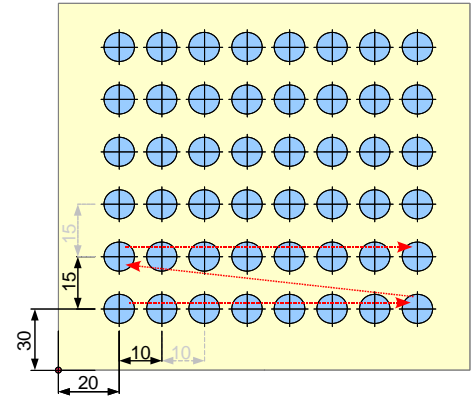
Egy összetettebb példa

6 sorban, soronként 8db 3mm mély furatot szeretnénk fúrni, X=10 mm furattávolságban, Y=15 mm sortávolságonként. (X=20, Y=30 mm kezdőpozícióból.)

Ehhez külön írunk egy eljárást, ami X irányban léptetve végzi el a fúrásokat, majd egy másik eljárást, amely a sorok végeztével Y irányba léptet, majd megismétli a fúrás-sorozatot (alprogramból hívunk meg egy másik alprogramot). Az előbbi példához hasonlóan most is kígyóvonalban haladunk (haladhatnánk más sorrendben is, de az egyszerűség kedvéért most ezt a megoldást választjuk). Az előbbieket képest ott láthatunk eltérést, hogy nem csak végighaladunk X irányban, hanem ciklikusan el is végzünk közben más egyéb műveleteket is. Ehhez egy másik alprogramot hívunk meg a sorok léptetését végző alprogramból.

Természetesen a megmunkálási sorrend is lehetne más irány, Y irányban haladva is végezhetnénk a fúrásokat, most azonban X sorrendben tesszük.

G00 Z5	(Felemeljük a Z-t)
G00 X20 Y30	(A kezdőpontra állunk)
M3	(Elindítjuk a főorsót – elhagyható kód)
M98 P10 L6	(A sorozatfúró eljárás)
G00 Z5	(Ezzel folytatódik a programunk az alprogramból való visszatérés után)
M5	(Főorsó ki)
G00 X0 Z0	(nullpontra pozicionálunk)
M2	(Programvége)
O10	(A sorok ismétlését végző alprogram)
M98 P20 L8	(Meghívjuk a 8-szor a fúrást végző eljárást)
G91	(Ismét relatív távolság-módba váltunk)
G00 X-100 Y15	(X-10*10 mm vissza pozicionálunk, Y+15 mm-t pedig előre)
G90	(Mindig visszaállítjuk az abszolút módot!)
M99	
O20	(A fúrást végző alprogram)
G01 Z-3	(Z=-3 mm mélyen fúrunk egy furatot)
G00 Z2	(2mm magasra kiemljük a szerszámot)
G91	(Átváltunk relatív távolság-módba!)
G00 X10	(X+10 mm-re pozicionálunk)
G90	(Visszaállunk abszolút távolság-módba)
M99	(Mindig a meghívó eljárásához tér vissza!)
%	(Fájlvége)



Nem muszáj saját fúró műveletet írunk, helyette használhatjuk itt is a G7x vagy G8x fúróciklusokat!

Megfigyelhető, hogy a kódunk ez esetben is tesz néhány szükségtelen „kitérő” lépést a végrehajtás során. A program módosításával ezek a fölösleges műveletek minimalizálhatók, néhány esetben azonban csak paraméterek segítségével kerülhetők el teljességgel. Ez utóbbiakra most nem térek ki. Mivel a Mach3 g-kód értelmezője nem ad lehetőséget feltételvizsgálatra, sem pedig feltételes (elől vagy hátultesztelés) ciklusvégrehajtásra, a fenti példa feleslegesen megtett mozgásait valamelyest csökkenthetjük, ha a ciklikus műveleteket nem a kezdőpontból (pontosabban nem az első furatunk koordinátáitól) indítjuk az alábbiak szerinti módosításokkal:

```

G00 X10 Y15      (Egy sor- és egy furattávolságnyt eltoljuk a kezdőpontunkat a nullpont felé.)
M3
M98 P10 L6
...
M5 M2
O10
G91
G00 Y15
G90
M98 P20 L8      (Eltérően az előbbi esettől, előbb lépünk Y irányba, csak utána hívjuk meg a sorozatfúró ciklust.)
G00 X10
M98

O20
G91
G00 X10
G90
G01 Z-3
G00 Z2          (szintén megváltoztattuk a sorrendet: először X irányba léptetünk és csak utána fúrunk.)
M99
%
```

Az iménti módosításból kiderül az is, hogy a programunkat (azon belül az eljárásainkat), akár optimalizálás, akár technológiai igények miatt, mindig tetszőlegesen átszervezhetjük. Egy-egy műveletsorozatot pedig sokféle képen megírhatunk úgy, hogy végeredményként ugyanazt a munkadarabot kapjuk.

Fontosnak tartom itt megemlíteni, hogy alprogramok írásánál **mindig különösen figyeljünk a szerszám helyzetére (általában a Z tengely helyzetére)**, mert egy-egy téves parancs akár azonnal szerszámtöréshez vagy egyéb károkhoz is vezethet (pl. fogásban hagyott szerszám esetén G00 kóddal pozicionálunk).

Végezetül nézzünk meg egy harmadik lehetőséget is, amit a Mach3 biztosít számunkra. A **G81** (zárt) ciklus használatával az alábbiak szerint módosítva, elegendő egy alprogram, a sorozatfűrást a kód elvégzi helyettünk. Fontos! A G81 kóddal csak úgy tudunk ciklikusan műveletet végezni, ha a relatív távolság módba váltunk (G91).

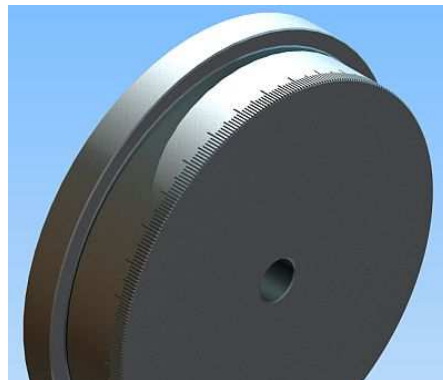
G00 X10 Y30	
M3	
M98 P10 L6	
G00 X0 Y0	
M2	
O10	
G91	
G81 X10 Z-3 L8	(Zárt ciklussal fűrünk sorozatot!)
G00 Y15 X-80	
G90	
M99	
%	

Példa forgató egység használatára

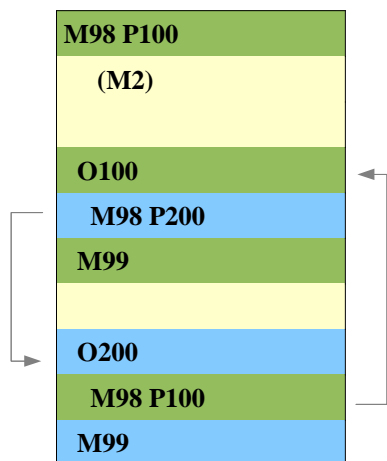
Egy hengerpalástra szeretnénk nóniusz skálát gravírozni 1°-os osztásonként, azon a jól ismert módon, hogy minden ötödik és minden tizedik osztást hosszabb vonalakkal emelünk ki. Forgató egységnek az A-tengelyt használjuk, az osztásokat pedig az X-tengely hosszában gravírozuk.

Mivel az első osztásunk 10. foknak megfelelő hosszú jelölés ezzel kezdjük a műveletsort, majd csökkenő sorrendben az 5. végül az egy fokok jelöléseit gravírozuk fel.

M3	<i>(Elindítjuk a főorsót – elhagyható kód)</i>
M98 P100 L36	<i>(A 10-es értékeket gravírozó ciklus)</i>
G01 A5	<i>(A köv. ciklus előtt beállítjuk 5 fokos kezdőpozícióba a forgatót)</i>
M98 P200 L36	<i>(Az 5. értékeket gravírozó ciklus)</i>
G01 A2	<i>(Most a 2. foktól kezdjük a ciklusunkat)</i>
M98 P200 L72	<i>(1 fokos osztások gravírozása)</i>
...	
M2	<i>(programvége)</i>
...	
O100	<i>(10. fokok gravírozása)</i>
G01 Z-0.15	<i>(Z=-0.15 mm mélyen gravírozunk)</i>
G01 X3.5	<i>(3.5 mm hosszú osztásokat gravírozunk)</i>
G00 Z5	
G91	<i>(Relatív módba kell váltatnunk)</i>
G00 A10 X-3.5	<i>(Az A tengelyt 10 fokkal elforgatjuk, miközben visszaállunk az X tengelyen nullára)</i>
G90	<i>(Mindig visszaállunk abszolút módba!)</i>
M99	
O200	<i>(5. fokok gravírozás.)</i>
G01 Z-0.15	
G01 X2.75	<i>(Most rövidebb vonalakat gravírozunk.)</i>
G00 Z5	
G91	<i>(Itt is relatív módba váltunk.)</i>
G00 A10 X-2.75	<i>(Mivel A=5 fokról indultunk, az első lépés után A=15 fokon fog állni a forgatónk.)</i>
M99	
O300	<i>(1 fokok gravírozása)</i>
M98 P400 L4	<i>(Egy segédciklust használunk 4x1 fok gravírozására)</i>
G91	
G00 A2	<i>(a köztes 4 jelzés után 2 fokokat lépünk relatív módban)</i>
G90	
M99	
O400	
G01 Z-0.15	
G01 X2.25	
G00 Z5	
G91	
G00 A1 X-2.25	
G90	
M99	<i>(innen az O300 alprogramba tér vissza a program!)</i>
%	



A Mach3 kézikönyve említést tesz arról, hogy feltételvizsgálat hiányában rekurzív eljáráshívásnak nincs különösebb értelme. Hogy erről is ejtsék néhány szót, nézzük mit is jelent ez.



Az első hívás meghívja az O100 alprogramot, ami pedig az O200-at hívja és ezzel eddig nincs is semmi probléma. Az O200 viszont ismét az O100-at hívja, ezáltal egy ún. „végtelen ciklus” keletkezik (ez tekinthető egyfajta „rövidzárlatnak” a programunkban :). Ebből csak feltételvizsgálat után lehetne „kiugrasztani” a program futását, különben a végtelenségig ugrálna ide-oda az O100 és O200 között. Ha a Mach3 ilyen kóddal szembesül, hibüzenetet küld (de nem kizárt, hogy kiakadhat ilyenkor).

Hasonló a helyzet akkor is, ha nagyon sok ciklust futtató alprogramok szintén nagyon sok ciklust futtató alprogramokat hívnak meg (esetleg mindezt többszörösen is teszik). Ilyenkor a Mach3 a „Too many nodes” hibüzenetet küldi, és nem futtatja le a programot!