

Sipos Jenő

sipos.jeno@zmne.hu

Apostol Attila

apostol.attila@gmail.com

Nagy Balázs

nblevelei@gmail.com

MAKRÓ PROGRAMOK FEJLESZTÉSE A CNC SZERSZÁMGÉPEK PROGRAMOZÁSÁBAN

Absztrakt

A cikkben bemutatjuk a makró programozás előnyét, alkalmazásának területeit. Bemutatjuk továbbá a tesztelt és alkalmazott, a FANUC alapú NCT104-M vezérlésekre fejlesztett sokszögzseb- illetve szigetmaró makrót. Az általunk kifejlesztett makrókat az olyan egyszerű munkadarabok legyártása tette szükségessé, melyek nem követelik meg a CAM programok alkalmazását. Makrónk elkészítése előtt egy ötszögzseb legyártása okozhatott problémát, mivel idáig nem létezett olyan program, ami lehetővé tette volna tetszőleges geometriai méretű sokszögzseb legyártását.

The article explains the benefits of macro programming, application areas. Introducing also tested and used in the FANUC NCT104-M-based controllers developed polygonal pocket or island milling macro. The macros we have developed a simple manufacture of the pieces made it necessary that do not require the use of CAM programs. Pentagon pocket been manufactured prior to completion of a macro could cause a problem, because until now there was a program that would allow arbitrary geometric dimensions polygonal pocket manufacture.

Kulcsszavak: *felhasználói makrók, FANUC programozás, változók ~ custom macros, FANUC programming, variable*

A paraméteres programozás előnyei

A paraméteres programozás óriási programozói szabadságot nyújt a CNC szerszámgépek programozásában. Azonban felmerül a kérdés, hogy az egyre többet tudó CAD/CAM rendszerek mellett van-e még létjogosultsága. A rendkívül összetett és bonyolult alkatrészek (pl.: turbinalapátok, 3D-s felületek megmunkálása) több tengelyes megmunkáló központokban, sok esetben egyidejűleg több tengely mentén történő megmunkálását csak CAD/CAM rendszerrel tudunk megvalósítani. A CAD/CAM rendszerek megjelenésével természetesen csökken a törekvés a programozókban a makrók fejlesztésére, de mint azt a vezérlések fejlesztésének iránya valamint a nálunk fejlettebb, a CNC technikában előbbre járó országok programozói mutatják,¹ a felhasználó számára egyszerűen használható, de bonyolult makrók még napjainkban is szép számmal készülnek. Jellemzően a kevésbé „bonyolult” alkatrészek gyártásához készítünk és használunk makrókat, főleg műhelyprogramozásban, 2.5-3D-s munkadarabok előállításánál, ahol egyidejűleg csak két, esetenként három tengely mentén történik a megmunkálás, valamint hajtott szerszámos kettő vagy több tengelyes esztergaközpontokban. Ezen eszközökön az alkatrészek a műhelyprogramozással is elkészíthetők, ezáltal nincs szükségünk CAD/CAM programokra, de felmerülhetnek olyan speciális felületek, feladatok (körívvel nem leírható görbék, ellipszisek, parabolaívek, különböző zseb- és szigetmarások, azonos alkatrészek gyártása, de eltérő méretekkel), melyek hagyományos programozással nem, de paraméteres programozással megvalósíthatók. Ennek a területnek jelent a paraméteres programozás és a felhasználói makró készítés nagyobb segítséget, mert a paraméteres² programozáshoz és a makrók³ készítéséhez nincs szükség külön programok megvásárlására, mivel ezt minden vezérlő alapkiépítésben tartalmazza. Néhány példa arra, ami meg tudunk valósítani paraméteres és felhasználói makró programozással: a fűróciklusok, menet-, fűró, maró ciklusok jelentős része, a koordináta transzformációk, mérőciklusok, spirál pályán történő zsebmárások, íves horony zsebek egyenes és lekerekített véggel. A fent felsorolt makrók természetesen nem csak marógépeken, hanem apróbb módosításokkal C-tengelyes hajtott szerszámos esztergaközpontokban, ellenorsós hajtott szerszámos megmunkáló központokban is megvalósíthatók.

A 32 bit alatti processzorral rendelkező vezérléseknél (nagysebességű megmunkálásnál) jelentkezhet némi hátrány a CAD/CAM programokkal szemben. Mégpedig az, hogy nem előre kiszámított adatokkal dolgozik, mint a CAM post processzált szerszámpálya, hanem programfutás közben számolja ki az elmozduláshoz szükséges koordinátákat. Emiatt elképzelhető, hogy a mozgás nem a megadott előtolással történik, hanem a processzoridő határozza meg, mivel a mozgásparancsot tartalmazó mondatok közé beékelődnek a számítást tartalmazó ciklusok, ami a több mondatot tartalmazó számítások során jelenthet problémát.

A gyorsabb processzorok terjedésével ez a probléma idővel megoldódik. A lassabb processzorok esetére is léteznek megoldások: a legegyszerűbb a számítást tartalmazó mondatok egyszerűsítése és a számítások kevesebb mondatba tömörítése. Lehetőség van arra,

1 Peter Smid: FANUC CNC Custom Macros

2 A paraméteres programozás lényegében nem más, mint az alkatrészprogram vagy alprogram konstans számértékeinek (amik lehetnek a kontúrt leíró mondatok koordinátái, technológiai adatok számértékei, ciklusok címei) változókkal történő helyettesítése. Ezen felül tartalmaz még aritmetikai műveleteket és matematikai függvényeket, feltételes kifejezéseket, elágazásokat, utasításokat. Ezáltal a programfutás tetszőleges matematikai függvények szerint alakulhat és feltételekhez köthető. Mindez együttesen óriási programozói szabadságot nyújt. Pl.: ha egy alkatrész méretadatait parametrizáljuk, vagyis a koordináták számértékeit változókkal helyettesítjük, akkor hasonló alakú, de eltérő méretű darabok esetén elég csak a megfelelő változók átírása - nem kell új programot készíteni.

3 Az igazi jelentősége azonban a felhasználói makró készítésben van. Azokat a speciális alprogramokat, amelyek egyáltalán nem tartalmaznak konstans számértékeket felhasználói makróknak nevezzük.

hogy a vezérlő az NC mondatok végrehajtása közben végezze el a makró utasításokat (ehhez az SBSTM paraméter értékét kell 1-re állítani).

A makró készítése során változókat alkalmazunk, amelyek nem mások, mint névvel ellátott memóriaterületek/regisztercímek, amelyeknek értéket adhatunk vagy a rendszer működéséről, állapotáról, szának helyzetéről szerezhetünk általuk információkat. A programmondat szavaiban a különböző címek nemcsak számértéket vehetnek föl, hanem változók értékeit is. A változók használatával a megfelelő méretadatok parametrizálhatók, ezáltal a programok sokkal rugalmasabbá tehetők. Bizonyos programok nem vagy csak nagyon nehezen készíthetők el változók használata nélkül.

A következőkben a BJKMK KGMBMI Fegyverzettechnikai, páncélos és gépjármű tanszék CNC, CAD/CAM képző- és vizsgaközpontjában a FANUC alapú NCT104M vezérlésre kifejlesztett és tesztelt makrók közül a sokszög sziget- és zsebmaró makrókat mutatjuk be.

A makrók matematikai alapjai

Egy sokszöget a legegyszerűbben egy kör sokszögesítésével állíthatunk elő, ugyanis ha egy körívet megfelelően nagyszámú pontokkal egyenlő részekre felosztunk, és a pontokat egymás után egyenesekkel összekötjük, akkor az így kapott szakaszok a körív burkológörbét adják. Minél jobban közelítjük a pontok számát a végtelenhez (ami által egyre rövidülnek a pontokat összekötő egyenesek), annál jobban közelíti a burkológörbe a kört. Természetesen a pontok számának csökkentésével egyre szögletesebb lesz a burkológörbe és a végén eljutunk a sokszögekhez. A pontok száma határozza meg a sokszög fajtáját. Ha a pontok száma 3 (ez a minimális érték, ennél kevesebb nem lehet) akkor egy háromszöget kapunk. Belátható, hogy a kör végeredményben végtelenszög.

A sokszög megadása paraméteresen: $X=R*\cos\alpha$

$$Y=R*\sin\alpha$$

X, Y a kör egy tetszőleges pontja, adott R-hez és

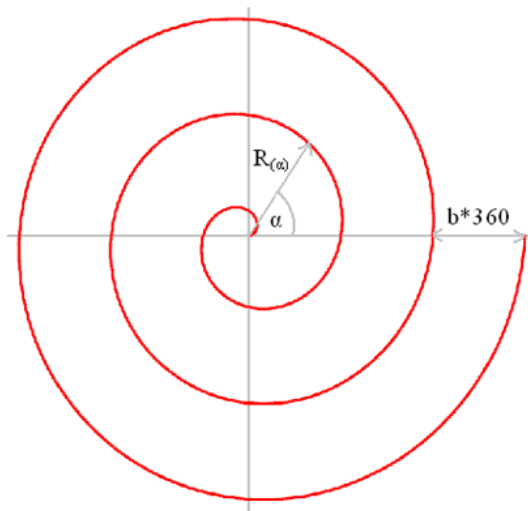
szögelforduláshoz

R a kör sugara, sokszög esetén a sokszög köré írható

kör sugara

α az R sugár szögelfordulása

Sokszög esetén, mivel a sokszög csúcspontjait írjuk le, az R sugárnak a sokszög köré írható kör sugarának kell lennie. Ha α -t 0-tól 360-fokig léptetjük, akkor a sokszög pontjait írjuk le. Ez ciklusszervezéssel, vagy feltételes elágazással megvalósítható. A sokszög fajtáját az α szög lépésmagysága határozza meg, ezt úgy kapjuk meg, hogy a 360 fokot elosztjuk a sokszög szögeinek számával. Pl. hatszög esetén az alfa szöget $360/6=60$ fokként növeljük 0-tól 360-ig. Az α szöget mindig a kiválasztott sík első tengelyétől számoljuk (G17-nél X tengely, G18-nál Z tengely) és iránya az óramutató járásával ellentétesen pozitív.



1. ábra. Az Arkhimédészi spirál, három 360°-os fordulata

A fentiekben ismertetett eljárással csak szigetet tudunk készíteni. Ahhoz, hogy egy zsebet tudjunk készíteni, a marót spirálpálya mentén kell mozgatnunk a zseb középpontjától a zseb kontúrjáig. A spirálpálya mentén történő szerszámmozgatás az Arkhimédészi spirálissal valósítható meg. Az Arkhimédészi spirál olyan spirális síkgörbe, mely azon pontok mértani helye, melyeket mozgása során pillanatnyilag elfoglal egy rögzített pont (a póluspont) körül egyenletes szögsebességgel forgó és a pólusponttól állandó sebességgel távolodó pont. Ezen spirál polárkoordinátás egyenlete a következő: $R(\alpha)=a+b*\alpha$, ahol a és b valós számok. Az „ a ” paraméter megváltoztatása elfordítja a pólus körül a görbét, a „ b ” paramétertől pedig a sorban következő fordulatok közötti távolság függ. Az egyenletből következik, hogy a és b valós számok, de értékük fokban értendő. A „ b ” paraméter nem más, mint a fordulatok közötti távolságnak és egy körülfordulás szögének (azaz 360 foknak) a hányadosa. Az egyenletből látható, hogy a sugár a szögelfordulással egyenes arányban növekszik, egy 360 fokos körülfordulás alatt 0-tól $b*360$ -ig. A teljes 360 fokos fordulatok, melyek száma függ a fordulatok közötti távolságtól és az elérni kívánt sugártól (ez nem más, mint a kör sugara, vagy a sokszög köré írható kör sugara).

Fordulatok száma=Sugár/fordulatok közötti távolság.

A spirál derékszögű egyenlete: $X=b*\alpha*\text{COS}[a+\alpha]$, $Y=b*\alpha*\text{SIN}[a+\alpha]$

A spirális ciklusszervezéssel vagy feltételes elágazással megvalósítható. Az előzőekben tárgyalt kör sokszögesítése erre az esetre is igaz. Ha tehát az α szög lépésmagysága megfelelően kicsi, akkor spirálvonal ív lesz, ha növeljük a lépésmagyságot, akkor a spirál szögletes lesz. Ha az α szöget a lépésmagysággal növeljük, akkor a spirál körüljárási iránya az óramutató járásával ellentétes. Ha az α szöget a léptetéssel csökkentjük, akkor a spirál körüljárási iránya az óramutató járásával megegyező. Az α szög mindig a kiválasztott sík első tengelyétől számítódik (G17-nél X tengely, G18-nál Z tengely) és iránya az óramutató járásával ellentétesen pozitív.

#100=0 (α)

#101=5 (a fordulatok közötti távolság, vagyis a b paraméter 360 szorososa)

#102=1 (az α szög léptetésének mértéke, mivel α egyesével növekszik így a spirál ív lesz)

#103=4 (teljes 360 fokos fordulatok száma)

#104=0 (a pólus körüli elforgatás szöge)

Ciklus:

WHILE [#100 LE [#103*360]] DO1

```
G1 X[#101/360]*#100*COS[#104+#100] Y[#101/360]*#100*SIN[#104+#100]]
#100=#100+#102
END1
```

Elágazás:

```
N1 G1 X[#101/360]*#100*COS[#104+#100] Y[#101/360]*#100*SIN[#104+#100]]
#100=#100+#102
IF [#100 LE [#103*360]] GOTO1
```

A fenti leírással egy olyan spirált hoztunk létre, amely

- 4 teljes fordulatból áll,
- a fordulatok közötti távolság 5 mm,
- a pólus körüli elforgatás nincs,
- a szög léptetése egy fokként történik,
- a körüljárás irány az óramutatóval ellentétes.

A spirál végpontjának a pólusponttól való távolsága $4 \cdot 5$, vagyis 20 mm. Ez csak a végpontra vonatkozik, a többi pontra nem. Ezért egy 20 mm sugarú körzseb elkészítéséhez szükséges még egy teljes körülfordulás, de már állandó (a szögelfordulástól független) 20 mm-es sugárral. Ehhez szükség van még a következő ciklusra:

#100=0 (α)

#101=20 (a kör R sugara sokszög esetén a sokszögmérete írható kör sugara)

#102=1 (az α szög léptetésének mértéke, mivel α egyesével növekszik így egy 360 szöget kapunk, vagyis a kört 360 ponttal osztottuk fel)

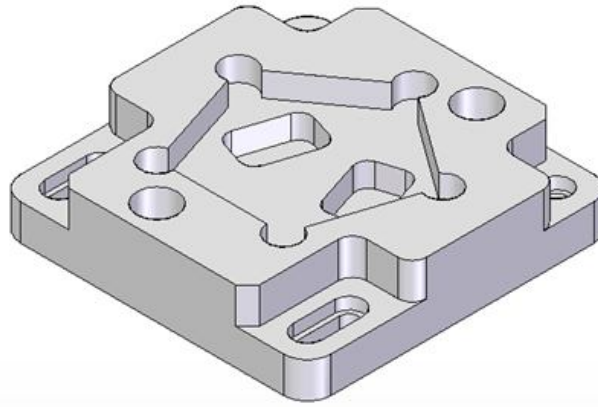
```
WHILE [#100 LE 360] DO1
G1 X[#101*COS#100] Y[#101*SIN#100] (#101 a kör sugara)
#100=#100+#102
END1
```

A fent ismertetett leírások az egyszerűség kedvéért nem foglalkoztak a szerszámátmérővel és a kontúrkövetéssel. Mivel a szerszám minden geometriai adata a megfelelő változókból leihívható ezért nincs is szükség kontúrkövetésre. (Ez zseb esetén a középről induló spirál miatt nem is lehetséges.) Csupán a sokszög megfelelő geometriai adatát kell változtatni a szerszámsugárral.

O9601 sokszögzseb makró ismertetése

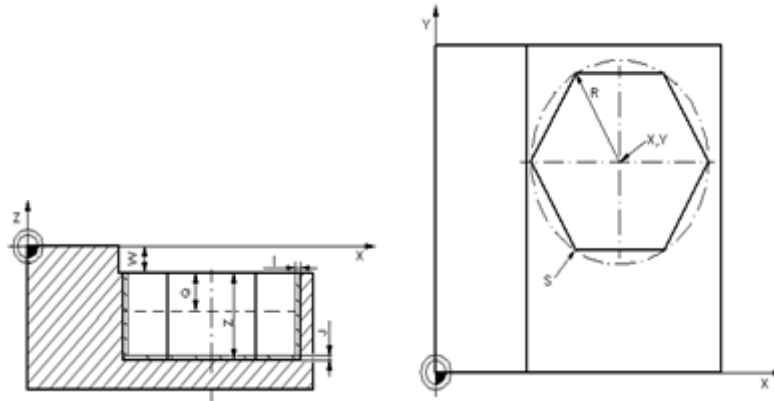
A következőkben a többször tesztelt általunk fejlesztett O9601 sokszögzseb felhasználó makrókat mutatjuk be a 2. ábrán látható sokszög zseb marási programján keresztül. A főprogram (melyet a CNC forgácsoló szakember ír) csak a következő mondatot tartalmazza:

```
G65 P9601 X0 Y0 Z10 U50 I0.2 S5 R45 K2 D6 Q5
```



2. ábra. A sokszögzseb makró alapján elkészült mintadarab

Az O9601 makró leírása: G65 P9601 X Y Z W U I J S R K D Q C



3. ábra. A sokszögzseb makró

A fenti utasítás a 3. ábra szerinti utasítással sokszögzseb maró ciklust indít. A 3. ábra segít a kért értékek értelmezésében. A ciklus végrehajtásához be kell tölteni az O9601 programszámú makró a programtárba. A makró a G17-es síkban használható, futtatása előtt a zseb X, Y koordinátájú középpontjára kell pozicionálni K biztonsági távolságnyra. Abban az esetben, ha a nullpontot azzal a szerszámmal mérjük be, amelyikkel a zsebet akarjuk készíteni, akkor hibaüzenettel leáll, mert ellenőrzi a szerszám hosszát. Ugyanígy, ha nagyobb a szerszám átmérője, mint a beírható kör sugara hibaüzenettel leáll. Ha nem adunk meg simítási ráhagyást, a névleges méretre készíti a zsebet.

A makró által kért paraméterek jelentése:

X: A zseb középpontja X irányban, abszolút értéként kell megadni;

Y: A zseb középpontja Y irányban; abszolút értéként kell megadni;

Z: A zseb mélysége, pozitív számértékként kell megadni;

W: A zseb felső síkja és a munkadarab nullpont távolsága Z irányban, növekményes értéként lehet megadni;

U: Fogásvétel X-Y irányban vett értéke, a szerszám átmérőjének adott százalékával, értéke 0-100 közötti számértékként kell megadni. Ha pozitív értéként adjuk meg, az óramutató járásával ellentétes irányban végez megmunkálást, ha negatív értéként adjuk meg, az óramutató járásával megegyező irányban végez megmunkálást;

I: Simítási ráhagyás X-Y irányban, megadása nem kötelező;

J: Simítási ráhagyás Z irányban, megadása nem kötelező;

S: Szögek száma, értéke 3 vagy annál nagyobb egész számértéket vagy 0-t kell megadni. 0 megadása esetén körzsebet készít el. Egész számtól eltérő megadás esetén hibajelzést küld;

R: A sokszög köré írható kör sugara, értéke pozitív nullától eltérő értéként kell megadni, más esetben hibajelzést küld;

K: biztonsági távolság Z irányban;

D: Megmunkálást végző szerszám tárhelye, értéke pozitív egész szám lehet;

Q: Fogásvétel Z irányban, értéke pozitív számként kell megadni, ha nagyobb, mint a megmunkálást végző szerszám átmérője akkor hibajelzést küld;

C: Az elforgatás szöge X tengelytől mérve, pozitív számérték esetén óramutató járásával ellentétes irányban, negatív számérték esetén óramutató járásával megegyező irányban forgatja el a koordináta rendszert.

A makró által használt regisztereket a 1. táblázat mutatja.

Változó típusa	Változó címe	Regiszter	Leírás
Argumentum átvadással	X	#24	Zseb középpontja X-irányban
	Y	#25	Zseb középpontja Y-irányban
	Z	#26	Zseb mélysége
	W	#23	Eltolás Z-irányban
	U	#21	Fogásvétel X-, Y- irányban
	I	#4	Simítási ráhagyás X-, Y-irányban
	J	#5	Simítási ráhagyás Z-irányban
	S	#19	Szögek száma
	R	#18	Köré írható kör sugara
	K	#6	Biztonsági távolság
	D	#7	Szerszám tárhely száma
	Q	#17	Fogásvétel Z-irányban
C	#3	Elforgatás szöge	
Globális		#100	Szerszám sugár
		#101	Szerszám átmérő
		#102	Fogásvétel X-, Y-irányban
		#103	Beírható kör sugara
		#104	Nagyolás X-, Y-irányban
		#105	Szerszám középpont számításához használt regiszter
		#106	Nagyolás Z-irányban
		#107	Léptetés a sokszög készítésekor
		#108	Elforgatás a sokszög készítésekor
		#109	Sokszög készítésekor használt regiszterek
		#110	Nagyoláshoz használt regiszterek
		#111	
		#112	Megmunkálás irányához használt regiszter
		#113	A sokszögre álláskor használt regiszterek
	#115		
	#114	Teszteléskor használt regiszter	
	#116	Ezeket az értékeket veszi fel ha körzsebet	

	#117	készít
	#150	Ciklusok léptetésekor használt regiszterek
	#151	
	#160	A mélység meghatározásához használt regiszter

1. táblázat. Regiszterek O9601

A következőkben a O9601-es programot és annak működését mutatom be.

%O9601(SOKSZGZSEB MAKRO)	O9601-es program
GOTO8	Feltétel nélküli elágazás az N8-es sorra
N1 #3000=301(HIBAS ERTEK ADAS S-RE)	Hibaüzenet küldése
N2 #3000=302(TUL NAGY FOGAS)	Hibaüzenet küldése
N3 #3000=303(TUL MELY FOGAS)	Hibaüzenet küldése
N4 #3000=304(HIBAS ERTEK ADAS D-RE)	Hibaüzenet küldése
N5 #3000=305(HIBAS ERTEK ADAS Z,J,Q)	Hibaüzenet küldése
N6 #3000=306(HIBAS ERTEK ADAS U,I,R)	Hibaüzenet küldése
N7 #3000=307(KICSI SZERSZAM ATMERO)	Hibaüzenet küldése
N8 IF[#6 LE0]#3006 =301(BIZONSAGI TAVOLSAG?)	Hibaüzenet küldése
IF[#19 LT0]OR[#19 EQ1] OR[#19 EQ2]OR[#19EQ#0]GOTO1 #114 =#19 MOD1 IF[#114 NE0]GOTO1	Vizsgálatot végez S-re
IF[#7 LT1]OR[#7EQ#0]GOTO4 #114 =#7 MOD1 IF[#114 NE0]GOTO4	Vizsgálatot végez D-re
IF[#26 LE0]OR[#26EQ#0] OR[[#17+#5]GT#26] OR[#5 LT0] OR[#17 LE0]OR[#26EQ#0] OR[#17 EQ#0]GOTO5	Vizsgálatot végez Z-re, Q-ra, J-re
IF[ABS#21 GT100]OR[#21 EQ#0] OR[#4 LT0] OR[#18 LE0]GOTO6	Vizsgálatot végez I-re, U-ra, R-re
IF[ABS[#106 +#23]GT#[10000 +#7] +#[11000 +#7]]GOTO3	Vizsgálatot végez a szerszám hosszára
IF[#21 LT0]#112 =-1 IF[#21 GT0]#112 =1	Értéket ad változóknak a nagyolás irányának meghatározására
IF[#19 EQ0]#116 =360 IF[#19 EQ0]#117 =1	Értéket ad változóknak
#116 =#19 #117 =#19 #100 =[#[12000 +#7]+#[13000 +#7]] #101 =[#100 *2] #102 =[ABS[#101 *#21 /100]] #103 =[ABS[#18 *COS[180 /#117]]] #104 =[#103- #4- #100] #105 =[[#103- #4- #100]/#102]	Értéket ad változóknak

#106 =[#26- #5] #107 =[360 /#116] #108 =[#107 /2] #110 =[#105 *360] #111 =#110 #113 =[[#116-2]*90 /#116] #115 =#113 IF[#19 EQ3]#113 =30 IF[#19 EQ4]#113 =45 IF[#19 EQ5]#113 =54 #160 =#17	
IF[2 *[#18- #103- [#100 /SIN#113] - #100]]GT#101]GOTO7	Vizsgálatot végez
IF[#17 GT#101]GOTO3	Vizsgálatot végez a fogás mélységre
IF[#100 GE#103]OR[#105 LT1]GOTO2	Vizsgálatot végez a fogás szélességre
G52 X#24 Y#25	Nullpont áthelyezése a zseb középpontjára
G68 X0 Y0 R#3	Koordinátarendszer elforgatása
G1 ZI#23	Ha a zseb felső szintje eltér a nullpont síkjától és lett megadva W-re érték, akkor oda pozicionál
GOTO10	Feltétel nélküli elágazás
N30 G1 X0 Y0 Z[- #106 +#6 +#17- #23]	Az utolsó nagyoló fogáskor innen kezd a megmunkálást
N10 #150 =0	Értékadás változónak
WHILE[#150 LE360]DO1 G1 X[#150 *#102 /720 *COS#150] Y[#112 *#150 *#102 /720 *SIN#150] ZI- [[#6 +#17]/361] #150 =#150 +1 END1	Ciklus kezdete: egy spirálvonal mentén lesüllyed egy fogásvételnyivel Z irányban, egy teljes kört tesz meg
#151 =#150	Értékadás változónak
WHILE[#151 LE720]DO1 G1 X[#151 *#102 /720 *COS#151] Y[#112 *#151 *#102 /720 *SIN#151] #151 =#151 +1 END1	Ebben a két ciklusban egy spirálvonal mentén kinagyolja a zsebet a beírható kör sugaráig
WHILE[#150 LE#110]DO1 X[#150 *#102 /360 *COS#150] Y[#112 *#150 *#102 /360 *SIN#150] #150 =#150 +1 END1	
WHILE[#150 LE[#110 +360]]DO1 X[#104 *COS#150] Y[#112 *#104 *SIN#150] #150 =#150 +1 END1	Következő ciklus kezdete: egy kört tesz meg a beírható kör sugara mentén
#160 =#160 +#17	Értékadás változónak
G1 ZI#6	Pozicionál
G0 X0 Y0	Pozicionál

IF[#160 LE#106]GOTO10	Vizsgálatot végez, hogy elérte-e az adott mélységet, ha nem visszaküldi az N10-es sorra
IF[#160 EQ#106 +#17]GOTO20	Ha elérte a kívánt Z mélységet, akkor N20-as sorra megy
IF[#160 GT#106]THEN #160 =#106 GOTO30	Utolsó fogás Z irányban
N20 G1 Z- [#26 +#23]	Lesüllyed adott Z mélységig
IF[#5 EQ0]#150 =720 IF[#5 EQ#0]#150 =720 #151 =0	Vizsgálatot végez, hogy van-e megadva Z irányban simítási ráhagyás
WHILE[#151 LE[#150-180]]DO1 X[#151 *#102 /360 *COS#151] Y[#112 *#151 *#102 /360 *SIN#151] #151 =#151 +1 END1	Értékkadás változónak
IF[#5 EQ0]GOTO50 IF[#5 EQ#0]GOTO50	A következő ciklus kezdete: megtesz a zseb alján egy kört
WHILE[#151 LE#150-360]DO1 X[#104 *COS#151] Y[#112 *#104 *SIN#151] #151 =#151 +1 END1	Ha van megadva Z irányú simítási ráhagyás, akkor a következő ciklusra lép és besimítja a zseb alját
N50 G1 ZI#6	A következő ciklus kezdete, Z irányú simítás
G0 X0 Y0	
G1 Z- #23	
#160 =#17	Pozicionál a sokszög elkészítése előtt
G16	Értékkadás változónak
N70 WHILE[#160 LE#26]DO1 #109 =1 G1 X#104 Y#113 Z- [#160 +#23] X[#18- [#100 /[SIN#115]]]YI[90- #113] WHILE[#109 LE#116]DO2 YI[360 /#116] #109 =#109 +1 END2 #160 =#160 +#17 END1	Polárkoordináta bekapcsolása
IF[#160 EQ[#17 +#26]]GOTO6 IF[#160 GT#26]THEN#160 =#26 GOTO70	Ebben a ciklusban készíti el a sokszöget a kívánt mélységben
N60 G15	
G52 X0 Y0	Vizsgálatot végez, hogy elérte-e az adott Z mélységet
G0 X#24 Y#25 Z#6	Polárkoordináta kikapcsolása
G69	Nullpont eltolás visszaállítása
M99%	Pozicionálás
	Elforgatás kikapcsolása
	Makró vége

A makró futtatásakor előforduló hibaüzeneteket a 2. táblázat mutatja

Üzenet kódja	Üzenet szövege	Leírása	Törlés és hibaelhárítás módja
4301	HIBAS ERTEK ADAS S-RE	Ha a szögek számának negatív- vagy törtszámot, vagy 1-et, vagy 2-t adott meg, vagy nem adott értéket S-nek.	Az üzenetet a RESET gomb törli, ha ezután START-ot adunk és nem javítjuk ki a hibát, az üzenet újra kiíródik
4302	TUL NAGY FOGAS	Ha a szerszám átmérője nagyobb, mint a sokszögbe írható kör sugara.	
4303	TUL MELY FOGAS	Ha a szerszám hossza kisebb, mint a zseb alsó síkja és a nullpont közötti távolság. Ha a megmunkáló szerszámmal mérte be a nullpontot.	
4304	HIBAS ERTEK ADAS D-RE	Ha a szerszám tárhelynek 0 vagy kisebb számot, vagy 100-nál nagyobbat vagy törtszámot adott meg. Ha nem adott értéket D-nek.	
4305	HIBAS ERTEK ADAS Z,J,Q	Ha zseb mélysége, vagy a Z-irányú simítási ráhagyás 0-nál kisebb szám. Ha fogásvétel és a Z-irányú simítási ráhagyás összege nagyobb, mint a zseb mélysége. Ha a fogásvétel 0 vagy nincs kitöltve. Ha nem adott értéket Z-nek vagy Q-nak.	
4306	HIBAS ERTEK ADAS U,I,R	Ha a fogásvétel X-, Y-irányba abszolút értékének 100-nál nagyobb számot, vagy 0-át adott meg, vagy nem adott értéket. Ha X-, Y-irányú simítási ráhagyásnak 0 vagy annál kisebb számot adott meg. Ha a köré írható kör sugarának 0 vagy kisebb számot adott meg. Ha nem adott értéket R-nek.	
4307	KICSI SZERSZAM ATMERO	Ha a szerszám átmérője kicsi, mert ekkor anyagot hagy a sarkokban.	
5301	BIZONSAGI TAVOLSAG?	Ha biztonsági távolságnak 0 vagy kisebb számot adott meg.	Az üzenet hatására az NC STOP állapotot vesz fel. START hatására a következő mondatra lép. Az üzenetet a RESET gomb törli.

2. táblázat. Hibaüzenetek O9601

Felhasznált irodalom

1. Mátyási Gyula - Sági György: Számítógéppel támogatott technológiák CNC, CAD/CAM, Műszaki Könyvkiadó, Budapest, 2007.
2. Peter Smid: FANUC CNC Custom Macros, Industrial Press, Inc., New York, 2005
3. www.nct.hu
4. www.iscar.com

A cikkben bemutatott programok

1. Solid Edge ST2 program
2. NCT-104M program