

GRIFF

3-dimenziós grafika



VIDEOTON

TV-Computer

TV COMPUTER

3-DIMENZIÓS

GRIFF GRAFIKA

FELHASZNÁLÓI KÉZIKÖNYV

Minden jog fenntartva!

Felelős kiadó: Dr. BARÁTH CSABA

Készült: Forma-Art

TARTALOM

1. Bevezetés	5
2. Mozgatás a térben	6
Relatív mozgató függvények	9
Nem relatív mozgató függvények	13
3. További függvények	14
4. Térbeli objektumok leképezése a képernyőre	16
Vetítést leíró függvények	18
5. A griff-grafika globális változói	19
6. Egyéb tudnivalók a rajzoláshoz	20
7. A program betöltése	20
8. Grafikai előtétprogram használata	21
9. Bemutató program	22
 Mellékletek: DEMO program lista	 27
Grafikai függvények összefoglaló táblázata	40

1. BEVEZETÉS

A griffmadár grafika lehetővé teszi a felhasználó számára pontok, vonalak, geometrikus alakzatok, kitöltött területek rajzolását egy 3-dimenziós térben. Ez a grafika a BASIC rendszer egy kiterjesztése, így a felhasználó saját BASIC programjaiban használhatja – kombinálva az eredeti BASIC grafikus parancsaival is.

A griff egy kibernetikai lény, amelyik a 3-dimenziós térben „él” és amelyet különböző parancsokkal mozgathatunk. Ha a griffnek azt a parancsot adjuk, hogy tegye le a „toll”-at, akkor a térben nyomot is hagy, amely ún. vetítés útján lesz leképezve a 3-dimenziós térből a 2-dimenziós képernyőre.

A grafikai parancsokat beépített függvények alkalmazásával adhatjuk ki, ahol a függvények első paramétere a parancs hivatkozási száma (címe), a második pedig a parancs tényleges argumentuma lesz.

```
PI.  
X=USR(6768,1)  
X=USR(6771,100)
```

Jelentése: tollat emeld fel és menj előre 100 egységet.

Szemléletesebb parancsokkal ugyanez a programrész:
(lásd később a grafikai előtétprogram)

```
PEN=6768:UP=1:FWARD=6771  
X=USR(PEN,UP)+USR(FWARD,100)
```

(A függvények értéként – mint mellékhatás – általában az éppen megváltoztatott paraméterek előző értékét adják vissza.)

A program használata előtt javasoljuk áttanulmányozni a BASIC programozási segédletben az USR (X, Y) függvény leírását.

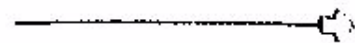
2. MOZGATÁS A TÉRBEN

Képzeljük el, hogy a térben mozgó griffmadár végrehajtja a parancsainkat és ennek megfelelően repül, útján nyomot hagyva változtatja helyét. Olyan egyszerű parancsokat közlünk, mint előre, hátra, emelkedj, fordulj, billenj, tollat fel. (A nyelvben a parancsok angol elnevezése szerepel.)

Figyeljük meg a griff mozgását először a síkban egyszerű parancsokat kiadva.



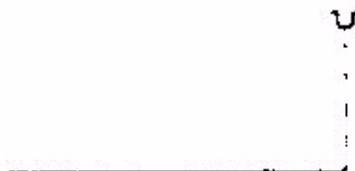
a. griff alaphelyzetben



b. X=USR(FWARD,100)



c. X=USR(TURN,90)



d. X=USR(FWARD,50)

1. ábra

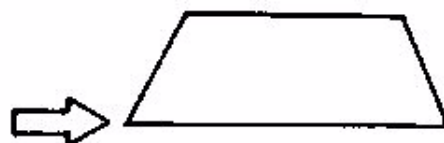
A síkbeli mozgáshoz hasonlóan, a griffet a 3-dimenziós térben is mozgathatjuk:

- térbeli forgatással – pl. billenés, vagy
- térbeli elmozdulással – pl. emelkedés, előremozdulás

Négyzet rajzolása a képernyőre merőleges síkban:

a) billenéssel:

```
10 FOR I=1 TO 4
20 X=USR(FWARD,HOSSZ)+USR(TILT,90)
30 NEXT I
```



b) térbeli (irányváltás nélküli) mozgással:

```
10 X=USR(FWARD,HOSSZ)+USR(SINK,HOSSZ)
20 X=USR(BACK,HOSSZ)+USR(LIFT,HOSSZ)
```

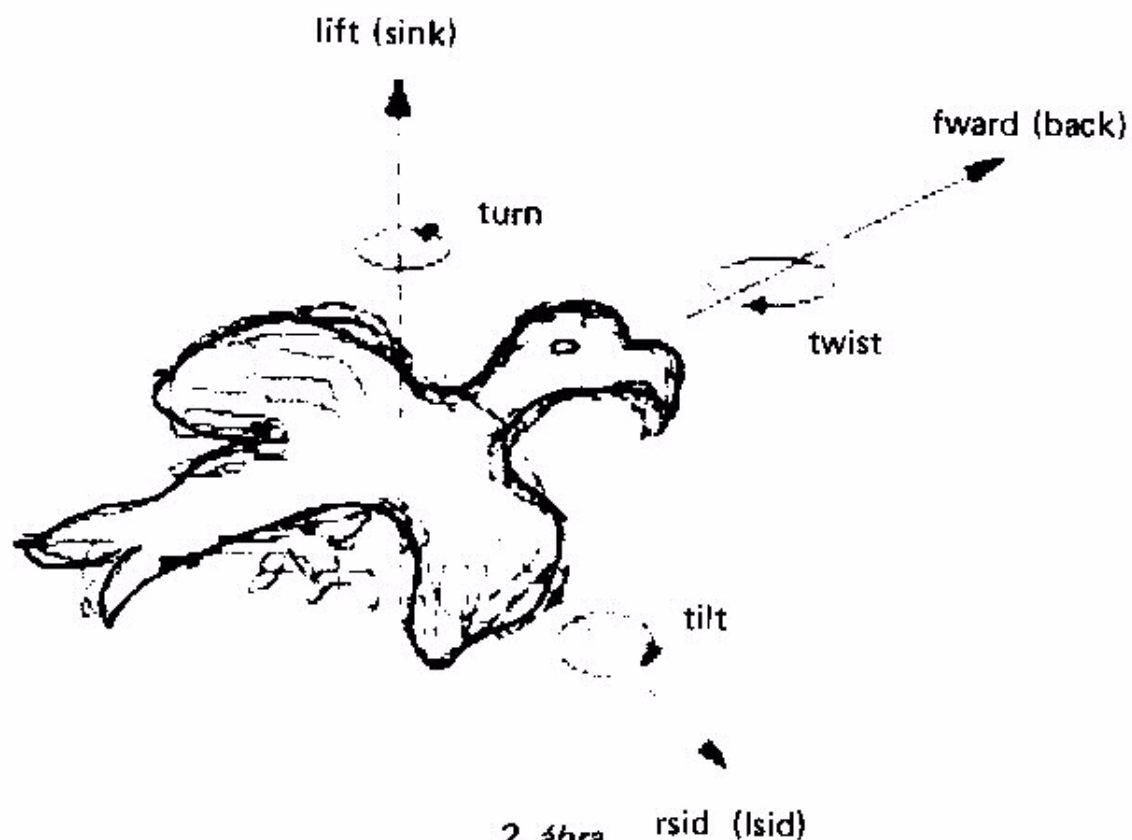
(FWARD—előre, BACK—hátra, SINK—süllyedj, LIFT—emelkedj, TILT—billenj)

* Megjegyzés: A négyzet rajzoló programok hívása előtt értéket kell adni a HOSSZ paraméternek, pl.:

5 HOSSZ=200

Ne feledkezzünk meg az előtét program meghívásáról sem: GOSUB 9000

A griffmadár olyan kurzornak tekinthető, amely a térbeli helyen kívül egy térbeli irányítással is rendelkezik. A térbeli irányítás kifejez egy (a griff szárnyai által kifeszített) síkot, valamint egy (a törzs által meghatározott) irányt. Másképpen kifejezve: a griffnek saját koordináta tengelye van, amelynek X-tengelye a griff törzse, Y-tengelye a szárnyak vonala, Z-tengelye pedig a szárnyak síkjára merőleges.



A mozgás kétféle értelemben történhet:

- abszolút mozgás, amikor egy adott térbeli helyzetet, ill. irányítást írunk elő, vagy
- relatív mozgás, amikor egy adott helyzethez viszonyítva írjuk elő az elmozdulást.

Relatív mozgással elérhetjük azt, hogy egy alakzatot könnyedén transzformálhassunk, azaz elmozdítsuk vagy forgassuk. Ehhez az szükséges, hogy a griffet egy tetszőleges kezdeti állapotba hozzuk, majd relatív mozgásokkal adjuk meg a tárgyat: a további rajzolás során a kiinduló állapot lesz a viszonyítás, azaz egy új koordináta rendszer, amelyben az alakzatunkat látjuk.

RELATÍV MOZGATÓ FÜGGVÉNYEK

FWARD hívása: USR(6771,L)
Adott helyzetéből L egységgel előre mozdítja a griffet.

BACK hívása: USR(6774,L)
Adott helyzetéből L egységgel hátra mozdítja a griffet.

LIFT hívása: USR(6783,L)
Adott helyzetéből L egységgel felemeli a griffet.

SINK hívása: USR(6786, L)
Adott helyzetéből L egységgel lesüllyeszti a griffet.

LSID(E) hívása: USR(6777,L)
Adott helyzetéből L egységgel balra mozdítja a griffet.

RSID(E) hívása: USR(6780,L)
Adott helyzetéből L egységgel jobbra mozdítja a griffet.

A fenti elmozdulások során a griff irányítása nem változik.

TURN hívása: USR(6792,A)
Adott helyzetéből balra fordítja a griffet (Z. tengelye körül), miközben a griff megmarad a síkjában). A forgatás nagyságát a fokban megadott A paraméter értéke adja meg.

TWIST hívása: USR(6798,A)
Adott helyzetéből törzse mentén elfordítja a griffet az óramutató járása szerint (X-tengelye mentén) A fokkal.

TILT hívása: USR(6795,A)
Adott helyzetéből előrebillenti a griffet (Y-tengelye mentén) A fokkal.

A fenti elfordulásoknál a griff pozíciója nem változik.

Alakzatok mozgatása, forgatása a griff-grafika segítségével igen egyszerűen megoldható. Hagyományos eszközökkel nehézség nélkül rajzolható egy négyzet, amíg annak élei párhuzamosak a koordináta tengelyekkel. Ha azonban – ahogy általában – ez nem áll fenn, akkor rögtön bonyolulttá válik a rajzolás, sok számítási művelet szükséges. A griff-grafika használatánál csupán a griff kezdeti helyzetét kell állítanunk, – pl. elforgatni – és az utána kirajzolt ábra a megfelelő helyre és helyzetbe kerül anélkül, hogy a felhasználónak bármit is foglalkoznia kellene a transzformációs számításokkal.

Például egy négyzet-rajzolás programja (szubrutin) síkbeli forgatással:

```
100 REM *** négyzet rajzolás ***
110 FOR I=1 TO 4
120 X=USR(FWARD,HOSSZ)+USR(TURN,90)
130 NEXT I
140 RETURN
```

Rajzoljuk ezután a négyzetet a képernyőre X-irányba eltolva, majd elforgatva úgy, hogy a négyzet élhosszúsága 100 legyen:

```
5 GOSUB 9000:CLS
10 X=USR(PEN,UP)+USR(FWARD,55)+USR(TURN,45)
  +USR(PEN,DOWN)
20 HOSSZ=100:GOSUB 100:END
```



3. ábra

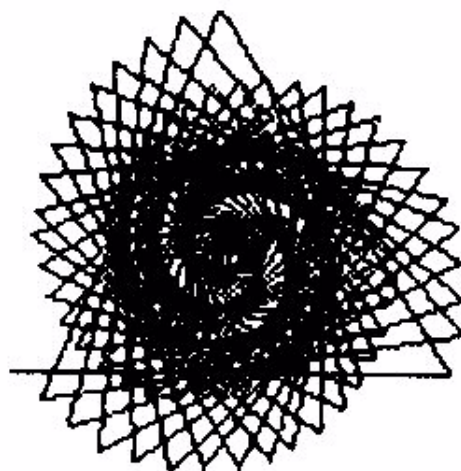
Több tengely körüli forgatással térbeli alakzatot is rajzolhatunk. Ennek legegyszerűbb példája kocka megrajzolása a négyzetrajz felhasználásával:

```
200 REM *** kocka rajzolása ***
210 FOR J=1 TO 4
220 GOSUB 100
230 X=USR(FWARD,HOSSZ)+USR(LEFT,90)
240 NEXT J
250 RETURN
```

Megjegyzés: Ez a programrész is szubrutinként használható. A GOSUB hívás előtt értéket kell adni a hossz paraméternek.

Relatív mozgatással már egyszerű programmal is látványos, játékos geometriai alakzatokat rajzolhatunk:

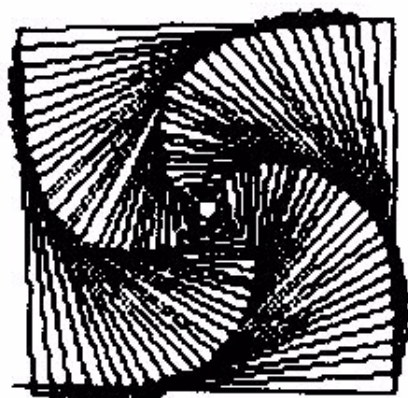
```
5 GOSUB 9000: X=USR(RESET)+USR(POSX,400)
  +USR(POSY,400):CLS:SZOG=123
10 FOR I=200 TO 5 STEP -5
20 X=USR(FWARD,I)+USR(TURN,SZOG)
30 NEXT I
40 END
```



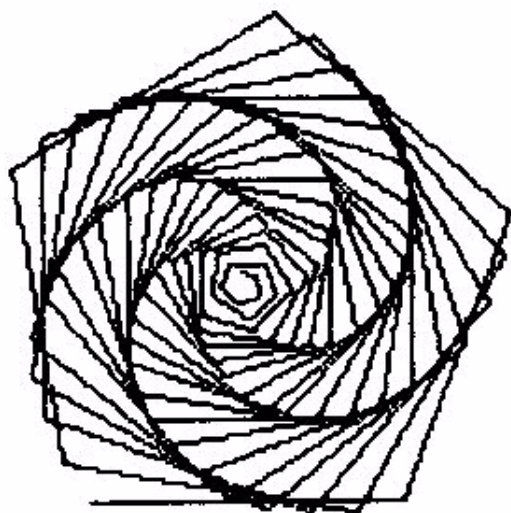
szog=123

4. ábra

Az előző példa program eredménye nagy mértékben különbözik, ha az 5. sorban a SZOG változónak más értéket adunk:



szog=91



szog=70

NEM RELATIV MOZGATÓ FÜGGVÉNYEK

POSX	hívása: USR(6720,X) A griff X koordináta értékét X-re állítja
POSY	hívása: USR(6723,Y) A griff y koordináta értékét Y-ra állítja
POSZ	hívása: USR(6726,Z) A griff z koordináta értékét Z-re állítja
MOVX	hívása: USR(6738,X) Beállít egy X-koordináta értéket, ahová a griff egy MOVE parancs kiadása után kerül, miközben vonalat is húz.
MOVY	hívása: USR(6741,Y) Beállítja az Y-koordináta értéket a MOVE parancshoz (lásd MOVX).
MOVZ	hívása: USR(6744,Z) Beállítja a Z-koordináta értéket a MOVE parancshoz (lásd MOVX).
MOVE	hívása: USR(6789) Adott helyzetből elmozdítja a griffet a MOVX, MOVY, MOVZ parancsokkal adott pontba, miközben vonalat húz.
GPOSX	hívása: USR(6729) A függvényérték megadja a griff X-koordináta értékét.
GPOSY	hívása: USR(6732) A függvényérték megadja a griff Y-koordináta értékét.
GPOSZ	hívása: USR(6735) A függvényérték megadja a griff Z-koordináta értékét.

3. TOVÁBBI GRAFIKAI FÜGGVÉNYEK

- PEN** hívása: USR(6768,X)
A tollat $X=0$ értékénél leteszi, egyéb értéknél felemeli.
- FILL** hívása: USR(6801)
A toll színére kiszínezi a zárt görbén belüli felületet az aktuális pontból kiindulva.
- VISI(BLE)** hívása: USR(6810)
- RESET** hívása: USR(6804)
A rendszer globális változóit alaphelyzetbe állítja. Ezeket a default értékeket lásd külön.
- RESHD** hívása: USR(6807)
A griff irányítást (HEADING) alaphelyzetbe állítja (síkja a képernyősík, irány az X-tengely)
- DOT** hívása: USR(6813)
Letesz egy pontot a griff aktuális pozíciójában.
- LOGO** hívása: USR(6819)
Speciális függvény, a griff grafika bejelentkező fantázia-jelét rajzolja.

VISI(BLE)

A függvény hívása igaz (0) értéket ad, ha a griff háta látható, azaz síkjának a normálisa kifelé mutat, egyébként hamis (értéke -1). Ez a függvény arra alkalmas, hogy síklapokkal határolt konvex testek esetében a nem látható éleket eltüntessük (azáltal, hogy a nem látható síkot nem rajzoljuk meg).

A kockarajzolásnál a takart éleket eltüntetjük úgy, hogy a határoló négyzetek megrajzolását tesszük feltételelessé a láthatóság alapján. Vigyáznunk kell arra is, hogy most nem elég a határoló éleket megrajzolni, hanem az összes határoló síkot is szükséges bejárni, mintegy befedni a testet.

A módosított rajzoló program:

```
5 GOSUB 9000: X=USR(RESET)+USR(POSX,50)
+USR(POSY,100):CLS:HOSSZ=200:GOSUB300:END
100 REM *** négyzet láthatósággal ***
105 IF USR(VISI)=-1 THEN RETURN
110 FOR I=1 TO 4
120 X=USR(FWARD,HOSSZ)+USR(TURN,90)
130 NEXT I
140 RETURN
200 REM *** kocka 4 palástja ***
210 FOR J=1 TO 4
220 GOSUB 100
230 X=USR(PEN,UP)+USR(FWARD,HOSSZ)
235 X=USR(TILT,90)+USR(PEN,DOWN)
240 NEXT J
250 RETURN
300 REM *** kocka takart élekkel ***
310 GOSUB 200
320 X=USR(TWIST,90)+USR(TURN,-90)
330 GOSUB 200
340 X=USR(TURN,90)+USR(TWIST,-90)
350 RETURN
```

4. TÉRBELI OBJEKTUMOK LEKÉPEZÉSE A KÉPERNYŐRE

A 3-dimenziós objektumok a számítógép képernyőjén közvetlenül nem láthatók, ezért egy transzformációval, a vetítéssel nyerhetünk 2-dimenziós, síkbeli alakzatot, amely már a képernyőn megjeleníthető. A griff grafika 2 különböző vetítést használ, amelyek között a felhasználó választhat:

- Perspektivikus (centrális) vetítés, amely a valóság illúzióját kelti.
- Párhuzamos (lineáris) vetítés, amely a mérnöki gyakorlatban jól használható (pl. axonometriák).

Az egyes vetítésekhez a felhasználó megadhatja a paramétereket; a perspektivikus vetítéshez a nézőpont koordinátákat (V_x , V_y , V_z) ill. a párhuzamos vetítéshez az axonometria paramétereket (a_x , a_y , a_z , b_x , b_y , b_z).

A párhuzamos (matematikaileg lineáris) vetítés megadja a leképezést a térbeli objektum pontjai és a képernyőpontok között a következők szerint:

$$\begin{aligned}x' \cdot 16384 &= a_x \cdot X + a_y \cdot Y + a_z \cdot Z \\y' \cdot 16384 &= b_x \cdot X + b_y \cdot Y + b_z \cdot Z\end{aligned}$$

ahol x', y' – képernyő koordináták
 X, Y, Z – 3-dimenziós koordináták
 $a_x, \dots, b_z$ – vetítési paraméterek.

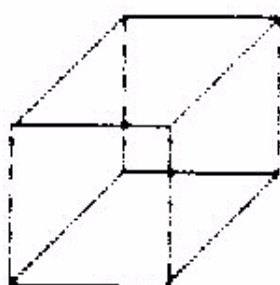
A következő táblázat néhány – a mérnöki gyakorlatban is – ismert párhuzamos vetítés (axonometria) paraméterét tartalmazza. (Az értékeket 16384-gyel meg kell szorozni.)

	a_x	a_y	a_z	b_x	b_y	b_z
előlnézet	1	0	0	0	1	0
felülnézet	1	0	0	0	0	1
oldalnézet	0	0	1	0	1	0
front axon.	1	0	0.707	0	1	0.707
izometria	0.866	0	-0.866	0.5	1	0.5
x-axon.	0.866	0	-0.5	0.5	1	0.433

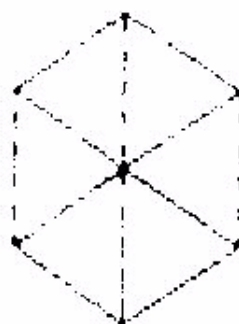
A kocka képe különböző párhuzamos vetítésekkel:



nézetek



front axonometria



izometria



x-axonometria

5. ábra

A perspektivikus vetítésnél a nézőpont X , Y koordinátája síkbeli koordinátát határoz meg a képernyőn, míg Z értéke a fókusztávolság (képernyő előtt pozitív). Kezdeti értékek $V_x, V_y, V_z = (512, 480, 2000)$

VETÍTÉST LEÍRÓ FÜGGVÉNYEK

PROJ hívása: USR(6765,L)

L értékétől függően változik a vetítés típusa. Egyidejűleg egyféle vetítés alkalmazható, ezért pl. párhuzamos vetítésnél a perspektivikus vetítés paraméterei nem értelmezhetők.

Fontos! A párhuzamos vetítés első három paramétere a memóriában azonos területen van a nézőpont értékekkel, ezért ezeket a vetítés váltásakor új értékre kell állítani.

L jelentése: L = 0 — perspektivikus vetítés

L = 1 — párhuzamos vetítés

AX hívása: USR(6747,AX)
A párhuzamos vetítés AX paraméterét adjuk meg.

AY hívása: USR(6750,AY)
A párhuzamos vetítés AY paraméterét adjuk meg.

AZ hívása: USR(6753,AZ)
A párhuzamos vetítés AZ paraméterét adjuk meg.

BX hívása: USR(6756,BX)
A párhuzamos vetítés BX paraméterét adjuk meg.

BY hívása: USR(6759,BY)
A párhuzamos vetítés BY paraméterét adjuk meg.

BZ hívása: USR(6762,BZ)
A párhuzamos vetítés BZ paraméterét adjuk meg.

VIEWX hívása: USR(6747,VX)
A perspektivikus vetítés Vx paraméterét adjuk meg.

VIEWY hívása: USR(6750,VY)
A perspektivikus vetítés Vy paraméterét adjuk meg.

VIEWZ hívása: USR(6753,VZ)
A perspektivikus vetítés Vz paraméterét adjuk meg.

5. A GRIFF GRAFIKA GLOBÁLIS VÁLTOZÓI

változók	abszolút cím	kezdeti érték
posx	6822	0
posy	6824	0
posz	6926	0
movx	6928	0
movy	6930	0
movz	6932	0
hdex	6934	16384
hdey	6936	0
fdez	6938	0
hdfx	6940	0
hdfy	6942	16384
hdfz	6944	0
hdgx	6946	0
hdgy	6948	0
hdgz	6950	16384
ax(Vx)	6952	512
ay(Vy)	6954	480
az(Vz)	6956	2000
bx	6958	0
by	6960	0
bz	6962	0
proj	6964	0
pen	6965	0

(A változónevek az értéket beállító/lekérdező függvények nevének kisbetűs változata.)

Megjegyzés: A hdex hdgz változók az ún. heading mátrix (a griff koordináta rendszerét az abszolút koordináta rendszerre leképező transzformációs mátrix) értékeit tartalmazza. Ezek egyértelműen leírják a griff térbeli irányítását. A különböző forgatások automatikusan állítják ennek értékeit.

6. EGYÉB TUDNIVALÓK A RAJZOLÁSHOZ.

A képernyő véges felbontása miatt a ferde egyenesek rajzolásánál fellép az ún. „lépcső hatás”. Ezért vonalas (nem kitöltött) rajzoknál — ha lehetséges a színek elhagyása — használjunk nagy felbontást (GRAPHICS 2).

Másik jelenség a vonalrajzolásnál a „keret-hatás”. Ha egy vonal a keretet elérné, akkor az nem rajzolódik ki mindaddig, amíg a vonal vissza nem tér a képernyő területére. Ekkor a kilépő vonal kezdete és a visszalépő vonal végpontja lesz összekötve, ami torz rajzot eredményezhet. Ezért magunk gondoskodjunk arról, hogy az egyenesek a rajzterületen belül maradjanak.

Programunkban vigyázzunk arra is, hogy ne használjunk olyan neveket, változókat, amelyek kezdete egybeesik bármely BASIC paranccsal. A később látható előtétprogramban is ezért nem használhattunk FORWARD (előre) parancsot, hanem helyette a FWARD nevet alkalmazzuk.

7. A PROGRAM BETÖLTÉSE

A TV Computerhez adott leírásnak megfelelően helyezzük üzembe a TV készüléket, a magnót, számítógépet.

Tegyük be a kazettát a magnóba, gépeljünk be "LOAD" és nyomjuk meg a "RETURN" gombot. A képernyőn meg fog jelenni a "Searching", majd néhány másodperc múlva a "Reading: GRAF" felirat. A program betöltése 20–30 másodpercig tart. A betöltés végén a képernyőn megjelenik egy fantáziarajz és a "3-D graphics loaded" felirat. A betöltést követően BASIC környezetben dolgozhatunk úgy, hogy az alapkészlet mellett a griff grafika parancsait is használhatjuk.

8. GRAFIKAI ELŐTÉTPROGRAM HASZNÁLATA

A kényelmesebb programozás érdekében érdemes használni a "gr-front-end" (előtét) programot is, amely azt biztosítja, hogy a parancsok argumentumaként a nehezen megjegyezhető címek helyett a parancs szimbólumát használjuk. Erre láthattunk példát a bevezetőben. Az előtét program a kazettán a GRAF program után helyezkedik el, betöltése hasonló az alapprograméhoz. A "LOAD" begépelése és "RETURN" lenyomása után megjelenik a "Searching", majd "Reading: GR-FRONT-END" felirat.

A programokban úgy használjuk az előtétet, hogy betöltése után írjuk meg a programot, amelynek első utasítása gosub 9000 legyen. Ne használjunk 9000 ill. ennél magasabb címkét. Ezután kimentjük a programunkat, amely már tartalmazza az előtétet, ezért a legközelebbi betöltésnél már ne hívjuk be külön az előtétet.

Az előtétprogram:

```
9000 TURN=6792:TILT=6795
9010 TWIST=6798:PEN=6768
9020 UP=1:DOWN=0
9030 FORWARD=6771:BACK=6774
9040 LSID=6777:RSID=6780
9050 LIFT=6783:SINK=6786
9060 POSX=6720:MOVX=6738
9070 POSY=6723:MOVY=6741
9080 POSZ=6726:MOVZ=6744
9090 MOVE=6789:FILL=6801
9100 VISI=6810:DOT=6813
9110 RESET=6804:RESHD=6807
9120 AX=6747:AY=6750:AZ=6753
9130 BX=6756:BY=6759:BZ=6762
9140 PROJ=6765:PAR=1:PERSP=0
9150 VIEWX=AX:VIEWY=AY:VIEWZ=AZ
9160 GPOSX=6729:GPOSY=6732
9170 GPOSZ=6735:LOGO=6819
9200 RETURN
```

9. BEMUTATÓ PROGRAM

A program kazettán található DEMO nevű bemutató program hat különböző példa segítségével szemlélteti a griff-grafika képességeit:

POLYSPIRÁL
INSPIRÁL
KOCKÁK
SIERPINSKY görbe
KARFIFA (karfiolgörbe)
DUOPOLY

A griff-grafikával való ismerkedésnél a leírás átnézése után javasoljuk a DEMO program kipróbálását, majd a funkciók összevetését a leíráshoz mellékelt programlistával. A listából láthatjuk, hogy a DEMO program is tartalmazza a 9000 sorszámtól az előtét programot (lásd 8. fejezet).

A DEMO program betöltése

A DEMO program a kazettán a harmadik program (a GRAF és GR-FRONT-END után található). Betöltése előtt természetesen be kell tölteni a GRAF programot (lásd 7. fejezet).

Ezután álljunk a kazettán a DEMO program kezdete elé, adjuk ki a LOAD vagy LOAD "DEMO" parancsot és állítsuk lejátszásra a magnót. A betöltés után a program automatikusan elindul.

A DEMO program kezelése

A program kezelését két menü segíti. A főmenüben választhatunk a hat példa közül; a "színek" menü sor választásával áttérhetünk a szín választó menübe; a "vége" sor választásával befejezzük a DEMO program futtatását.

A szín választó menüben beállíthatjuk a számunkra legkellemesebb színekre a háttér, előtér, keret és a menü két színét. A színek beállítása után a "vége" sor választásával térhetünk vissza a főmenübe.

A főmenüben kiválasztott példa programok röviden ismertetik a bemutatott feladatot, funkciót. Sok esetben módunk van a feladatokat kipróbálni többféle paraméterrel is.

A példa programok leglényegesebb sorai a képernyőn is megjelennek. Ezeket érdemes részletesen tanulmányozni.

A példa programok kezelése nagyon egyszerű, a képernyőre kiírt utasítások szerint történik.

MELLÉKLETEK

DEMO program lista

```
1 POKE 2917,15:GOSUB 580
2 DIM DEMO$(10)
3 DEMO$(0)="polySpi"
4 DEMO$(1)="inspi"
5 DEMO$(2)="kockák"
6 DEMO$(3)="sierpin"
7 DEMO$(4)="karfifa"
8 DEMO$(5)="duopoly"
9 DEMO$(6)="szinek"
10 DEMO$(7)="vége"
11 CLS:PRINT" GRIFF-GRAFIKA BEMUTATÓ PROGRAM"
12 PRINT AT 3,1,"A griff-grafika képességeit többprogram szemlélteti, amelyeket
a"
13 PRINT "botkormány segítségével választ-hatunk ki és a RETURN billentyűleny
omásával indíthatunk el."
14 GOTO 18
15 REM *** főciklus (menü) ***
16 PX=0:PY=0:CLS
17 PRINT AT 2,6,"PROGRAM kiválasztás"
18 GOSUB 9000:X=USR(RESET)
20 X=USR(PGX,700)+USR(PGSY,620)
22 X=USR(FWARD,300)+USR(TURN,-90)
24 X=USR(FWARD,400)+USR(TURN,-90)
26 X=USR(FWARD,300)+USR(TURN,-90)
28 X=USR(FWARD,400)+USR(TURN,-90)
30 X=USR(PGX,750)+USR(PGSY,550)
32 SET INK 1:X=USR(FILL)
40 SET INK 3
42 FOR I=0 TO 7
44 PRINT AT I+10,24,DEMO$(I)
```

```

46 NEXT I
48 SET INK 2:PRINT AT 10,24,DEMO$(0)
60 A$=INKEY$:IF A$="" THEN GOTO 60
64 P=ORD(A$)
70 IF P=24 THEN PY=PX+1: GOTO 78
72 IF P=13 THEN GOTO 90
74 IF P=5 THEN PY=PX-1
76 IF PY=-1 THEN GOTO 60
78 IF PY=8 THEN GOTO 60
80 SET INK 3:PRINT AT PX+10,24 ,DEMO$(PX)
82 SET INK 2:PRINT AT PY+10,24,DEMO$(PY)
84 PX=PY
86 GOTO 60
90 ON PX+1 GOSUB 100,200,600,1200,850,1500,500,94
91 IF PX=6 THEN GOTO 60
92 GOTO 15
94 CL$:END

```

```

100 REM *****polyspi *****
101 GOSUB 9000:CLS
102 PRINT " POLYSPI(RAL) bemutató program"
103 PRINT AT 3,1,"Csökkendő hosszúságú éleket rajzolunk, egymáshoz képest adott
szöggel elfordítva. Próbáljunk különböző szögeket megadni 70 és185 között.(0
ill. RETURN esetén visszatérünk a menübe.)"
104 PRINT:PRINT
106 LIST 130-195
108 PRINT AT 22,1,"folytatáshoz nyomjon le egy bil-lentyűt"
110 X$=INKEY$:IF X$=""THEN GOTO 110
120 GOSUB 9000:CLS

```

```

130 INPUT PROMPT "szög ":SZÖG
135 CLS:PRINT "szög ";SZÖG
140 IF SZÖG=0 THEN RETURN
150 X=USR(RESET)+USR(POSX,260)+USR(POSY,215)
160 FOR I=460 TO 10 STEP -3
170 X=USR(FWARD,I)+USR(TURN,SZÖG)
180 NEXT I
190 GOTO 130

200 REM ***** inspi *****
201 GOSUB 9000:CLS
202 PRINT " INSPI(RAL) bemutató program"
203 PRINT AT 3,1,"Azonos éleket rajzolunk, egy- máshoz képest növekvő szöggel
    elfordítva. Próbáljunk különbözőkiinduló és növekmény-szögeket megadni. Néhá
ny ajánlott érték: (8,11),(2,6),(15,17),(44,33)."
```

204 PRINT"0 vagy RETURN esetén visszatérése menübe.

```

206 LIST 230-295
208 PRINT AT 22,1,"folytatáshoz nyomjon le egy bil-lentyőt"
210 X$=INKEY$:IF X$=""THEN GOTO 210
220 GOSUB 9000:CLS
230 PRINT"ind,szög"
235 INPUT IND,NÖV:CLS
240 IF IND=0 THEN RETURN
260 X=USR(RESET)+USR(POSX,300)+USR(POSY,500):J=IND
270 FOR I=1 TO 700
280 X=USR(FWARD,25)+USR(TURN,J):J=J+NÖV
290 NEXT I
295 GOTO 230

```

```

330 SIZE=400:GOSUB 9000:CLS
332 X=USR(RESET)+USR(POSX,400)+USR(POSY,300)
335 GOSUB 340:END
340 IF SIZE<40 THEN RETURN
350 SIZE=SIZE/2
360 GOSUB 340:X=USR(FWARD,SIZE*2)+USR(TURN,120)
370 GOSUB 340:X=USR(FWARD,SIZE*2)+USR(TURN,120)
380 GOSUB 340:X=USR(FWARD,SIZE*2)+USR(TURN,120)
395 SIZE=SIZE*2
400 RETURN
420 END
500 GOSUB 9000:X=USR(RESET)
501 VX=0:VY=0
502 X=USR(POSX,370)+USR(POSY,400)
504 X=USR(FWARD,300)+USR(TURN,-90)
506 X=USR(FWARD,350)+USR(TURN,-90)
508 X=USR(FWARD,300)+USR(TURN,-90)
510 X=USR(FWARD,350)+USR(TURN,-90)
512 X=USR(POSX,375)+USR(POSY,390)
514 SET INK 1:X=USR(FILL)
516 SET INK 3
518 FOR I=0 TO 6
520 PRINT AT I+16,14,SZ$(I)
522 NEXT I
524 SET INK 2:PRINT AT 16,14,SZ$(0)
526 A$=INKEY$:IF A$="" THEN GOTO 526
528 P=ORD(A$)
530 IF P=24 THEN VY=VX+1: GOTO 538
532 IF P=13 THEN GOTO 550
534 IF P=5 THEN VY=VX-1

```

```

536 IF VY=-1 THEN GOTO 526
538 IF VY=6 THEN GOTO 526
540 SET INK 3:PRINT AT VX+16,14 ,SZ$(VX)
542 SET INK 2:PRINT AT VY+16,14,SZ$(VY)
544 VX=VY
546 GOTO 526
550 IF VX=5 THEN 560
551 C(VX)=C(VX)+1
552 IF C(VX)=16 THEN C(VX)=0
554 SET PALETTE K(C(0)),K(C(1)),K(C(2)),K(C(3))
556 SET BORDER K(C(4))
558 GOTO 526

560 X=USR(PCSX,445)+USR(POSY,280)
561 PRINT AT 14,12,"
562 PRINT AT 15,12,"
563 PRINT AT 16,12,"
564 PRINT AT 17,12,"
565 PRINT AT 18,12,"
566 PRINT AT 19,12,"
567 PRINT AT 20,12,"
568 PRINT AT 21,12,"
569 PRINT AT 22,12,"
571 PRINT AT 23,12,"
572 RETURN
580 DIM C(4),K(15),SZ$(6)
582 DATA 0,1,4,5,16,17,20,21,64,65,68,69,80,81,84,85
583 FOR I=0 TO 15
584 READ K(I)
585 NEXT I

```

```

587 SZ$(0)="háttér"
588 SZ$(1)="menü 1"
589 SZ$(2)="előtér"
590 SZ$(3)="menü 2"
591 SZ$(4)="keret "
592 SZ$(5)="vége "
593 C(0)=9:C(1)=11:C(2)=14
594 C(3)=5:C(4)=5
596 SET PALETTE K(C(0)),K(C(1)),K(C(2)),K(C(3))
597 SET BORDER K(C(4))
598 RETURN

600 GOSUB 9000:CLS:L=200
610 X=USR(RESET)+USR(POSX,400)+USR(POSY,650)+USR(VIEWZ,800)+USR(TWIST,20):GRAPH
ICS 2:X=USR(VIEWZ,800)+USR(TILT,30):GOSUB 670
611 X=USR(RSID,760):GOSUB 670
612 X=USR(LSID,510):GOSUB 670
613 FOR I=1 TO 40
617 POKE 2893,0:GOSUB 670
618 X=USR(TILT,9)
619 POKE 2893,1:GOSUB 670
620 X=USR(LSID,300)+USR(RSID,300)
622 NEXT I
624 X=USR(RSID,250):GOSUB 670
629 FOR I=1 TO 600:NEXT I
630 GRAPHICS 4
631 GOSUB 9000:X=USR(RESET):L=115:CLS
632 PRINT "          KOCKA RAJZOLÁS"
633 PRINT "Kockák perspektivikus, majd pár-huzamos vetítését mutatjuk be, amel
yhez paramétereket lehet megadni, vagy meghagyhatjuk az eredeti paramétereke
t."

```

```

634 PRINT "A kiírt paramétereket a RETURN lenyomásával érvényesíthetjük. A q b
    st0 leütésére továbblép a program, egyéb karakterre új pa-paraméterekkel megismét
    elhetjük a programrészt."
635 LIST 650-666:PRINT AT 22,1,"folytatáshoz nyomjon le egy bil-lentűt"

636 H$=INKEY$:IF A$="" THEN 636
637 CLS:PRINT "Forgatási szögek      nézőpont"
638 PRINT "  0, 0, 20,      480, 500, 1500"
639 PRINT AT 1,1
640 INPUT PROMPT "":TI,TU,TW,VX,VY,VZ
641 X=USR(TURN,TU)+USR(TILT,TI)+USR(TWIST,TW)+USR(VIEWX,VX)+USR(VIEWY,VY)+USR(V
   IEWZ,VZ)
642 FOR Z=60 TO 950 STEP 200
643 FOR Y= 70 TO 800 STEP 210
644 S=USR(POSX,Z)+USR(POSY,Y)
645 GOSUB 650
646 NEXT Y:NEXT Z
647 A$=INKEY$:IF A$="" THEN 647
648 IF A$<>"q" THEN 637
649 GOTO 684
650 X=USR(FWARD,L)+USR(SINK,L)
652 X=USR(LSID ,L)+USR(LIFT,L)
654 X=USR(RSID,L)+USR(LSID,L)
656 X=USR(BACK,L)+USR(SINK,L)
658 X=USR(FWARD,L)+USR(BACK,L)
660 X=USR(RSID,L)+USR(FWARD,L)
662 X=USR(BACK ,L)+USR(LIFT,L)
664 X=USR(LSID ,L)+USR(RSID,L)
666 RETURN

```

```

670 X=USR(FWARD,L)+USR(LSID,L)
672 X=USR(RSID,L)+USR(LIFT,L)
674 X=USR(LSID,L)+USR(SINK,L)
676 X=USR(BACK,L)+USR(RSID,L)
678 X=USR(SINK,L)+USR(LSID,L)
680 X=USR(LIFT,L)+USR(RSID,L)
682 RETURN
684 CLS:PRINT "      párhuzamos vetítés":X=USR(RESET)
685 PRINT"      Forgatási szögek"
686 PRINT"      0, 0, 0"
687 PRINT AT 2,1
688 INPUT PROMPT"":T1,TU,TW
689 X=USR(TURN,TU)+USR(TILT,T1)+USR(TWIST,TW)
690 X=USR(AX,16384)+USR(AY,0)
691 X=USR(AZ,11583)+USR(BY,16384)
692 X=USR(BX,0)+USR(BZ,11583)
693 X=USR(PRIJ,1):L=100
694 FOR Z=140 TO 950 STEP 300
696 FOR Y=100 TO 750 STEP 290
698 S=USR(POSX,Z)+USR(POSY,Y)
700 GOSUB 650
710 NEXT Y:NEXT Z
712 A$=INKEY$:IF A$="" THEN 712
714 IF A$<>"q" THEN 684
716 CLS:PRINT "Egy másik kockarajzoló program:"
718 PRINT:LIST 730-790
719 PRINT AT 19,1,"ilyen kockát rajzoljunk egy      spirál mentén"
720 PRINT AT 22,1,"folytatáshoz nyomjon le egy billentyűt"
722 X$=INKEY$:IF X$="" THEN GOTO 722
724 GOTO 792

```

```

730 REM ** kocka síklapokból **
735 FOR I=1 TO 4
740 GOSUB 765
745 X=USR(FWARD,L)
750 X=USR(TILT,90)
755 NEXT I
760 RETURN
765 REM ** négyzet vonalakból *
770 FOR J=1 TO 4
775 X=USR(FWARD,L)
780 X=USR(TURN,90)
785 NEXT J
790 RETURN
792 X=USR(RESET)+USR(POSX,470)+USR(POSY,50)+USR(VIEWZ, 800)
793 L=85:GRAPHICS 2
794 FOR H=1 TO 40
796 GOSUB 730
798 X=USR(FWARD,L)+USR(TILT,9)+USR(TWIST,Z)
800 NEXT H
805 FOR H=1 TO 1000:NEXT H
810 GRAPHICS 4
820 GOTO 596

850 CLS:SET INK 2
852 PRINT " gömbre vetített karfiolgörbe":SET INK 1
855 X=USR(RESET)+USR(TURN,90)
860 X=USR(POSX,410)+USR(POSY,500)+USR(TILT,35)
865 L=75:E=0.8:F=1/E:A=45:B=-2*A:V=15
870 FOR I=1 TO 2
875 GOSUB 1000

```

```

880 X=USR(TURN,-180)
885 NEXT I
890 X=USR(POSX,435)+USR(POSY,510)
895 SET INK 2:X=USR(FILL)
898 FOR H=1 TO 1500:NEXT H
900 RETURN
1000 IF L<15 THEN 1080
1010 X=USR(TURN,A)+USR(TILT,4)+USR(FWARD,L)
1020 L=L*E:GOSUB 1000:L=L*F
1030 X=USR(BACK,L)+USR(TILT,-4)+USR(TURN,B)+USR(TILT,4)
1040 X=USR(FWARD,L)
1050 L=L*E:GOSUB 1000:L=L*F
1060 X=USR(BACK,L)+USR(TILT,-4)+USR(TURN,A)
1070 RETURN
1080 IF V=0 THEN RETURN
1090 X=USR(TURN,-90)+USR(FWARD,V)+USR(TURN,90)
1100 RETURN

1200 GOSUB 9000:CLS:SET INK 2
1202 PRINT AT 2,1,"Az itt következő rajz a
- szerő, rekurzív görbe."
1203 PRINT" PROGRAMJA:";PRINT
1205 LIST 1300-1400
1206 PRINT AT 22,1,"folytatáshoz nyomjon le egy bil-lentyőt"
1207 X$=INKEY$:IF X$=""THEN GOTO 1207
1208 CLS:PRINT " SIERPINSKY görbe elforgatva"
1209 PRINT" (kis fókusszal)"
1210 X=USR(RESET)
1220 X=USR(POSX,100)+USR(POSY,880)+USR(TILT,20)+USR(VIEWZ,300)
1230 L=4:D=22:S=ZZ

```

Sierpinsky görbe, mely egy egy

```

1240 FOR F=1 TO 4
1250 GOSUB 1300
1260 X=USR(TURN,-45)+USR(FWARD,D)+USR(TURN,-45)
1270 NEXT F
1275 FOR H=1 TO 1000:NEXT H
1280 X=USR(POSX,260)+USR(POSY,585)
1282 SET INK 1:X=USR(FILL)
1285 FOR H=1 TO 1000:NEXT H
1288 SET INK 3:X=USR(FILL)
1289 FOR H=1 TO 300:NEXT H
1291 SET INK 0:X=USR(FILL)
1292 FOR H=1 TO 300:NEXT H
1293 SET INK 2:X=USR(FILL)
1295 FOR H=1 TO 1100:NEXT H
1299 RETURN
1300 IF L=0 THEN RETURN
1310 L=L-1
1320 GOSUB 1300:X=USR(TURN,-45)
1330 X=USR(FWARD,D)+USR(TURN,-45)
1340 GOSUB 1300:X=USR(TURN,90)
1350 X=USR(FWARD,3)+USR(TURN,90)
1360 GOSUB 1300:X=USR(TURN,-45)
1370 X=USR(FWARD,D)+USR(TURN,-45)
1380 GOSUB 1300
1390 L=L+1
1400 RETURN

```

```

1500 REM *** multipoly ***
1510 CLS:PRINT "    DUOPOLY"
1520 PRINT:PRINT"A teknős grafika irodalmából    ismert DUOPOLY egyszerű vektor
    technikát használva érdekes szimmetrikus görbéket rajzol. Kísér-letezzünk külön
    böz értékek meg-adásával.
1525 PRINT"A paraméterek: El1, Szög1, El2, Szög2, Posx, Posy (kiinduló po- zíció
    ), Ism (ismétlési szám)."
```

Line	Code
1526	PRINT
1528	PRINT "Ajánlott értékek:"
1530	PRINT "(150,90,150,320,300,600, 60)"
1535	PRINT "(40,10, 40,-15,400,500, 90)"
1538	PRINT "(200,62,200,300,300,500, 90)"
1540	PRINT "(200,90,200,300,400,500, 30)"
1542	PRINT "(4, 2, 8, -2,500,500,200)"
1544	PRINT "(60,19, 60,-20,500,500,380)"
1546	PRINT "(100,45, 40, 9,500,100, 45)"
1548	PRINT "(35,32, 20, 4,500,100,100)"
1550	PRINT AT 22,1,"Folytatáshoz nyomjon le egy bil-lentyőt!"
1555	AS=INKEY\$:IF AS="" THEN 1555
1558	IF AS="q" THEN RETURN
1560	CLS:PRINT"A kiírt paraméterek módosíthatókmajd a RETURN leütésével érvé-n
1562	PRINT"El1,Szög1,El2,Szög2,Posx,Posy,Ism"
1565	PRINT " 100, 45, 40, 9, 500,100, 45
1567	PRINT:PRINT"Rajzolás után kilépés e program-részből a q billentyű leütéséve
1568	PRINT"Más karakterre e programrész is-métlődik."
1570	PRINT AT 5,1
1575	INPUT PROMPT "":S1,A1,S2,A2,PX,PY,ISM
1580	GOSUB 9000:CLS:X=USR(RESET)+USR(POSX,PX)+USR(POSY,PY)

1610 FOR R=0 TO ISM
1615 D=R*A1: IF D>30000 THEN D=D-28800
1620 X=USR(RESHD)+USR(TURN,D)+USR(FWARD,S1)
1625 D=R*A2: IF D>30000 THEN D=D-28800
1630 X=USR(RESHD)+USR(TURN,D)+USR(FWARD,S2)
1640 NEXT R
2000 GOTO1555

9000 TURN=6792:TILT=6795
9010 TWIST=6798:PEN=6768
9020 UP=1:DOWN=0
9030 FWARD=6771:BACK=6774
9040 LSID=6777:RSID=6780
9050 LIFT=6783:SINK=6786
9060 POSX=6720:MOVX=6738
9070 POSY=6723:MOVY=6741
9080 POSZ=6726:MOVZ=6744
9090 MOVE=6789:FILL=6801
9100 VISI=6810:DQT=6813
9110 RESET=6804:RESHD=6807
9120 AX=6747:AY=6750:AZ=6753
9130 BX=6756:BY=6759:BZ=6762
9140 PROJ=6765:PAK=1:PERSP=0
9150 VIEWX=AX:VIEWY=AY:VIEWZ=AZ
9160 GPUSX=6729:GPUSY=6732
9170 GPUSZ=6735:LOGO=6819
9200 RETURN

GRAFIKAI FÜGGVÉNYEK ÖSSZEFOGLALÓ TÁBLÁZATA

Név	Funkció	Hívás előttét program használatával	Paraméter
FWARD	mozgatás előre	USR(FWARD,L)	L=hossz
BACK	mozgatás hátra	USR(BACK,L)	L=hossz
LIFT	felemelés	USR(LIFT,L)	L=hossz
SINK	lesüllyesztés	USR(SINK,L)	L=hossz
LSID	mozgatás balra	USR(LSID,L)	L=hossz
RSID	mozgatás jobbra	USR(RSID,L)	L=hossz
TURN	forgatás Z tengely körül	USR(TURN,A)	A=szög
TWIST	forgatás X tengely körül	USR(TWIST,A)	A=szög
TILT	forgatás Y tengely körül	USR(TILT,A)	A=szög
POSX	X koordináta beállítása	USR(POSX,X)	X=koord.
POSY	Y koordináta beállítása	USR(POSY,Y)	Y=koord.
POSZ	Z koordináta beállítása	USR(POSZ,Z)	Z=koord.
MOVX	X koord. MOVE parancshoz	USR(MOVX,X)	X=koord.
MOVY	Y koord. MOVE parancshoz	USR(MOVY,Y)	Y=koord.
MOVZ	Z koord. MOVE parancshoz	USR(MOVZ,Z)	Z=koord.
MOVE	mozgatás (MOVX,MOVY,MOVZ) pontba	USR(MOVE)	
GP0SX	X koord. lekérdezése	USR(GP0SX)	
GP0SY	Y koord. lekérdezése	USR(GP0SY)	
GP0SZ	Z koord. lekérdezése	USR(GP0SZ)	

PEN	toll letevése felemelése	USR(PEN,DOWN) USR(PEN,UP) USR(FILL) USR(VISI)	USR(6768,0) USR(6768,1) USR(6801) USR(6810)
FILL	szinezés		
VISI	láthatóság vizsgálata		
RESET	változók alaphelyzetbe állítása	USR(RESET)	USR(6804)
RESHD	irányítás alaphelyzetbe állítása	USR(RESHD)	USR(6807)
DOT	pont rajzolása	USR(DOT)	USR(6813)
LOGO	bejelentkező grafika rajzolása	USR(LOGO)	USR(6819)
PROJ	vetítés perspektívikus párhuzamos	USR(PROJ,Persp) USR(PROJ,PAR)	USR(6765,0) USR(6765,1)
AX	párhuzamos vetítés ax paramétere	USR(AX,ax)	USR(6747,ax) ax=par.
AY	párhuzamos vetítés ay paramétere	USR(AY,ay)	USR(6750,ay) ay=par.
AZ	párhuzamos vetítés az paramétere	USR(AZ,az)	USR(6753,az) az=par.
BX	párhuzamos vetítés bx paramétere	USR(BX,bx)	USR(6756,bx) bx=par.
BY	párhuzamos vetítés by paramétere	USR(BY,by)	USR(6759,by) by=par.
BZ	párhuzamos vetítés bz paramétere	USR(BZ,bz)	USR(6762,bz) bz=par.
VIEWX	persp. vetítés vx paramétere	USR(VIEWX,vx)	USR(6747,vx) vx=par.
VIEWY	persp. vetítés vy paramétere	USR(VIEWY,vy)	USR(6750,vy) vy=par.
VIEWZ	persp. vetítés vz paramétere	USR(VIEWZ,vz)	USR(6753,vz) vz=par.

VIDEOTON

ELEKTRONIKAI VÁLLALAT
SZÁMÍTÁSTECHNIKAI GYÁRA