

Az Országos Commodore Egyesület lapja

újság

1989/12

KELLEMEK ÜNNEPEKET  
ÉS BOLDOG ÚJ ÉVET  
KÍVÁNUNK  
MINDEN KEDVES TAGTÁRSUNKNAK  
ÉS OLVASÓNKNNAK!

TIPPEK – TRÜKKÖK  
KÖNYVEK  
ESPERATO  
EGÉRANIMÁCIÓ  
JÁTÉKPIACI  
ÚJDONSÁGOK



# ISMERED MÁR?



## A LEGJOBB SEGÉDESZKÖZ A TANULÁSHOZ



## MIT, HOGYAN, HOL, MIKOR?

**EGYESÜLETI ÜGYEK:** Egyesületünk tagja lehet mindenki, aki a tagsági díjat befizeti. A tagdíjat személyesen az Egyesület irodájában (1133 Budapest, Kárpát u. 7/a. I. em. 11., tel: 497-559), vagy átutalással az MNB 217-98292, OTP 565-3610-8 számlára lehet befizetni. Megrendelés esetén számlát küldünk.

Minden tagunk rendelkezésére áll PÖTYÖGŐSZOLGÁLTUNK, a szervizkedvezmény és az apróhirdetési lehetőség.

A Deákpáholy tagjai minden hónapban megkapják a C-újságot, a tagsági díj fél évre 366 forint.

A Pluszpáholy tagjai minden hónapban megkapják a C-újságot, és minden hónapban 2x60 forint vásárlási utalványt is kapnak. A tagsági díj fél évre 911 forint.

A Szuperpáholy tagjai havonta 15 példányt kapnak a C-újságból, és ezzel havonta 1800 forint vásárlási utalványt. A féléves tagsági díj 10 000 forint.

**ÜGYFÉLFOGADÁS:** minden kedden és csütörtökön 14–18 óra között várjuk tagjainkat és az érdeklődőket.

**PÖTYÖGŐSZOLGÁLTAT:** Az újságban megjelenő programokat másolja a megrendelők részére. Megrendelhető személyesen az Egyesület irodájában, vagy postai utánvétellel.

1989 novemberétől a pötyögőszolgálat időpontja az alábbiak szerint változik: minden harmadik héten kedden és csütörtökön 15 órától 20 óráig az Egyesület irodájában (1133 Kárpát utca 7/A).

1989. év végéig a pötyögőszolgálat időpontjai: november 7., november 9., november 28., november 30., december 19., december 21.

**APRÓHIRDETÉS:** Az egyesületi tagoknak ingyen áll rendelkezésére. Nem tagoknak a hirdetés ára 80 forint. A hirdetés módja: az újságban megjelenő nyomtatvány kitöltésével.

A C-újság régebbi számai megvásárolhatók az Egyesület irodájában, vagy megrendelhetők utánvétellel.

Kedvezményes ár! Tagoknak olcsóbb!

Az újságban eddig megjelent programok gépenként összegyűjtve megrendelhetők. VC 20, C16, PLUS/4, C128, C64. További felvilágosítást is adunk a 497-559-es telefonszámon, vagy levélben!

Vidéki Pluszpáholy-tagjaink három havi tikkett összegyűjtésekor igénybe vehetik a NOVOTRADE 2C áruház ingyenes csomagküldő szolgálatát.

Vidéken további információk kaphatók:

Jászberényi Városi Könyvtár

Győri Bartók Béla Művelődési Ház

Zalaegerszegi Ságvári Endre Gimnázium

Pécsi Apáczai Csere János Gimnázium

**Budapesten havonta klubdélelőtt  
a Petőfi Csarnokban.**

Az Országos Commodore Egyesület módszertani kiadványa.

Egyesületi iroda és szerkesztőség: 1133 Budapest, Kárpát u. 7/a I. em. 11. Tel.: 1497-559

Felelős kiadó: Horváth Judit, az egyesület elnöke

Főszerkesztő: Rados Péter, az OCE főtítkára

Felelős szerkesztő: Dr. Horváth András

Művészeti szerkesztő: Szulyovszky József

Egyesületi szervező: Winter Júlia

Lapmenedzser: Kovács Gábor

Levél cím: Commodore Ujság, 1388 Bp. 62. Pf. 86.

Index: ISSN 0237-756 X

Terjeszti a Magyar Posta.

Megvásárolható a hírlapárusoknál.

MSZH Nyomda

## Tisztelt Szerkesztőség!

Sokak rettennek vissza az AMIGA gépi kódú programozástól, mert azt hiszik, hogy ez rendkívül bonyolult lehet. Szeretném most röviden ismertetni az egyik legelterjedtebb assembler kezelését, és egy egyszerű példaprogramot mutatni azoknak, akik már egy kicsit járatosak a gépi kódban (65xx, Z80), de nem mertek, vagy nem tudtak nekivágni az AMIGA assembler szintű programozásának. A legelterjedtebb assemblerek egyike a SEKA assembler. Nem igazán azért terjedt el, mert nagyon jó, hanem mert nagyon rövid, és a SOUNDTRACKER programlemezeken legtöbbször megtalálható. Így esetleg olyanoknak is megvan, akik ezt eddig nem is sejtették. Érdemes tehát alaposabban körülnézni a SOUNDTRACKER lemezen, hátha rajta van a SEKA assembler is!

Használatát tekintve, tartalmaz egy egyszerű forrásszöveg-szerkesztőt, és egy beépített monitort is. A szerkesztő meglehetősen kezdetleges, nagyon keveset tud. Nem lehet benne sorokat törölni vagy másolni, csak a szöveget írhatjuk be. Szerencsére elfogadja a más szerkesztőkkel készített hle-okat is, Sajnos, nem lehet benne használni ún. INCLUDE file-okat, és a lefordítandó program nagyságának határt szab a rendelkezésre álló memória, ugyanis a teljes fordítást a RAM-ban végzi el. Természetesen ez nem mindig elegendő, mivel nem képes a lemezegységet felhasználni a fordításhoz. (Bizonyos assemblerek a lemezen készítik el a fordítást, így a forrásszöveg igen nagy lehet, a lefordított, futáskész program a lemezen jelenik meg).

**Egyesületünk és az újság szerkesztősége várja az Önök programjait, hardver leírásait, cikkeit. Szerkesszük együtt az újságot!**

**FIGYELEM!**

Kérjük az 1990 évi tagdíjakat befizetni!

**A kedvezmények a következő vidéki könyvesboltok 2C sarkaiban válthatók be.**

**PÉCS:** Zrínyi Miklós Könyvesbolt. 7621 Jókai u. 25. Tel.: 72-12835

**DEBRECEN:** Szak- és ismeretterjesztő Könyváruház. 4024 Hunyadi u. 8. Tel.: 52-23237

**SZOMBATHELY:** Savaria Könyvesbolt. 9700 Mártírok tere 1. Tel.: 94-12341

**VESZPRÉM:** Kölcsey Ferenc Könyvesbolt. 8200 Cserhát út 7.

**BÉKÉSCSABA:** Radnóti M. Könyvesbolt. 5600 Tanácsköztársaság út 2. Tel.: 25-207

**GYŐR:** Pattantyús Á. Géza Szakkönyvesbolt. 9021 Molnár Ferenc u. 9.

**SZEGED:** Tömörkény Könyvesbolt. 6720 Lenin krt. 48. Tel.: 62-21453

**SZOLNOK:** Szigligeti Könyvesbolt. 5000 Ságvári krt. 35. Tel.: 56-11133

**MISKOLC:** Chip-kuckó. 3530 Tanácsház tér 14.

Az első listán az assembler parancskészlete található.

A második listán egy egyszerűbb programocská van, ami a képernyőre ír ki egy üdvözlő szöveget.

Az assembler listán láthatjuk, hogy az igencsak eltér az általunk eddig ismert assembler listáktól. A ROM-ban (KickStart-ban) található LIBRARY-k tartalmazzák a különféle belépési címeket. Ezek a belépési címek más-más KickStart verzióknál eltérők lehetnek, de a LIBRARY-ból mindig az aktuális KickStart belépési címek töltődnek be. Nem kell tehát megjegyeznünk egy hatalmas hexaszámot (mondjuk \$ fffffd0), hanem elegendő megnyitni a szükséges LIBRARY-t. Ezek a LIBRARY-k a ROM-ban találhatóak. (Csak néhány rövid, ritkán használatos LIBRARY van lemezen. Pl.: DISKFONT, LIBRARY). Ha a szükséges LIBRARY-t megnyitottuk, a DO regiszterben visszakapjuk az ún. báziscímet. A megfelelő ROM rutin hívásakor a báziscímnek az A6-os regiszterben kell lennie. Egy ROM hívás valahogy így néz ki:

JSR Write (A6); azaz JumpSubRutin "funkció" + (A6)

Az A6-os regiszter tartalma + "funkció" értéke egy címre mutat, ahol a ROM belépési cím található, a szubrutinunk az itt található értékre ugrik. Lehet, hogy első hallásra egy kicsit bonyolultnak tűnik, de a későbbiek során ez a módszer nagyon kényelmesnek bizonyult számomra. Az AMIGA gépi kódú programozásával foglalkozó könyvekben és újságokban kivétel nélkül megtalálható a teljes LIBRARY lista, a különféle értékekkel. Az assemblerek többsége ezeket az ún. INCLUDE file-okat tölti be, tehát nekünk csak a funkció nevét kell ismerni, a funkcióhoz tartozó értéket már nem nekünk kell beállítani. Sajnos, a SEKA Assembler ezt nem tudja, itt a megfelelő számokat nekünk kell beírni. Természetesen ezek az értékek az összes KickStart verzióknál azonosak, csak a báziscímen található érték változik.

**Figyelem!** A többi (Z80,65xx) assemblertől eltérően az adatok mozgatása fordítva történik. A MOVE .b # 10,D0 utasítás egy byte hosszan betölti a 10-et a D0 regiszterbe.

Egy ilyen rövid kis cikk nem elegendő, hogy tökéletesen megtanítsa mindenkit a 68000-es assembler minden utasításának használatára. Akit a dolog bővebben érdekel, olvassa el a NOVOTRADE — DATAECKER könyvet: A 68000-es mikroprocesszor programozása, továbbá tanulmányozza az AMIGA újságokban megjelent gépi kódú programozásról szóló írásokat.

Malmarits Tamás

A SEKA assembler:

- T x.....A sormutatót az x-edik sorra állítja.
- P x.....x sort listáz a képernyőre a sormutatótól kezdődően.
- Px...+(CTRL P)..Így a nyomtatóra megy a lista!
- I.....Beszúr egy sort a sormutató után, és a sormutatót eggyel növeli.
- U x .....A sormutatót x sorral visszaállítja.
- D x .....A sormutatót x sorral előre állítja.
- Z x.....A sormutatótól kezdve x sort töröl.
- M.....Monitor (A már lefordított programhoz).
- Q.....Monitor + ASCH karakterek.
- N.....Dissassembler.
- H.....Kiírja a rendelkezésre álló memória nagyságát. (Editor, puffer, lefordított program stb.)
- V.....Az aktuális alkönyvtár tartalmának kiírása a képernyőre.
- !.....Kilépés a SEKA assemblerből.
- A.....Assembler.  
 OPTIONS/: ENTER esetén normál assembler, hiba esetén leáll.  
 V esetén a képernyőre listáz.  
 P esetén a nyomtatóra listáz.
- R.....Betölt egy forrásfile-ot az editorban.
- RI.....Betölt egy adatfile-ot a lefordított programba a kívánt helyre, a megadott hosszszon. Ha az END kérdésre csak ENTER-t válaszolunk, akkor az adatfile-ot teljes egészében betölti.  
 Filename>: Az adathle neve.  
 Begin>: Hova szúrja be?  
 End>: Milyen hosszban?
- W.....Az editorban lévő forrásfile kimentése az aktuális alkönyvtárba.
- WO.....A lefordított program kimentése az aktuális alkönyvtárba.
- WI.....IMAGE típusú hle kimentése.
- G.....A lefordított program indítása. Lehetőség van megszakításpontok (break point) beállítására is.
- KS.....Törli az editor tartalmát.
- O.....Visszaállítja az editor tartalmát (ha azóta azt nem piszkáltuk)!

```

*****
#   Assembler példaprogram 1.   #
#                               #
#   Commodore Újság számára     #
#                               #
#                               #
*****

```

```

ExecBase:   equ   4
Length:     equ   13
Write:      equ  -48
OutPut:     equ  -60
OpenLib:    equ  -552
CloseLib:   equ  -414

```

```

move.l     ExecBase,a6           ; Exec báziscím A6-ba
lea        dosname,a1           ; megnyitandó library nevének kezdőcíme az A1-be
moveq     #0,d0
jsr        OpenLib(a6)          ; library megnyitása
tst.l     d0                    ; volt hiba?
beq       vege                  ; ha igen, akkor ugrás a VÉGÉRE

```







```

175 PRINT " MI LA NOVAJN          LERNOLIBROJN. "
198 GETA#:IFA#<>"|"THEN198
199 GOTO41
200 REM      *** A MASODIK LECKE ***
201 POKE53280,0:POKE53281,0:PRINTCHR$(14);"32.LECKE3          301.LAP# 131"
202 PRINT " *Z ELSD RESZBEN MAR MEGISMERTUK AZ          ESZPERANTO NYELVTANANAK NEHAN
Y";
203 PRINT " SAJATOS-SAGAT.\OST EGY MONDAT SEGITSEGEVEL SZIN-TE MINDEN FONTOS ";
204 PRINT"NYELVTANI SZABALYT TA-NULMANYOZHATUNK:31"
205 PRINT" HODIA32 MI EKLEGIS LA LIBRETON DE TETRO,EN KIU ESTAS TRE ";
206 PRINT"INTERESAJ ANEKDOTOJ.31"
207 PRINT" LOSZOR IS NEZZUK MEG, MIT JELENTENEK AMONDATBAN SZEREPLD, MEG NEM";
208 PRINT" ISMERT ESZ-PERANTO SZAVAK:"
209 PRINT" HODIA32 - MA":PRINT" LEGI - OLVASNI":PRINT" LIBRO - KONYV"
210 PRINT" TETRO - TETER":PRINT" KIU - KI,AKI,AMELY":PRINT" ESTI - VAN (LETIGE)
211 PRINT" TRE - MAGYON":PRINT" INTERESA - ERDEKES":PRINT" ANEKDOTO - ANEKDOTA,
ICC"
212 GETA#:IFA#<>"|"THEN212
213 PRINT"32.LECKE3          302.LAP# 131"
214 PRINT" IA EMLEKEZNEK , MAR TANULTUNK A JELEN- ES A JOVO IDEJU ";
215 PRINT"IGERAGDZASROL(-AS A JELEN IDO,-OS A JOVO IDO JELE ),DE A ";
216 PRINT"MULT IDEJURD MEG NEM BESZELTUNK."
217 PRINT" * MULT IDEJU IGE VEGZODESE: -IS31"
218 PRINT"MI LEGIS - OLVASTAM":PRINT"VI LEGIS - OLVASTAL = MAGA/ON OLVASOTT"
219 PRINT"301/3031 LEGIS - OLVASOTT":PRINT"MI LEGIS - OLVASTUNK"
220 PRINT"VI LEGIS - OLVASTATOK = MAGUK OLVASTAK":PRINT"ILI LEGIS - OLVASTAK31"
221 PRINT"\GY MOST MAR ISMERJUK MINDHAROM IGEIDOT,ES HASZNALTUNK MAR EGY ";
222 PRINT" ELOLJAROSZOT IS(AZ ELOZO LECKEBEN TANULT 'DE' ELOLJARO-SZO A ";
223 PRINT"BIRTOKVISZONY JELOLESERE SZOLGAL).ZEK UTAN ISMERKEDJUNK MEG MOST ";
224 PRINT" AZ ESZ-PERANTO NYELV KEPZORENDSZEREVEL."
225 GETA#:IFA#<>"|"THEN225
226 PRINT"32.LECKE3          303.LAP# 131"
227 PRINT" * * * * * "
228 PRINT" *AMENHOFROL, AZ ESZPERANTO NYELV ALAP-JAINAK MASTEREMTOJERD AZT ";
229 PRINT"JEGYEZTEK FELKORTARSAI, HOGY GYORSAN TANULT NYELVEKET";
230 PRINT"(NEGY NYELVET ANYANYELVI, TOVABBI NEGYETTARSALGASI FOKON BESZELT).";
231 PRINT"*AMENHOF MEGISSOKAT TOPRENGETT, HOGYAN CSOKKENTHETNEMINIMALISRA ";
232 PRINT"A SZOTANULAS FARADSAGAT. TOPRENGES EREDMENYE AZ ESZPERANTO ";
233 PRINT"NYELVZSENIALIS SZOKEPZESI RENDSZERE. *LAPJATA NEMZETI NYELVEKBEN ";
234 PRINT"IS MEGLEVO, DE OTTNEM KOVETKEZETESEN HASZNALT ELO- ES UTO-KEPZOK ";
235 PRINT"ALKOTJAK. *Z ELO- ES UTOKEPZOKMAXIMALIS HASZNALATAVAL EGY-EGY ";
236 PRINT"SZOGYOK-BOL (SZOTOBOL) ATLAGOSAN TIZ SZOT KEPEZ-HETUNK, IGY A ";
237 PRINT"SZOTANULASRA CSAK TIZEDNYIIDOT KELL FORDITANUNK MAS NYELVEKHEZ VI-";
238 PRINT"SZONYITVA.":PRINT" ZUTAN TANULJUK MEG A GYAKORLATBAN AZ";
239 PRINT"ELO- ES UTOKEPZOK HASZNALATANAK MODJAT!";
240 GETA#:IFA#<>"|"THEN240
241 PRINT"32.LECKE3          304.LAP# 131":PRINT" /EHANY KEPZO:"
242 PRINT"LOKEPZOK:" :PRINT" / EK- / = KEZDEST JELOL"
243 PRINT" L. : MI LEGIS - OLVASTAM"
244 PRINT" MI EKLEGIS - ELKEZDTEM OLVASNI"
245 PRINT" / MAL- / = A SZOGYOK ELLENTETENEK. KIFEJEZESERE SZ
LGAL
246 PRINT" L. : SANA - EGESZSEGES"
247 PRINT" MALSANA - BETEG":PRINT" /TOKEPZOK:"
248 PRINT" / -ET- / = KICSINYITOKEPZO":PRINT" L. : LIBRO - KONYV"
249 PRINT" LIBRETO - KONYVECSKE":PRINT" / -IL- / = ESZKOZKEPZO"
250 PRINT" L. : TRAN301 - VAGNI":PRINT" TRAN301ILO - KES"
251 PRINT" *Z ESZPERANTOBAN A MAGYAR NYELV RAGJAITELDLJAROKKAL FEJEZZUK KI. L.
252 PRINT" 'EN' = -BAN, -BENI EN LA LIBRO-A KONYVBEN
253 PRINT" 'AL' = -NAK, -NEKI AL LA KNABO-A FIUNAK
254 PRINT"* MOST TANULT NYELVTANI SZABALYOK SEGIT-SEGEVEL ";
255 PRINT"FORDITSA LE A FELDAMONDATOT!"
298 GETA#:IFA#<>"|"THEN298
299 GOTO41
300 REM      *** A HARMADIK LECKE ***
301 POKE53280,0:POKE53281,0:PRINTCHR$(14);"33.LECKE3          301.LAP# 131"
302 PRINT" *Z ESZPERANTOBAN AKAR 5 PERC ALATT MEG LEHET TANULNI 1-1000-IG "
303 PRINT"SZAMOLNI.TH-HEZ UGYANIS CSUPAN 12 (S RARDASUL NEM ISTELJESEN ";
304 PRINT"ISMRETLEN) SZOT KELL MEGTANULNI*Z ESZPERANTO TOSZAMNEVEK:31"
305 PRINT" 1 - UNU          2 - DU"

```



# Az AMIGA lemezei (folytatás)

A cikksorozat első részében a legalapvetőbb ismereteket közöltem az Amiga lemezekről. Mostantól kezdve részletesen fogunk foglalkozni az egyes szektorok típusával és azok felépítésével. Természetesen szükségünk lesz egy ún. Disk-monitor programra, amelynek segítségével a lemez egyes részeihez hozzá tudunk férni. Erre a célra használhatunk egy egyszerű monitorprogramot is, de választhatunk feladatorientáltabb programok közül is, pl. Project-D, Mirror Hacker Package, Disk-Craft, SmartDisk... stb. (A sorozat listáit egyébként a SmartDisk nevű programmal készítettem.) Az előző részben már szó esett arról, hogy a lemezen összesen 1760 (oldalanként 880-880) szektor (blokk) található. Számozásuk az ún. logikai sorszámmal történik, ami 0—1759 intervallumba esik. Ezután a blokkokat a logikai sorszámmal ellátott egységek (512 byte-os lemezrészecskék), a szektorokat pedig a sávon belüli egységek (512 byte-os lemezrészecskék) megnevezésére fogjuk használni. A lemez meghajtó a többi 2-oldalas meghajtóhoz hasonlóan a lemez alsó oldalához (BOTTOM oldal) rendeli hozzá a 0-ás olvasó-írófejet (HEAD 0) és a felső oldalhoz (TOP oldal) az 1-es olvasó-írófejet (HEAD 1). A lemezekben a blokkok logikai számozása a következő:

— 0. blokk — 0. oldal (BOTTOM) 0. sáv 0. szektor

— 1. blokk — 0. oldal (BOTTOM) 0. sáv 1. szektor

— 10. blokk — 0. oldal (BOTTOM) 0. sáv 10. szektor

— 11. blokk — 1. oldal (TOP) 0. sáv 0. szektor

— 1758. blokk — 1. oldal (TOP) 79. sáv 9. szektor

— 1759. blokk — 1. oldal (TOP) 79. sáv 10. szektor

Általánosan igaz a következő (blokkok, oldalak, sávok, szektorok között fennálló) összefüggés:

$LBN = TN * 22 + DS * 11 + SPN$ , ahol LBN jelöli a blokk logikai sorszáma (Logical Block Number — értéke 0—1759 lehet), TN jelöli az illető sáv számát (Track Number — értéke 0—79 lehet), DS jelöli a disk oldalát (Disk Side — értéke a BOTTOM oldal esetén 0, a TOP oldal esetében pedig 1), SPN pedig az adott szektor DS által meghatározott lemezoldalának megfelelő sávján belüli fizikai sorszáma jelöli (Sector Physical Number — értéke 0—11 lehet). Ha történetesen a 0. oldal 40. sáv 0. szek-

TRACK = 00	SECTOR = 00	HEAD = 0	BLOCK = 0000	Checksum = 444F530
Diskname - FREE				
000:	444F 5300	444F 5301	444F 5302	444F 5303
010:	444F 5304	444F 5305	444F 5306	444F 5307
020:	444F 5308	444F 5309	444F 530A	444F 530B
030:	444F 530C	444F 530D	444F 530E	444F 530F
040:	444F 5310	444F 5311	444F 5312	444F 5313
050:	444F 5314	444F 5315	444F 5316	444F 5317
060:	444F 5318	444F 5319	444F 531A	444F 531B
070:	444F 531C	444F 531D	444F 531E	444F 531F
080:	444F 5320	444F 5321	444F 5322	444F 5323
090:	444F 5324	444F 5325	444F 5326	444F 5327
0A0:	444F 5328	444F 5329	444F 532A	444F 532B
0B0:	444F 532C	444F 532D	444F 532E	444F 532F
0C0:	444F 5330	444F 5331	444F 5332	444F 5333
0D0:	444F 5334	444F 5335	444F 5336	444F 5337
0E0:	444F 5338	444F 5339	444F 533A	444F 533B
0F0:	444F 533C	444F 533D	444F 533E	444F 533F
100:	444F 5340	444F 5341	444F 5342	444F 5343
110:	444F 5344	444F 5345	444F 5346	444F 5347
120:	444F 5348	444F 5349	444F 534A	444F 534B
130:	444F 534C	444F 534D	444F 534E	444F 534F
140:	444F 5350	444F 5351	444F 5352	444F 5353
150:	444F 5354	444F 5355	444F 5356	444F 5357
160:	444F 5358	444F 5359	444F 535A	444F 535B
170:	444F 535C	444F 535D	444F 535E	444F 535F
180:	444F 5360	444F 5361	444F 5362	444F 5363
190:	444F 5364	444F 5365	444F 5366	444F 5367
1A0:	444F 5368	444F 5369	444F 536A	444F 536B
1B0:	444F 536C	444F 536D	444F 536E	444F 536F
1C0:	444F 5370	444F 5371	444F 5372	444F 5373
1D0:	444F 5374	444F 5375	444F 5376	444F 5377
1E0:	444F 5378	444F 5379	444F 537A	444F 537B
1F0:	444F 537C	444F 537D	444F 537E	444F 537F
Sector type - Unknown				
000:	444F 5380	444F 5381	444F 5382	444F 5383
010:	444F 5384	444F 5385	444F 5386	444F 5387
020:	444F 5388	444F 5389	444F 538A	444F 538B
030:	444F 538C	444F 538D	444F 538E	444F 538F
040:	444F 5390	444F 5391	444F 5392	444F 5393
050:	444F 5394	444F 5395	444F 5396	444F 5397
060:	444F 5398	444F 5399	444F 539A	444F 539B
070:	444F 539C	444F 539D	444F 539E	444F 539F
080:	444F 53A0	444F 53A1	444F 53A2	444F 53A3
090:	444F 53A4	444F 53A5	444F 53A6	444F 53A7
0A0:	444F 53A8	444F 53A9	444F 53AA	444F 53AB
0B0:	444F 53AC	444F 53AD	444F 53AE	444F 53AF
0C0:	444F 53B0	444F 53B1	444F 53B2	444F 53B3
0D0:	444F 53B4	444F 53B5	444F 53B6	444F 53B7
0E0:	444F 53B8	444F 53B9	444F 53BA	444F 53BB
0F0:	444F 53BC	444F 53BD	444F 53BE	444F 53BF
100:	444F 53C0	444F 53C1	444F 53C2	444F 53C3
110:	444F 53C4	444F 53C5	444F 53C6	444F 53C7
120:	444F 53C8	444F 53C9	444F 53CA	444F 53CB
130:	444F 53CC	444F 53CD	444F 53CE	444F 53CF
140:	444F 53D0	444F 53D1	444F 53D2	444F 53D3
150:	444F 53D4	444F 53D5	444F 53D6	444F 53D7
160:	444F 53D8	444F 53D9	444F 53DA	444F 53DB
170:	444F 53DC	444F 53DD	444F 53DE	444F 53DF
180:	444F 53E0	444F 53E1	444F 53E2	444F 53E3
190:	444F 53E4	444F 53E5	444F 53E6	444F 53E7
1A0:	444F 53E8	444F 53E9	444F 53EA	444F 53EB
1B0:	444F 53EC	444F 53ED	444F 53EE	444F 53EF
1C0:	444F 53F0	444F 53F1	444F 53F2	444F 53F3
1D0:	444F 53F4	444F 53F5	444F 53F6	444F 53F7
1E0:	444F 53F8	444F 53F9	444F 53FA	444F 53FB
1F0:	444F 53FC	444F 53FD	444F 53FE	444F 53FF
Sector type - Unknown				

torának a tartalmára lenne szükségünk, akkor  $40 \cdot 22 + 0 \cdot 11 + 0 = 880$  logikai sorszámú blokkot kell a lemeztől beolvasnunk (ez egyébként az ún. ROOT blokk — részletesen lásd később). Az Amiga felhasználója a lemezen 7 különböző blokk típussal találkozhat, amelyek a következők:

- BOOT blokk
- ROOT blokk
- BAM (szektormappa) blokk
- FILE HEADER blokk
- FILE LIST blokk
- DIRECTORY blokk
- DATA blokk.

A felsorolt 7 típus közül a BOOT blokknak és a BAM (szektormappa) blokknak nincs megkülönböztető azonosítója a lemezen. Mivel ez a blokk a lemez mindig egy meghatározott részén helyezkedik el (a 0-ás és 1-es logikai sorszámú blokkokban), ezért nincs is szükség a tulajdonképeni azonosítóra. Minden lemezt a használat előtt elő kell készíteni. Ezt a műveletet nevezik formátálásnak (inicializálásnak). A formátáló program az Amiga esetében létrehozza a blokkokat, a gyökér (ROOT) könyvtárat, a BAM (szektormappa) blokkot, a BOOT részt, valamint ellenőrzést végez a végrehajtott műveletekről. A különböző WORK Bench verziók különböző BOOT részt hoznak létre, ám a rendszer számára fontos 4 byte-os azonosító a 0-ás blokk első 4 pozícióján megegyezik ("DOS" + nullabyte). Az 1.1 és 1.2-es verziók által létrehozott blokkrészt a 2. lista mutatja. A 3. lista az 1.3 verzió által létrehozott BOOT részből csak a 0-ás blokkot tartalmazza, mivel az 1-es blokk felépítése teljesen megegyezik az 1.1 és 1.2-es verzióval létrehozott blokk rész 1-es blokkjával. De térjünk vissza a 4 byte-os azonosítóra! Valahányszor egy lemezt behelyezünk a meghajtóba, a rendszer beolvassa a ROOT blokkot (lásd később), ellenőrzőösszevizsgálatot végez, és megvizsgálja a BOOT

TRACK = 00	SECTOR = 00	HEAD = 0	BLOCK = 0000	Checksum = 444F5300
Diskname - FREE				
Sector type - Unknown				
000:	444F	5300	444F	5300
010:	444F	5300	444F	5300
020:	444F	5300	444F	5300
030:	444F	5300	444F	5300
040:	444F	5300	444F	5300
050:	444F	5300	444F	5300
060:	444F	5300	444F	5300
070:	444F	5300	444F	5300
080:	444F	5300	444F	5300
090:	444F	5300	444F	5300
0A0:	444F	5300	444F	5300
0B0:	444F	5300	444F	5300
0C0:	444F	5300	444F	5300
0D0:	444F	5300	444F	5300
0E0:	444F	5300	444F	5300
0F0:	444F	5300	444F	5300
100:	444F	5300	444F	5300
110:	444F	5300	444F	5300
120:	444F	5300	444F	5300
130:	444F	5300	444F	5300
140:	444F	5300	444F	5300
150:	444F	5300	444F	5300
160:	444F	5300	444F	5300
170:	444F	5300	444F	5300
180:	444F	5300	444F	5300
190:	444F	5300	444F	5300
1A0:	444F	5300	444F	5300
1B0:	444F	5300	444F	5300
1C0:	444F	5300	444F	5300
1D0:	444F	5300	444F	5300
1E0:	444F	5300	444F	5300
1F0:	444F	5300	444F	5300
3 . L I S T A				

részben az azonosító sorozatot. Ha az első 4 pozíción a "DOS"+nullabyte ASCII sorozat szerepel, akkor a lemez egy érvényes formázott lemez, és tudunk vele dolgozni. Ellenkező esetben a gép a NOT a DOS DISK... hibaüzenetet írja ki. Az AMIGA 1000-es ismer egy másik azonosítót is, ez pedig a "KICK". Az ilyen azonosítóval

ellátott lemezt a gép kickstart lemezként értelmezi. Természetesen a Kickstart bootolása után az A1000 sem tudja ezt a lemezt kezelni. A sorozat következő részében a BOOT blokk rejtjelmeivel és az itt található rövid programmal fogunk megismerkedni.

Farkas András

## A Copper programozása

A Copper egy **coprocesszor**, amely az Amiga custom chip-jeinek egyikében helyezkedik el. Utasításait a DMA-n (Direkt Memória Bemenet) keresztül hajtja végre. Szabályozza csaknem az egész grafikus rendszert, és szabaddá teszi a 68000-es mikroprocesszort, hogy az a program logikai lépéseit végrehajtsa. Közvetlenül befolyásolhatja a legtöbb chip kontrolregisztereinek tartalmát is. Különösképpen jól alkalmazható a grafikus képernyőn megjelenő közép-kép modifikációk és a vertikális várakozási periódus alatt megjelenő regiszterváltozók irányítására. Egyebek között szabályozási területébe tartozik a színskála, az újra pozicionáló sprite-ok és a Blitter. A Copper képes várni (WAIT) egy speciális video beam pozícióra, majd áttölteni az adatot egy rendszerregiszterbe. A WAIT utasítás időtartama alatt a Copper közvetlenül megvizsgálja a video beam pozíciószámológójának tartalmát. Tehát amíg a Copper várja, hogy a beam elérje a speciális pozíciót, egyáltalán nem használja a memóriabuszt, így a busz használhatóvá válik egy más DMA csatorna, vagy a 68000-es mikroprocesszor számára.

Amikor a WAIT állapot beteljesedik, a Copper lefoglalja valamennyi Blitter, vagy a 68000-es mikroprocesszor memóriaköreit, hogy az adatot a kiválasztott regiszterbe töltsse.

A Copper kétkörös processzor, amely csak páratlan számú memóriakör esetén jelentkezik a busznak. Ez megakadályozza az összeütközést az audióval, a lemezzel (floppy), a sprite-okkal és a legtöbb alacsony felbontású DMA képernyővel, amelyek kizárólag a páros számú memóriaköröket használják. Ezért a Coppernek csak a 68000-es mikroprocesszornál és a Blitternél van szüksége elsőbbségre.

Ugyanúgy, ahogy az összes többi DMA csatorna az Amiga rendszerben, a Copper is csak az alsó 512 Kb-ot tudja kezelni utasításaival.

### A Copper utasításai

Coprocesszorként a Copper hozzáadja utasításkészletét a 68000-es mikroprocesszor utasításaihoz. Mindössze három utasítása van, ezek azonban nagyon sok mindenre felhasználhatóak.

— WAIT: várni egy speciális képpozícióra, amely X és Y koordinátaként van megadva.

— MOVE: betölteni egy közvetlen adatot a kiválasztott regiszterek egyikébe.

— SKIP: átugrani a következő utasítást, ha a video beam már elérte a megfelelő képpozíciót.

Minden Copper utasítás két, tizenhat bites szóból áll a szekvenciális memória elhelyezésben. Mindenkor, ha a Copper egy utasítást fogad, mindkét szót fogadja. A MOVE és SKIP utasítások két memóriakört és két utasításszót kívánnak. Amellett, hogy a Copper direkt módon csak a gépregisztereket befolyásolhatja, kezelheti a memóriát is a Blitter művelet elindításával.

### A MOVE UTASÍTÁS

A MOVE utasítás átszállítja az adatot a RAM-ból a regiszterek rendeltetési helyére. A szállított adatot a MOVE utasítás második szava tartalmazza, az első szó a regiszter címét jelöli.

#### AZ ELSŐ UTASÍTÁS SZÓ

- Bit 0 Mindig 0-ra van állítva
- Bit 8—1 A célregiszter címe
- Bit 15—9 Nem használt, de 0-ra kell állítani

#### A MÁSODIK UTASÍTÁS SZÓ

- Bit 15—0 Az adat célregiszterbe töltendő 16 bitje
- A Copper a következő regiszterekbe tud adatot tölteni:
  - Bármely regiszter, amelynek címe \$20, vagy e fölötti.
  - Bármely regiszter, amelynek címe \$10—\$20 között van, ha a Copper veszélybit 1. (A Copper veszélybit a Copper kontrolregiszterében található meg. COPCON)
  - A Copper nem tud írni olyan regiszterbe, amelynek a címe kisebb, mint \$10.

A következő példában beállítjuk a Bit-Lapok kontrolregisztereit \$41000 és \$45000-re.

- DC.W \$00E0,\$0004 ; MOVE (tölts) \$0002-t a \$00E0-ra
- DC.W \$00E2,\$1000 ; MOVE (tölts) \$1000-t a \$00E2-re
- DC.W \$00E4,\$0004 ; MOVE (tölts) \$0004-et a \$00E4-be
- DC.W \$00E6,\$5000 ; MOVE (tölts) \$5000-et a \$00E6-ra
- Az eddig használt regiszterek leírása: \$02E (COPCON)

Ez egy egy bites regiszter, amely ha értéke egy, lehetővé teszi a Coppernek a Blitterhez való hozzáférést. Resetre nullára állítódik.

- \$0E0 (BPL1PTH)  
Az első Bit-Lap mutatójának felső három bitje.
- \$0E2 (BPL1PTL)  
Az első Bit-Lap mutatójának alsó tizenöt bitje.
- \$0E4 (BPL2PTH)  
A második Bit-Lap mutatójának felső három bitje.
- \$0E6 (BPL2PTL)  
A második Bit-Lap mutatójának alsó tizenöt bitje.

### A WAIT UTASÍTÁS

A WAIT utasítás várakozásra készíti a Coppert, amíg a video beam számlálók egyenlőek, vagy nagyobbak lesznek, mint az utasításban kijelölt koordináták. Amíg várakozik, a Copper kívül van a buszon, és nem használ memóriaköröket.

Az első utasításszó tartalmazza a beam pozíció vertikális és horizontális koordinátáit. A második szó jelöli az úgynevezett „mask” kialakításához használatos biteket. A „mask” egy olyan rendszer, amely megmutatja, hogy a beam pozíció mely biteit kell használni az összehasonlításban.

#### AZ ELSŐ UTASÍTÁS SZÓ

- Bit 0 Mindig 1-re kell állítani
- Bit 15—8 Vertikális beam pozíció
- Bit 7—1 Horizontális beam pozíció

#### A MÁSODIK UTASÍTÁS SZÓ

- Bit 0 Mindig 0-ra kell állítani
- Bit 15 A Blitter végének csökkentése, normál esetben ez 1.
- Bit 14—8 A vertikális pozíciót összehasonlító bitek
- Bit 7—1 A horizontális pozíciót összehasonlító bitek.

A következő példában a WAIT utasítás lemaszkolt horizontális pozícióval vár a 150(\$96)-es rasztercsíkra.

DC.W \$9601,\$FFF0

Ebben a példában pedig a WAIT a 255-ös rasztercsíkra és a 254-es horizontális pozícióra vár. Mivel ez soha nem következik be, a Copper megáll, amíg a következő üres intervallum kezdődik. Itt végződik a Copper lista.

DC.W \$FFFF,\$FFFE

A következő sorok mind a WAIT, mind a SKIP utasításra vonatkoznak.

A horizontális beam pozíció.

A horizontális beam pozíció értéke \$0—\$E2-ig terjed. Az utolsó jelentős bit összehasonlításban nem használatos, tehát a Copper műveletnek 113 pozíció áll rendelkezésére. Ez négy pixelnek felel meg alacsony, és nyolc pixelnek magas felbontásban. A horizontális blanking értéke csökken \$F0 és \$35 között. A 320 pixel széles standard képnek van egy használatlan horizontális része \$04—\$47-ig. (De csak addig, amíg a háttérszín a képernyőn van.)

A vertikális beam pozíció

A vertikális beam pozíciót fel lehet bontani egyetlen vonalra, 255-ös maximum értékkel. A valóságban 262 vertikális beam pozíció létezik. Néhány kisebb probléma merülhet fel, ha azt szeretné, hogy ebben az utolsó hat vagy hét sorban történjen valami.

### A SKIP UTASÍTÁS

A skip utasítás hatására a Cooper átugorja a következő utasítást, ha a video beam számlálói egyenlőek vagy nagyobbak az utasításban megadott értéknél.

A skip utasítás szavak tartalma ugyanaz, mint a WAIT utasításnál, kivéve, hogy a második utasítás nulla bitje egy. Ez azonosítja a SKIP utasítást.

#### AZ ELSŐ UTASÍTÁS SZÓ

- Bit 0 Mindig egyre kell állítani
- Bit 15—8 Vertikális pozíció
- Bit 7—1 Horizontális pozíció
- Ugorj, ha a beam számláló nagyobb, mint ezek a kombinált bitek.

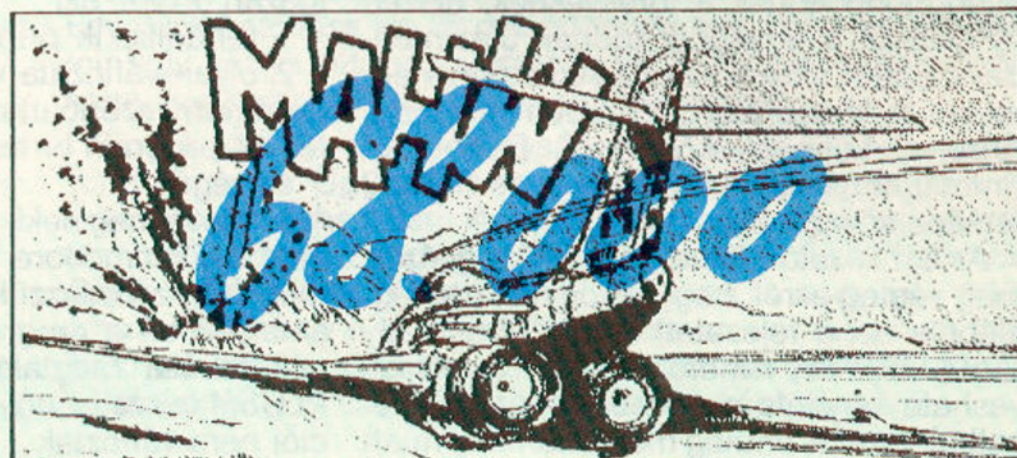
#### A MÁSODIK UTASÍTÁS SZÓ

- Bit 0 Mindig egyre kell állítani
- Bit 15 A Blitter végének csökkentése. Ez általában 1
- Bit 14—8 A vertikális pozíciót összehasonlító bitek
- Bit 7—1 A horizontális pozíciót összehasonlító bitek

Bíró Zsolt

### Kisszótár:

- coprocessor: segédprocesszor vagy társprocesszor
- video beam : a képernyő kiírósugara (x,y koordináta)
- blitter : az Amiga másik grafikus segédprocesszora
- szóhossz : 2 \* 8 bit = 1 szó, 32 bit = dupla vagy hosszú szó
- vertikális : függőleges
- horizontális : vízszintes
- custom chip: a \$DFF000 feletti memóriacímek tartoznak ide



# Egy csalódott vásárló

## Játékok és felhasználói programok C16

Megjelent az LSI-ATSZ játékleírás-sorozatának C16-C/+4-re szóló harmadik kötete. Ez felépítésében teljesen különbözik az előző két kötettől, amit Tóthné Máriássy Éva és Szabó Szilárd (1. kötet), illetve Gál József és Szabó Szilárd (2. kötet) jegyezték. Az új szerzők: Ollós László és Kiss László.

Ez utóbbi névvel találkozhattak már a Commodore-rajongók, ő volt az egyik szerzője az 1001 játék C64-en sorozat második és harmadik kötetének (bár erre csak később jöttem rá).

A könyv 10 fejezetre tagolt, némileg az 1001 sorozathoz hasonlóan. Ezek:

Bevezetés  
Játékpoke-ok  
Mikrolexikon  
Játékleírások  
TEDPRINT  
EASY SCRIPT  
SYS CRACKER  
GAME DESIGNER  
Programozástechnika  
Indítási címek

Az örökéleteket és egyéb poke-okat tartalmazó fejezet nagy segítség a felhasználóknak.

A Mikrolexikonon sajnos nem látszik az, hogy a kézirat május elsején lett lezárva, sok újabb játék hiányzik belőle. A változásokhoz tartozik, hogy nincs a programok megjelenésének éve feltüntetve, viszont osztályozzák a programokat 1-től 4-ig. Ez utóbbira — mint az olvasók által kért újításra — fel is hívják a figyelmünket már a bevezetőben. Úgy látszik, a szerzők figyelmét elkerülte az első kötet, melyben az említett „újítás” már szerepelt, csak nem számokkal 1-től 4-ig, hanem csillagokkal 1-től 3-ig. Az osztályzatok szubjektívek, ezt el lehet (sőt el kell) fogadni, de még ezen belül sem következetesek. Pl.: Auf Wiedersehen Monty ...4, de Monty on the run ...3, holott a két játék csupán a pályák felépítésében különbözik.

A következő fejezet (Játékleírások) a legbővebb. Ebben a fejezetben (és az egész könyvben, a bevezető szerint) a szerzők arra törekedtek, hogy bonyolult játékok maximális részletességű leírását adják. Sajnos, azt nem sikerült megvalósítaniuk. Már a második leírás (Magician's Curse 2) ellentmond ennek a törekvésnek, hiszen itt csupán a térképpel tudnak újat mondani, a játék leírása az előző kötetben megjelent. A térkép viszont szépen van megrajzolva, könnyen követhető. Ezek után (mint újabb leírás!!!) a Magician's Curse 3 térképe következik...

Az ezt követő SPY vs SPY leírás győzött meg végleg arról, hogy a könyvet (de legalábbis ezt a fejezetet) nem C16-os vagy C-PLUS/4-est ismerő szerző írta. Itt olvasható ugyanis a következő: „...változatból — valószínűleg memóriahiány miatt

— kimaradt jó néhány rész...”, mármint a Spectrum és a C64 verzióhoz képest. Tudtommal a Spectrum 48K-s (legalábbis azon fut a program), míg a C64-es ugyanúgy 64 Kbyte-os gép, mint a C-PLUS/4, sőt a C64-en Basic-ből csak 38911 byte érhető el, míg a C-PLUS/4-en és a bővített C16-on 60671!

A játékhoz térképet nem mellékeltek (szerintük felesleges), így ez is igazán a teljességre törekedő leírás...

Az ALIENS című program leírásával is több helyen találkozhattunk már, csakúgy, mint az előző programéval. Például lapunkban is megjelent Ripley történetének C64-re való leírása, Tihor Miklós tollából (1988/4-es szám). A C64-es verzió annyiban több, hogy 6 embert kezel, míg a PLUS/4-es csak négyet.

A már említett Auf Wiedersehen Monty leírása és térképe következik ez után. A leírás kissé bőbeszédű (és felesleges), míg a térkép ennél a játéknál nagyon gyengére sikerült. Ezt valamelyest ellensúlyozza az ez után következő Tomb of Tarrabash remek térképe.

STORM. Ez a régi korokat idéző játék következik, melyben egy hercegnőt kéne kiszabadítani. (A program GLADIATORS néven is terjed.) A szerzők ebben a részben humorizálnak egy kicsit (pl. a varázsló bizományi varázslása abból áll, hogy megpróbál Reformot szerezni...), és talán ez ment a térkép rovására, ami nem méltó ehhez a játékhoz. Azt már csak félve merem említeni, hogy a szerzők szerint ez a könyv a „Világirodalom remekei” sorozatban jelent meg. (!?!)

Tir Na Nog. A leírás jó, részletes, a térkép is megfelelő, az egyetlen probléma, hogy szinte szóról szóra megegyezik az 1001 játék C64-re/3-ban megjelenttel. (Itt jöttem rá arra, hogy azon könyv egyik társ-szerzője szintén Kiss László volt...)

A következő fejezetek felhasználói programokkal foglalkoznak, az első a TED PRINT nevű magyar rajzprogrammal. Ezzel a programmal a Commodore Újság 1987/8—9-es száma foglalkozott, a PROGRAMFUTAM rovatban. Talán ez a leírás sikerült a legjobbra, a legrafináltabb alkalmazási lehetőségeket is jól bemutatja.

Megtalálható a könyvben az Easy Script szövegszerkesztő leírása, aminek az tud igazán örülni, aki

1. rendelkezik printerrel,
2. olyan változata van, amelyik ismeri a formátumvezérlő utasításokat, vagy a Deltex+4 program birtokosa, és végül
- 3., nincs meg neki, vagy nem olvasta például a Commodore Újság 1988/4-es számát, vagy a Fedezzük fel... című könyvet, amelyekben a program C64-es változatának leírása megtalálható. Ez egyben a PLUS/4 leírás is, ugyanis a program funkciói nem változtak...

A SYS CRACKER leírása viszont hiányos, talán legalább a négy szempontot elárulhatták volna, ami alapján a program a lehetséges indítási címeket keresi...

A GAMES DESIGNER játékszerkesztő következik, amihez talán csak annyit lehet hozzátenni, hogy a program túl régi, túl keveset tud, mégis a C64-es és Spectrumos legjobbakkal van egy lapon emlegetve...

Igazán ingoványos területre a „Programozástechnika” fejezettel tévedtek, ahol is egy örökélet kiderítését és egy keretre írást mutatnak be. Előbbinél kapunk egy általános receptet a játékok örökéletesítéséhez, a Magician's Curse 1—2—3. (a továbbiakban M1, ill. M2, M3) példáján keresztül, itt éppen az M2-t vizsgálják. Ez már csak azért is jó, mert arra a játékra, ahol az életek száma fogy, nem (vagy csak igen nyakatekert módon) alkalmazható, ugyanis az M2-ben egy folyamatosan csökkenő vízszintes vonal jelzi az „energiákat”.

Ráadásul az M1—2—3 olyan szépen van megcsinálva, hogy az egymástól csupán a pályák rajzában különbözik, így — mivel a vezérlő rész teljesen azonos — a 100+4/2-es kötetben megjelent örökélet erre a játékra is tökéletes. De ha mindez nem lenne, akkor is elég nehéz lenne rájönni, miért ott kezdik a visszafejtést a szerzők, ahol az 50-es címmel találkoznak...

Az írás a keretre a +4-en mindig kényes téma. Itt sem sikerült igazán jól megfogni, a program a haladóknak gyenge, a magyarázat viszont a kezdőknek erős. Esetleg jó lett volna egy BASIC töltős változatot is leköszölni, hogy aki nem óhajt foglalkozni a gépi kódú programozással, az is használhassa a programot.

A könyv összességében sajnos nem éri el az általam várt (és az előző kötetekben tapasztalt) szintet, túlságosan is hasonlít az „1001 játék C64-re” sorozathoz. Kevés leírást közöl, és azok közül még kevesebb az eredeti, így a 110 oldalas könyvért soknak érzem a 137 forintos árat. Sajnos, valószínű (bár ne legyen igazam!), hogy a sorozat ezzel a könyvvel jelentős hanyatlásnak indul, ha nem próbálnak visszatérni a bevált formához, vagy legalább közelíteni hozzá. Talán nem volt szerencsés egy sikeres munkán és elfogadott formán változtatni. Egy kérdés zárójelben: esetleg a régi szerzők nem vállalták?

Lengyel István



# A jövő háttértárolói

## C 64-es szoftverek CD-n

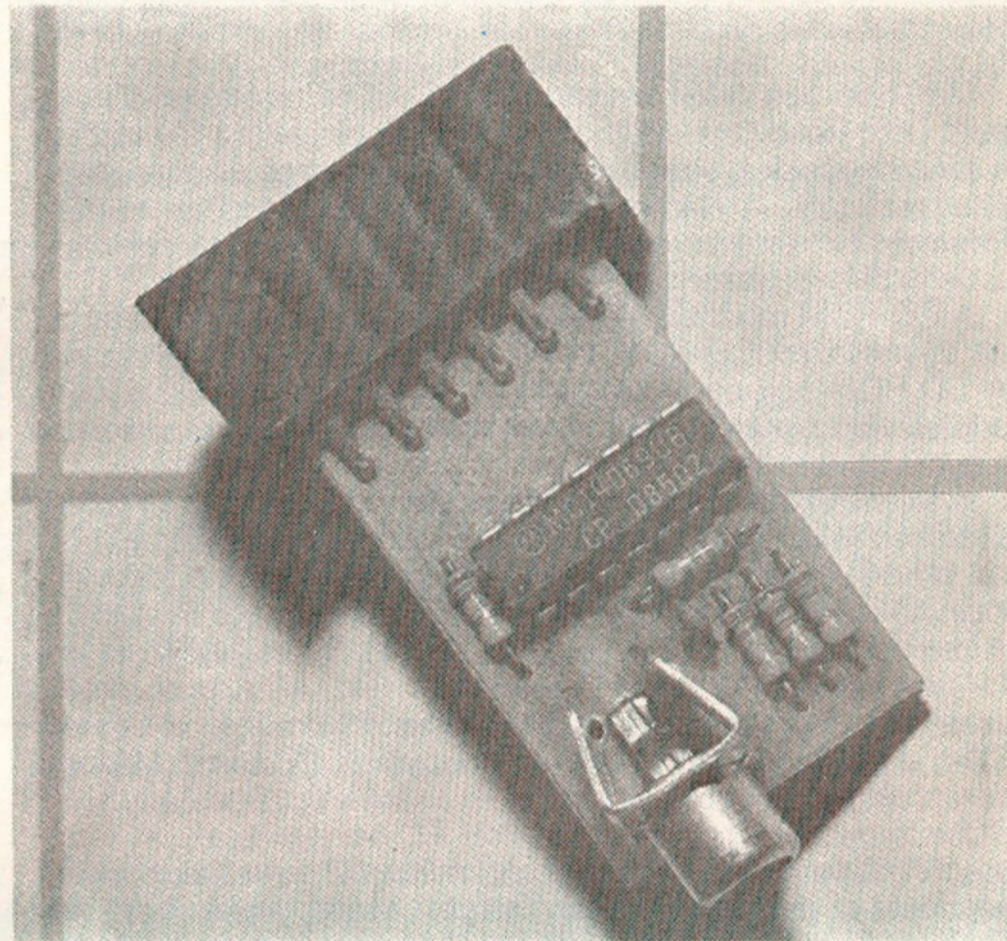
Ma mindez realitás. A CD (Compact Disc) mint háttértároló megjelent a C64-eshez a piacon. A gépen kívül csak egy „normál” lejátszóra van szükség. De lépést tarthatunk a profi CD-ROM egységekkel?

Ha a jövőben valaki egy lézerlemez tesz a lejátsszókészülékbe, lehet, hogy a zenehallgatáson kívül valami más szándék van. Mert a szóban forgó CD-ről jövő „zaj” mindennek nevezhető, csak szépnek nem. Mégis lehet, hogy nem tévedés ennek a lemeznek a kiválasztása.

De semmi baj, szó sincs valami önkínzásról, épp ellenkezőleg. Úgy néz ki, fölfedezték a Compact Discet mint C64-es háttértárolót. A düsseldorh szoftverház, a Rainbow Arts elsőként dobott piacra egy játékgűjteményt ezen a szívárványszínbén játszó optikai tárolóeszközön. Az ominózus „szelet” olyan játékokat tartalmaz, mint a Dropzone, a Jinks az Impossible Mission, a Fist II és hat másik klassz program.

Egy teljesen normál, a kereskedelemben kapható CD lejátszóra van szükség, hogy tíz játékot élvezhessünk. A csatlakozása nagyon egyszerű. A csomagban találunk egy összekötőkábelt és egy adaptert (1. kép). Ez teszi lehetővé, hogy a lejátszót a cinch kimenetek, sőt a fejhallgató segítségével a C64-es kazettaportájára csatlakoztassuk (2. kép). Az adatok átvitele hasonlóképpen történik, mint azt a gyorsított (turboított) kazettás programoknál megszokhattuk. A CD egy

lemezhez hasonlóan sávokra (trackek) van fölosztva. A CD nullás trackjén egy speciális gyorsító található, amelyet elsőként, normál sebességgel kell betölteni. Ez azután betölt egy főmenüt, amely ezután az egyes játékok indításáról és kiválasztásáról ad információt.



2. kép: Ez a kis adapter illeszti a CD lejátszót a C64-hez. Az átviteli sebesség több mint 2,5-szer jobb, mint egy Turbo Tape, vagy akár egy 1541 esetén.



1. kép: A Rainbow Arts CD komplett csomagja. A kábel a CD lejátszóhoz cinch kimenettel készül. Más szabványokhoz kiegészítő kábel kell, ez nincs benne a csomagban.

### A GYORS ADATHORDOZÓ

A játékszoftverek trackenként találhatók a lemezen. Az adott sávokat a főmenüben adott utasítások szerint választhatjuk ki. Egy játék betöltése kb. 30 másodpercig tart. Ezzel a CD audio mintegy két és félszer gyorsabb, mint egy turbóított Data-sette, vagy egy 1541-es (floppygyorsító nélkül).

Ennek ellenére a lemezegységgel az audio CD nem veheti föl a versenyt. Ugyanis a saját adataink és programjaink kimentése oda lehetetlen. Azaz nem használhatjuk föl azt szövegszerkesztésnél vagy adatbáziskezelésnél. És a highscore listák fölírása sem megy. Ennek oka nem is annyira a fölülírható CD, amely a technika mai állása szerint még nem állítható elő a kívánságainknak megfelelően sorozatban, hanem a hagyományos CD lejátszók teljes írásképtelensége.

## TÁROLÓK, TÁROLÓK, TÁROLÓK

A CD leglényegesebb előnye a tárolókapacitás. Maximum 6 Mbyte adatot vagy információt lehet fölvenni oda. Ez több mint ezer géppel írt oldalnak felel meg. Így tíz top játékot 100 márkás ár alatt lehet árusítani. S mivel az adaptert csak az első CD-vel kell megvenni, az ár méginkább csökkenhet.

Egy professzionális CD-ROM egységgel való összevetés természetesen kijózanító hatású. De azt nem felejthetjük el, hogy a C64-es CD most 100 márkába kerül, s ha nincs még CD lejátszó, azért úgy 300 márkát kell még letenni az asztalra. A profi rendszer például egy irodai alkalmazáshoz minimum 2000 márkába kerül, egy-egy adathordozó lemez kb. 300-ba, de inkább többre. Ha azt is figyelembe vesszük, hogy a profi rendszer másra nem használható, és a C64-eshez sem kapcsolható, a CD lejátszó viszont „zenegép” is, akkor a C64-es CD sokkal inkább ajánlatos kiegészítés, mint azt első pillantásra gondolnánk.

Most csak azt kell kivárni, elfogadják-e a felhasználók az audió CD-t mint kiegészítő tárolót. Egy olyan tárolóeszközt, amely sok jó programot elfogadható árértékű kínál, s azt széles hozzáférhetővé teszi, biztos, hogy fölvesz a piac. A jövőben minden esetben jó lesz megmondolni, hogy a kedvenc együttesünk, vagy a legújabb szoftverek lemezét tesszük „föl” a CD-re!

## A CD ROM

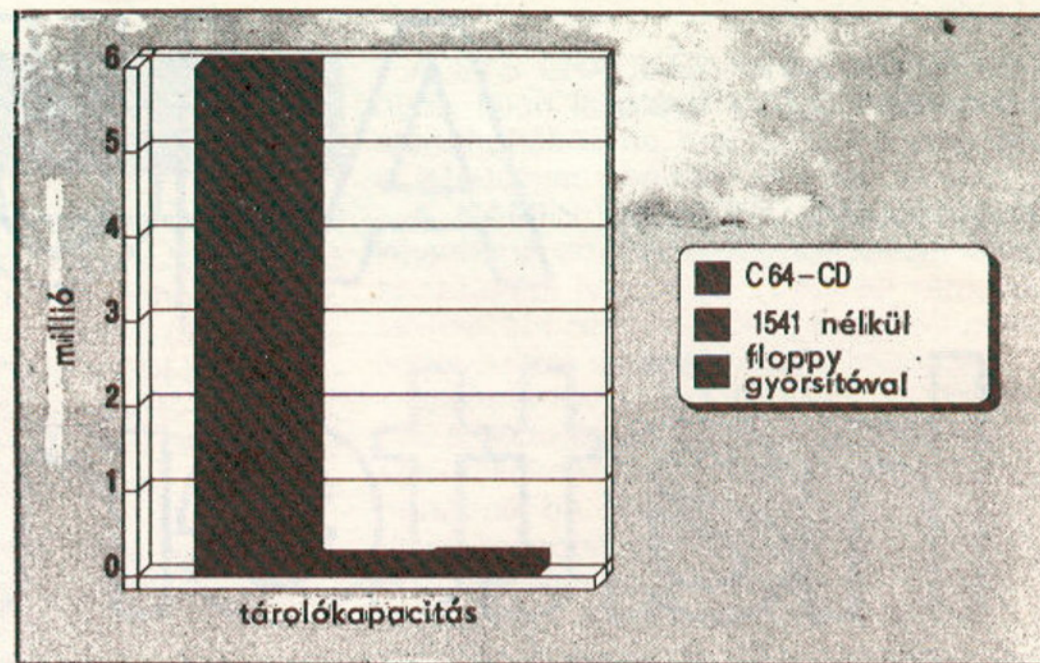
Már 1985-ben találkozhattunk a CD-ROM tárolókkal (Compact Disc Read Only Memory, azaz csak olvasható CD tároló) a számítógépek szabadpiacán. 1986 végén ezek alkalmazását kiterjesztették. A Német Szövetségi Köztársaságban háttér-információs rendszereket kínálnak optikai lemezen, például a belföldi gyógyszeripar összes kapható termékét stb.

A CD-ROM rögzített adatait egy speciális olvasóegységgel, az úgynevezett CD-ROM Drive segítségével lehet leolvasni. A CD-ROM egy nagy hátránya mindjárt kézenfekvő: a fölirt adatok változtatására nincsen mód, azaz rá vagyunk szorulva a gyors update szervizre.

A fejlesztő laboratóriumok legmodernebb készülékei többszörösen fölülírható, Gigabyte nagyságrendű tárolókapacitással rendelkeznek. A ma a kereskedelemben kapható, csak olvasható lemezekre azonban még így is mintegy 550 megabyte információ fér el (a 3-as és 4-es ábrák). Ez kb. 3200 da°ab 1541-es formátumú lemez kapacitását teszi ki. A profi rendszerek átviteli sebességei sokszorososan fölülmúlják a C64-es CD-t, de egy gyorsított 1541-es floppyét is (5. ábra). Másodpercenként maximum 500 kbyte átvitele lehetséges egy CD-ROM lemezről egy csatlakoztatott PC-be (táblázat). A letapogató lézersugár csekély pozicionálási sebessége miatt azonban a gyakorlatban az átlagos átviteli érték úgy 150 kbyte/sec körül van. Egy gyorsított 1541-es maximum 10 kbyte/sec-ra képes. Egy PC fixlemezen (általában 20 és 80 mbyte közötti kapacitással) az átvitel még háromszor vagy négyszer gyorsabb lehet.

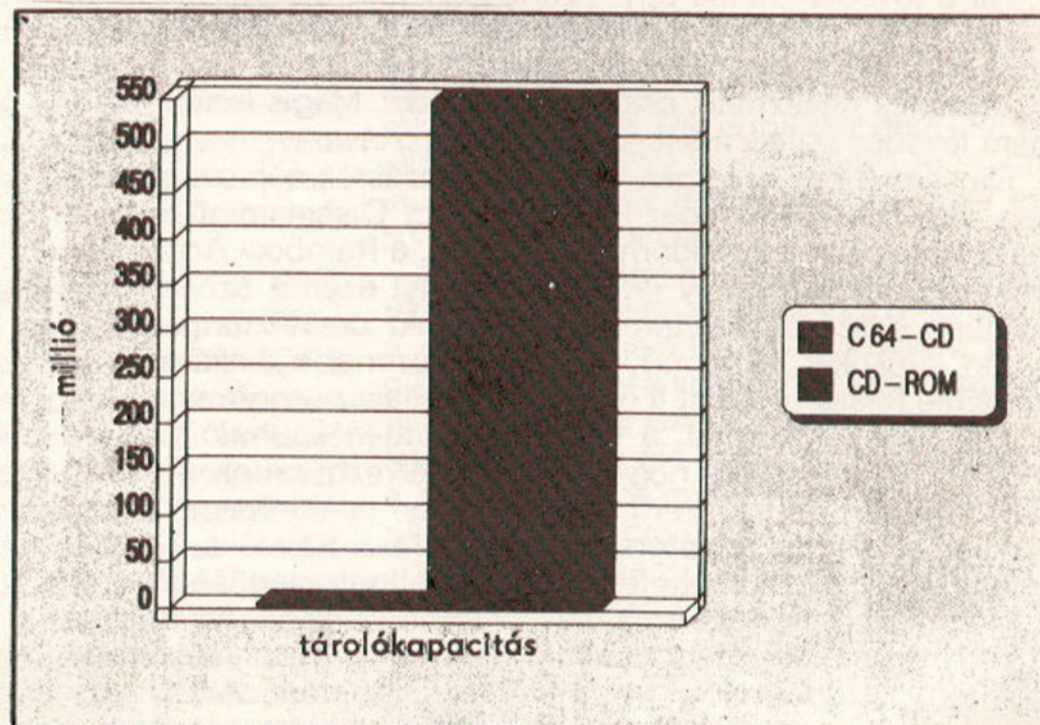
Igen komolyak a különbségek a CD-ROM és a Rainbow Arts CD-je, valamint egy 1541-es között.

	C64-CD	CD-ROM	1541	1541+Speeder
tárolókapacitás	kb. 6 Mbyte	kb. 550 Mbyte	164 kbyte	max. 187 kbyte
átviteli sebesség	kb. 1,3 kbyte/sec	max. 500 kbyte/sec átlag 150 kbyte/sec	0,4 kbyte/sec	max. 10 kbyte/sec

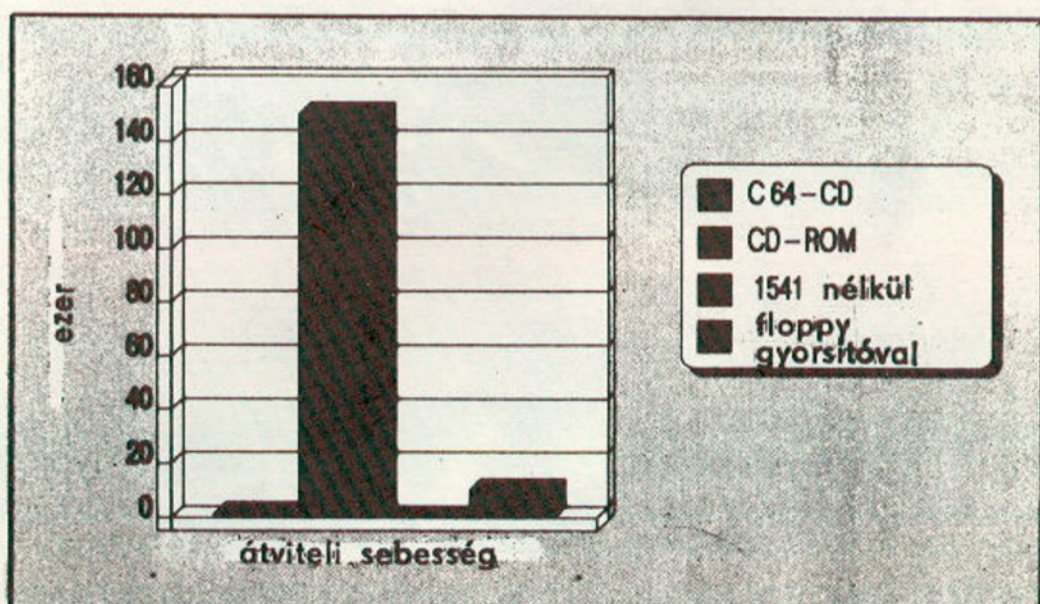


3. ábra: A képi összevetés többet mond, mint a csupasz számok (lásd a táblázatot).

A C64-CD mintegy negyvenszer annyi adatot képes tárolni (6 Megabyte), mint egy 1541-es (170 kbyte).



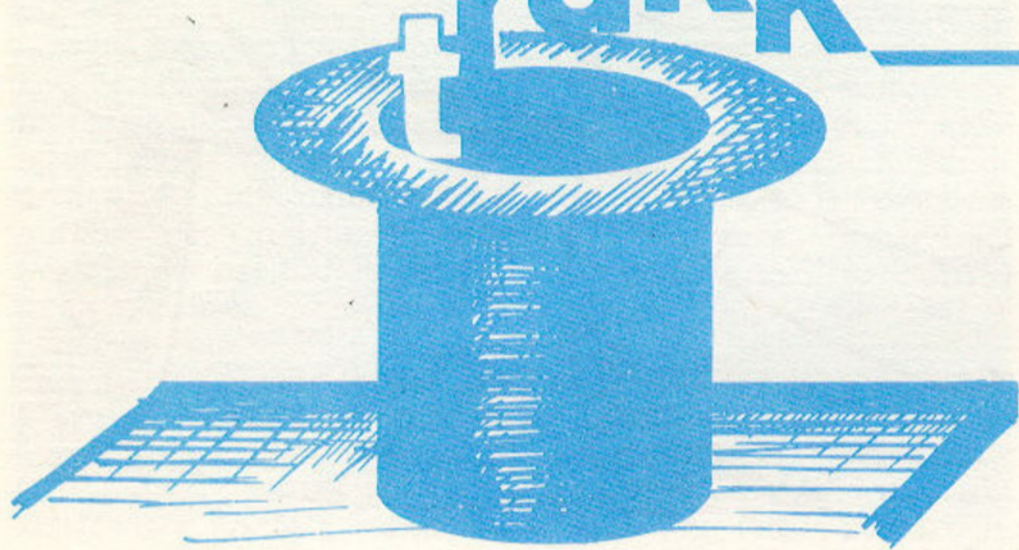
4. ábra: A profi CD-ROM viszont ennél is százszor többet tud. Összesen 550 megabyte, azaz több mint 90 000 géppel írt oldalnak megfelelő információnak van helye.



5. ábra: Az átviteli sebességnél is a CD-ROM a győztes. Még egy floppygyorsítóval ellátott 1441-es sem képes ezzel lépést tartani.

# Tippek Trükkök

## — a floppyhoz



A különböző lemezegységek nem annyira parancskompatibilisek egymással, mint azt szeretnénk. Segíteni szeretnénk az egyszerű programok átírásában az egyikről a másikra.

hogyan hatékonyabban kezelhessük az értékes floppykat. A három új parancs a SUBDEF, SUBOPEN és a SUBCLOSE.

SUBDEF "név",Ux,Dx,Px to Px

Ezzel definiálhatjuk a lemez egy részét mint partíciót. Az U-t követő paraméter a lemezegység száma, a D mögötti pedig a meghajtószám (ez mindig nulla). A Px jelzi a partíció által lefoglalandó tracket (kezdő- és végtrack).

SUBOPEN "név",Ux,Dx

Ezzel lehet kiválasztani egy partíciót. Ha azt az első alkalommal hívjuk föl, azt előbb át kell alakítani aldirectoryvá. Ez a HEADER "név",ID utasítással történik. Ez azonban csak akkor működik, ha a partíció megszerkesztéséhez legalább három sávot igénybe vettünk.

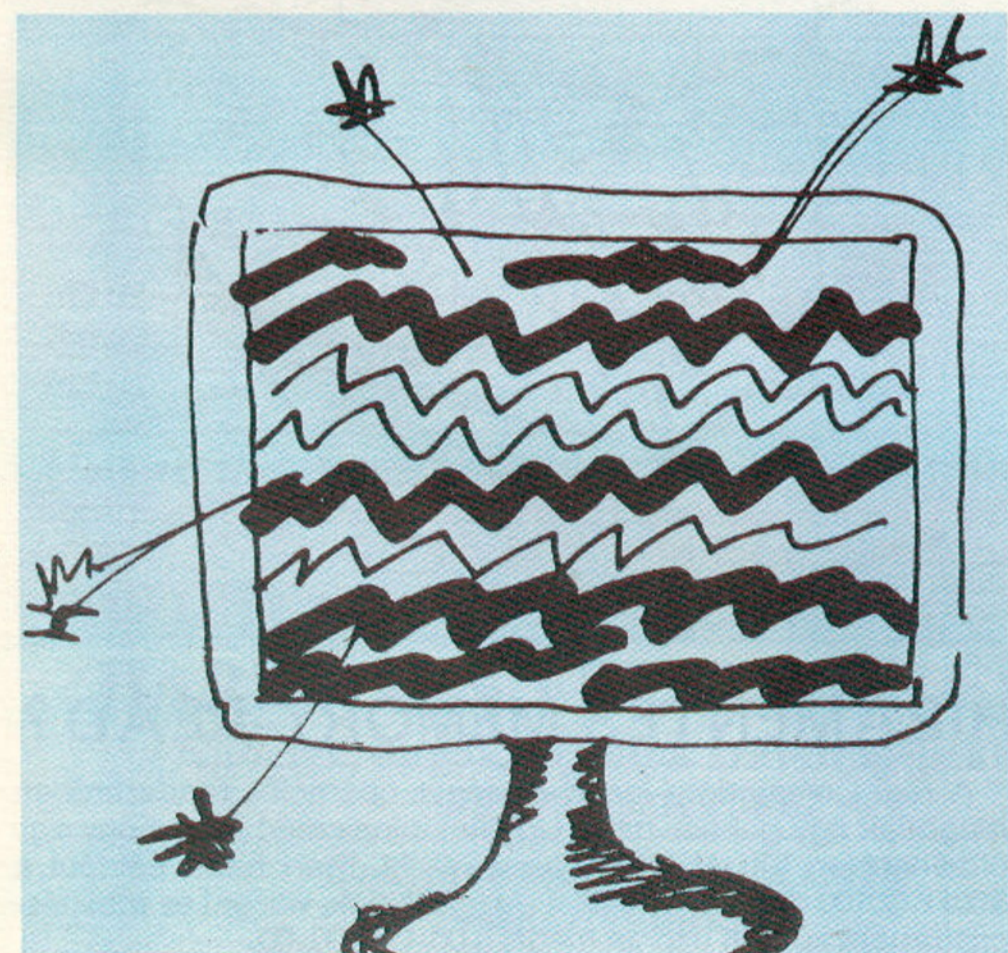
SUBCLOSE

Ezzel az utasítással visszatérünk a fő tartalomjegyzékbe.

## ALDIRECTORYK AZ 1581-ESSEL

Az aldirectoryk kezelése a 1581-es esetében igen komplikált, mivel az igen jó BASIC 7.0 ezt nem támogatja. A mellékelt „Partition” program három új utasítást bocsát a rendelkezésre,

○	partition	1300 1421	○	1360 : 45 ce 53 55 42 44 45 c6 75	○	13d0 : a8 13 88 d0 eb 8d aa 13 3a	
	-----			1368 : 00 a9 e7 20 c1 a3 a9 2f 79		13d8 : 18 a9 28 6d a8 13 8d a8 27	
○	1300 : a2 05 bd 0c 13 9d 0c 03 6a			1370 : 8d 00 11 a9 01 20 c0 a6 d8		13e0 : 13 a9 00 6d a9 13 8d a9 32 ○	
○	1308 : ca 10 f7 60 12 13 2e 13 7d		○	1378 : 4c 9a a3 a9 2f 8d 00 11 b1		○	13e8 : 13 20 7b 13 98 18 69 0a 51
	1310 : 3b 13 48 a9 13 a0 53 20 e0			1380 : ad 12 01 8d 01 11 a9 3a dc			13f0 : 48 a9 2c 99 03 11 c8 ad 82
	1318 : e2 43 90 0d aa 68 8a 29 cc			1388 : 8d 02 11 a0 00 b9 b7 12 40			13f8 : 17 01 99 03 11 c8 a9 00 55 ○
○	1320 : 7f 18 69 27 a2 00 4c b2 ab		○	1390 : 99 03 11 c8 cc 11 01 90 83		○	1400 : 99 03 11 c8 ad a8 13 99 18
	1328 : 43 68 38 4c 21 43 09 80 88			1398 : f4 60 a9 e6 20 c1 a3 20 e2			1408 : 03 11 c8 ad a9 13 99 03 1b
	1330 : 38 e9 27 aa a9 13 a0 53 d8		○	13a0 : 7b 13 98 18 69 03 d0 cd 5c			1410 : 11 c8 a9 2c 99 03 11 c8 fd ○
○	1338 : 4c 6a 51 29 7f 38 e9 27 e3		○	13a8 : 00 00 00 a9 e6 a2 f9 20 89		○	1418 : a9 43 99 03 11 68 4c 75 9a
	1340 : 0a a8 b9 4e 13 48 b9 4d cb			13b0 : c3 a3 38 ad 19 01 ed 17 88			1420 : 13 ff 00 00 00 00 ff ff 33 ○
	1348 : 13 48 4c 80 03 68 13 99 95		○	13b8 : 01 8d 19 01 a0 10 a9 00 18			
○	1350 : 13 aa 13 53 55 42 43 4c f5			13c0 : 4e 1a 01 6e 19 01 90 03 0b			
	1358 : 4f 53 c5 53 55 42 4f 50 72		○	13c8 : 18 69 28 6a 6e a9 13 6e 4a			



## PROBLÉMÁK AZ 1541-ESSEL

Alkalmanként előfordul, hogy a floppy megbokrosodik. Olvasási hibák jelentkeznek mindenféle ok nélkül, lemezeket nem tudunk formátálni. Néha a hiba oka egészen egyszerű. A monitor, illetve egy tv erős elektromágneses sugárzást bocsát ki, ami igen zavaró a lemezegység érzékeny író/olvasó fejére és a floppy elektronikára. Ilyenkor fordul elő, hogy az adatokat részben vagy teljesen nem lehet rendesen olvasni. Vigyük tehát a floppyt vagy a monitort messzebb! De a soros busz kábeleit se érjenek bele a „sugárövezetbe”. Gyakran már ez is megoldást jelent. Ha ez sem segít, lehet, hogy az író/olvasó fej elállítódott. Ennek előidézője, hogy a fej gyakran ütközik neki a mozgását gátló mechanikai sorompónak (amikor zörög a „daráló”). Ezt a hibát a szakszervizben lehet kijavítani.

## SZÉTROMBOLT LEMEZEK?

Sok olvasó ismeri már ezt a jelenséget, mégis ismét és ismét szóvá kell tenni a kezdők érdekében, hogy amennyiben a programok tárolására a SAVE "@:név",8 utasítást használjuk, fennáll a lemeztartalom teljes összekeverésének veszélye.

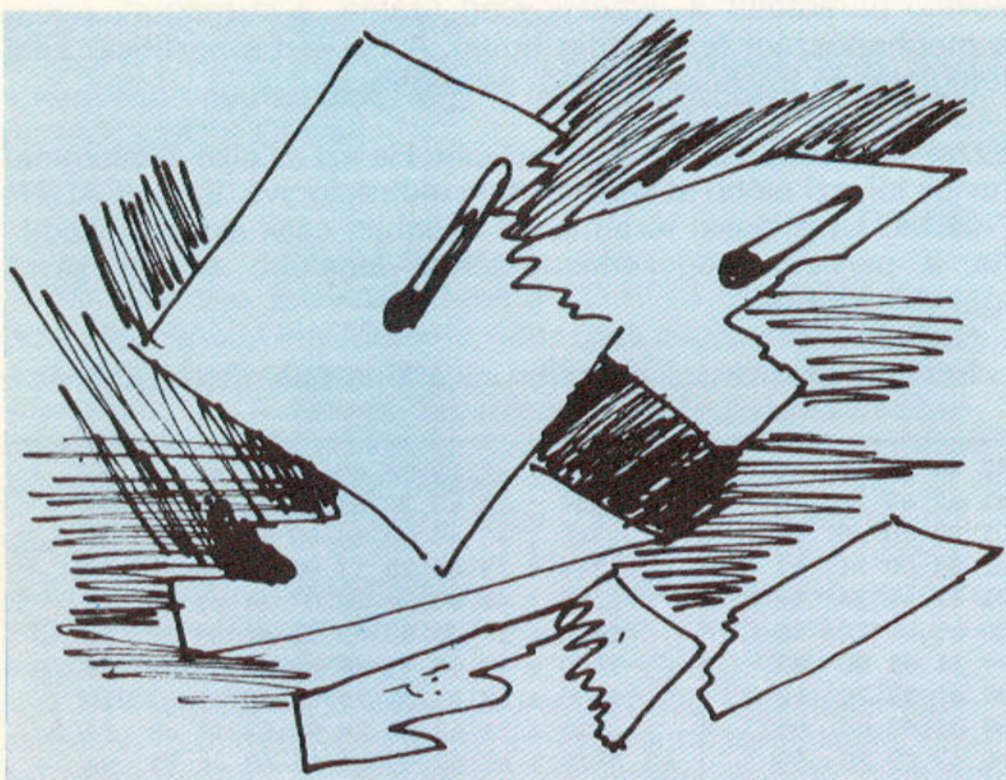
Amikor ezt az utasítást használjuk, a meghajtó operációs rendszere előbb fölírja az újat a lemezre, majd ezután törli a régit. Rendes körülmények között a parancs hibátlanul mű-

ködik. Alkalmoszerűen azonban hiba lép föl, méghozzá akkor, ha a floppy-nak a munkájához egy újabb pufferre van szüksége. A rendelkezésre álló lehetőségek közül azonban az operációs rendszer azt a puffert is kiválaszthatja, amelyben a behelyezett lemez blokkfoglaltsági térképének (BAM) köztes tárolása történt! Miközben tehát a programunkat lemezre írjuk, teljesen tönkretesszük ezt az életfontosságú táblázatot, aminek az a következménye, hogy olyan blokkokat is fölírunk, amelyek más állományokhoz tartoznak. A lemezen tárolt információkat ezzel igen komolyan károsítottuk. Azaz egy megírt program új változatának lemezre írásakor vagy adjunk annak új nevet, vagy töröljük a régi állományt:

```
OPEN 1,8,15, "S:név":CLOSE1
```

```
SAVE "név",8
```

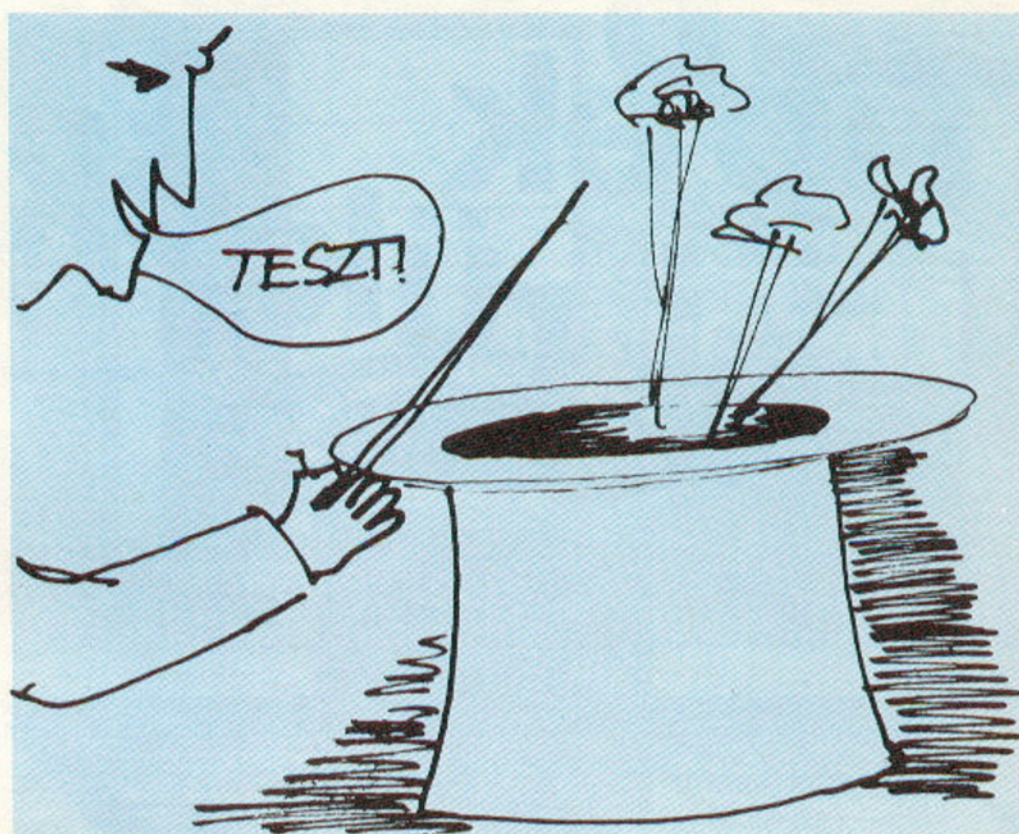
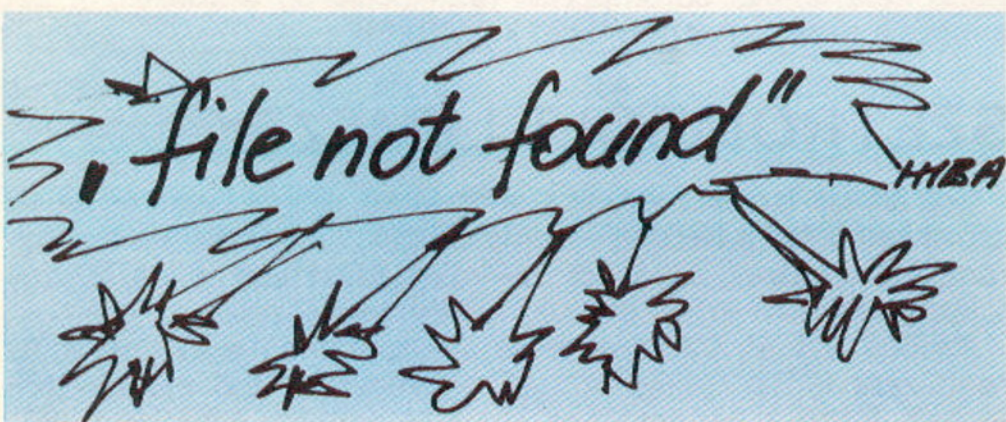
Az 1581-es és a (fémdobozos) C128D-be épített 1571-es esetében ezt a hibát kiküszöböltük, igaz, cserébe mellékeltek egy másikat. Itt ugyanis ha majdnem teljesen tele van a lemez, akkor megtörténhet, hogy az új állomány lemezre írása nem lesz befejezve, mert az operációs rendszer csak addig ír, amíg szabad helyet talál.



## PROGRAMOK A FLOPPYBAN

Az 1541-es, illetve egy 1570/71-esnél a 41-es módban könnyen el lehet indítani egy, a floppy RAM-jába vitt programot: `OPEN1,8,15,"& név":CLOSE1`.

Mivel a többi Commodore floppy parancskompatibilis az 1541-essel, azt hinné az ember, hogy az 1570/71-esen a (128-as módban), illetve az 1581-nél ez a parancs ugyanígy hangzik. Sajnos, az általam történt alkalmazásnál a „file not found” hibajelzést kaptam. A gépkönyvben nem volt megadva más szintaktika. Csak amikor a ROM listát néztem át, találtam meg az & parancs helyes szintaxisát: `OPEN 1,8,15,"&:név":CLOSE1`. Ez a fölírás azonban az 1541-esen és az 1570/71-esen a 41-es módban (azaz egy C64-essel) nem használható. Az alkalmazó főprogramnak tehát minden esetben ki kell derítenie, hogy a rendelkezésre álló floppy milyen típusú, illetve milyen módban dolgozunk.



## TRÜKKÖS LEMEZFORMATÁLÁS

A formatálásnál bizony sokáig szoktunk gondolkodni azon, mi is legyen a lemez neve. A mellékelt program egy kis trükköt alkalmaz.

Az első gombnyomásra a lemezt a „teszt” néven formatáljuk. Miközben a művelet tart, már a képernyőn áll a „lemez neve?” kérdés. Bőven van időnk meggondolni magunkat. Ha beadtuk a kívánt nevet s a floppy megformálta a lemezt, mindjárt át is írja az ID nélküli formatálási paranccsal, ami alig vesz időt igénybe.

```
0 PRINT „kérek egy gombot megnyomni a formatáláshoz!”
```

```
1 GET A$: IF A$ = "" THEN 1
```

```
2 OPEN 1,8,15, "N:TESZT,ID"
```

```
3 INPUT "A lemez neve legyen: "; A$
```

```
4 CLOSE1: OPEN1,8,15, "N:" + A$: CLOSE 1
```



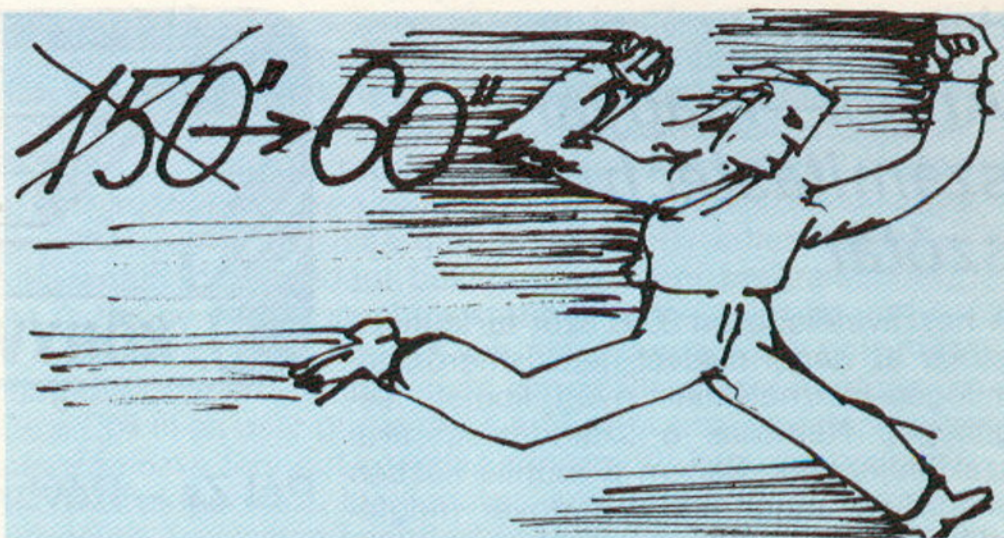
## HIBAMENTES MEMORY-READ?

A régi Commodore lemezegységek operációs rendszerében megbújik egy gonosz hiba. Ugyanis nincs mód arra, hogy egy oldalhatáron túl adatokat olvassunk ki. Ilyen határ például a \$00FF \$0100 cím. Az utasítást megszakítjuk, viszont az utasítás-csatornára küldött üzenet mégis a 00, OK, 00, 00.



Ennek oka pedig a következő: Az M—R parancsnál először kiküldünk egy meghívott byte-ot. Ha több mint egy byte olvasása szükséges, akkor annak a puffernek a címét, amelyet a hibajelzésekre használunk, a kiolvasandó terület elejére állítjuk. A DOS így a maradék byte-okat az utasításcsatornán (15-ös) mint egy hibajelzést olvassa be. Ez az \$D443-nál (az 1581-esnél pedig a \$938F-nél) történik. Ott előbb egy alprogramban a \$D137-nél (1581: \$9071) kiolvasunk egy byte-ot a hibajelzések pufferéből (ami, ugye, most M—R terület), majd eggyel megnöveli a futó cím low byte-ját. A zeroflaget akkor kapcsoljuk be, ha a cím növelésekor túlcserélődés következett be. A főprogramban a kiolvasott byte-ot a kivitelhez köztesen tároljuk. Ha most a zeroflag be van kapcsolva, ami a page váltáskor megtörténik, az aktuális puffercímét a \$024D-re állítjuk be (ez a hibajelzés puffer eredeti helye), így az ezt követő byte-ok a hibajelzést jelentik. Ha ezt tudjuk, az már nem olyan rossz, mert ügyes programozással az oldalhatáron (page) túlnyúló olvasást el lehet kerülni.

A Memory-Read rutinnak azonban van egy másik gyengéje is. Nem lehetséges ugyanis az M—R használata, ha a területben a hibajelzés puffer is benne van (kezdeté \$02D5, 1581; \$02D0). Itt a puffer címének elérésekor mindig a hibacsatorna kerül kiolvasásra. Ennek a lekérdezése a \$D414-nél (1581; \$9360) van. Az mindenesetre rejtély, miért történik ez a lekérdezés. A 1541/1570/1571-es esetében ez a lekérdezés megkerülhető úgy, hogy egyszerűen a tükrözött RAM területet olvassuk ki. De ez csak ezeknél mehet, mivel ott a RAM kódolásnál néhány címvezeték szabadon marad. Az 1581-esnél ezt a trükköt nem lehet alkalmazni, mivel az a RAM és a ki/beviteli építőelemek között byte értéként mindig a magasabb értékű címbyte-ot adja ki.



## AZ UNI-COPY JAVÍTÁSA

Az 1570/71-es és a 1581-es lemezegységekhez adott rendszerlemezen mellékelt Uni-Copy nevű program kissé lassú. Bár a C128-as tulajdonosoknak esetleg megvan a Der Hexer 1,4 másoló, de ez nem feltétlenül van így. Van azonban egy másik lehetőség is. Adjuk be az alábbi sorokat, amely a lemezegységet dupla sebességre állítja át, majd ezután elindítja az Uni-Copyt:

```
1 OPEN 15,8,15, "U0>M1" : CLOSE1
2 OPEN 15,9,15, "U0>M1" : CLOSE1
3 RUN "UNI*"
```

Ettől kezdve pl. 140 blokk másolása az 1571-esről az 1581-re már nem 150, hanem 60 másodpercet vesz igénybe. Adott esetben ez is nagy haladás.

1989. október végén jelenik meg az **1001** c. könyvsorozat soron következő, **C64** és **AMIGA** tulajdonosoknak szóló **5. kötet**. Röviden a tartalomról: **DEFENDER OF THE CROWN, HEROES OF LANCE, STEALTH FIGHTER, STRIKE FLEET, ZAK McKRACKEN, AMICA PAINT, GIGAPAIN, MACRO ASSEMBLER.**

225 oldal, 256,- Ft

További könyvajánlatunk:

<b>1001/1</b>	114 oldal,	111,- Ft	<b>1001/4</b>	149 oldal,	168,- Ft	<b>100 + 4/3</b>	110 oldal,	137,- Ft
<b>1001/2</b>	134 oldal,	99,- Ft	<b>100 + 4/1</b>	184 oldal,	161,- Ft	<b>A sárkány háborúja</b>		
<b>1001/3</b>	161 oldal,	155,- Ft	<b>100 + 4/2</b>	209 oldal,	169,- Ft		136 oldal,	99,- Ft

Szereted a kockázatos kalandokat, a hihetetlen izgalmakat? Szereted a fantázia világában játszódó történeteket? Kedved a számítógépes kalandjátékokat? Például olyanokat, mint a *Bard's Tale*? Mindezt esetleg — egy könyvben?! Akkor most eljött a Te idő! Októberben jelenik meg kiadásunkban egy ilyen jellegű könyv (**A sárkány háborúja**), amelynek a főhőse Te lehetsz!

## HARDWARE FELSZERELÉSEK

Kábelek, dugók, aljzatok számítógépekhez nagy választékban!

Tekintse meg játékkazetta választékunkat!

**LSS SHOP**

1077 Bp. Majakovszkij u.91. Telefon.: 12-21-076

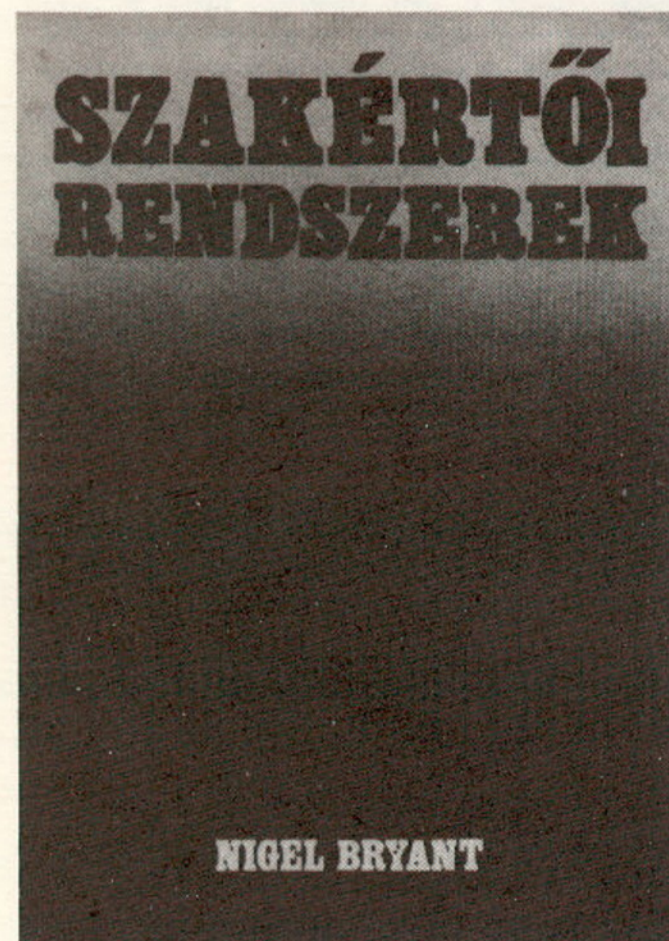
Homonnay P.:

**Angol—magyar  
számítástechnikai  
szótár**

A hozzávetőlegesen 3000 szót tartalmazó szótár az angol nyelvet nem ismerő számítógép-használóknak nyújt komoly segítséget. Nemcsak a számítástechnikai szakkifejezések, hanem a munka közben előforduló angol szavak magyar megfelelőit is megtalálhatjuk a szótárban.



Ára: 76 Ft  
Megjelent: 1986



Pál Zs.—Révbíró T.:

**Hetedhét  
Commodore 64**

A nagy sikerű sorozat e kötetéből a C64-es megismeréséhez juthat játékos, könnyed formában gyerek és felnőtt egyaránt. Ez a könyv — a sorozat többi tagjához hasonlóan — didaktikus felépítésével mind a csoportosan, mind az egyénileg tanulóknak hasznos segítőtársa lehet.



Ára: 99 Ft  
Megjelent: 1985

N. Bryant:

**Szakértői  
rendszerek  
választása**

Ára: 232 Ft  
Megjelent: 1989

Őri István:

**Fedezzük fel...**

A könyv a C64-es gépen igen elterjedt EASY SCRIPT, ill. ennek magyar változatát, a DELTEX szövegfeldolgozó rendszert kívánja minden alkalmazó számára közzétehetően bemutatni.

Az 1. fejezet a szövegfeldolgozás általános ismertetését adja, a 2. fejezet példaanyagokon keresztül bemutatja a kiválasztott EPSON nyomtatócsalád funkcióinak működtetését BASIC-ből és EASY SCRIPT szövegszerkesztőből. A 3. fejezet az EASY SCRIPT igényesebb felhasználási lehetőségeit tárgyalja, a 4. fejezetben bőséges példaanyagot mutat be, az 5. fejezet pedig egy komplett iratelőállítási és -kezelési rendszert ismertet.



Ára: 190 Ft  
Érkezik

E könyv a nem szakember olvasónak mutatja be ennek a hazánkban is lassan elterjedő „eszköznek” a lényegét. Megtudhatjuk belőle, hogy mit várhatunk egy ilyen rendszertől, és milyen szempontok alapján kell azt kiválasztanunk. Azok az érdeklődő — többnyire döntést hozó — emberek, akik szakemberek véleményére kell, hogy hagyatkozzanak döntéseik során, mindenféle számítógép nélkül is nagyon hasznos tudnivalókat szerezhetnek a kötetből.

# ILYEN LEGYEN AZ ÚJ GÉP

## Lehetőségek és elvárások

Talán már nemsokára új „C64”-essel találkozhatunk a boltokban. Nagy a spekuláció, milyen is legyen az új gép? Elgondolkodtató volt az a kijelentés, amelyet a Commodore boss, Irving Gould tett bankszakemberek előtt. Egy C64-es utód van kilátásban. Azóta mindenki tudni vél valamit az új számítógépről, amelyet nevezünk most egyszerűen C64 III-nak. Persze működő példányt még senki sem látott. Ennek ellenére néhány részlet „kiszivárgott” már erről a csodakasztniról. Ha a tényeket és a kívánságokat összevetjük, egész használható összképet kapunk.

Az emberek több színt, nagyobb sebességet, több RAM-ot, egy 3 1/2 collos lemezegységet, RGB-monitor csatlakozást, s mindenekelőtt ergonómiailag szépen kialakított házat szeretnének. Ehhez járul még, hogy az új gép legyen 100%-ig kompatibilis a régi C64-essel. Ez utóbbit a 65SC816-os mikroprocesszor alkalmazásával el is lehet érni. Ez a chip egy 16 bites CPU, mely a 6502-es (illetve a C64-esnél alkalmazott 6518-as) mikroprocesszor parancskészletét tökéletesen „érti”. Ennek a megoldásnak az előnye, hogy azzal mindjárt nagyobb műveleti sebesség is jár. Emellett megint csak azzal járna ez, hogy támogatnánk egy új, C64 III-as, különleges képességű „üzemmódot” is. Különleges volna a 80 karakteres képi kijelzés, az egyszerre 256 szín megjeleníthetősége 320×200 pixeles fölbontás mellett. Mindez a C64 III-at egy álomkomputer szintjére emelné. Ezekkel az értékekkel állná az újjonc az Amigával és az Atari ST-vel való direkt összehasonlítási próbát is. Az egészet még azzal is javítani lehetne, hogy a meglévő videochip mellé a Btx modul grafikus chipjét is beépítenék. A C64 III ekkor tökéletes Btx terminálként is működhetne.

A C64-esnél már megismert sprite-okat természetesen az új gépben is megtaláljuk. A három szólamú monó hangból hat szólamú sztereo lehetne. Kell-e szép játékok, igényes grafikák és zenék programozásához ennél több?

Kellene még egy beépített 3 1/2 collos lemezegység. Ennek 800 kbyte-os kapacitása jelentős. Természetesen ez azt je-

lenti, hogy a C64-es lemezek nem lennének használhatók. Egy külsőleg csatlakoztatható 1541-es azonban ezt a problémát megoldja. Azt lehetne akár 8-as egységszámmal is kezelni, a másolás ellen védett programok használhatóságának érdekében. Minden programot a mellékelt másoló átvinne az új formátumra. Erre a célra már az is elég, ha kölcsönveszünk egy 1541-est. A másolás után az új floppy teljes komfortját kihasználhatjuk, például bekapcsolás utáni programboot-ot. Az új floppy persze jóval gyorsabb is lenne, mint az 1541-es.

Egyébként nemcsak arról van szó, hogy a soros port megmaradna, hanem minden más ismert illesztési lehetőség is. De a C64 III-as user portja mindjárt Centronics illesztésként is programozva lenne. Lenne azután egy új kapu is, nevezzük ezt modul portnak. Ide lehetne csatlakoztatni a memóriabővítőket, amelyek a beépített 64 kbyte-ból 1 Mbyte-ot csinálnának. Ezzel a kapacitással és a 6502-esnél szokásos assemblerrel (ellentétben az Amigával, ahol a C nyelv került a BASIC elé) hihetetlen komfortos és hasznos programok írására lenne mód. Természetesen az új gép új üzemmódja sokkal gyorsabb BASIC-et kapna, mint az eredeti. A ház formája még nem tisztázott. A PC kinézet lenne kívánatos, vagy az Amigáé. A laptop fölépítés sem volna rossz. Egy dolog viszont már ismert a C64 III-ról. Lesznek rajta német különleges jelek is (ezzel megintcsak a milliós német piac felé kacsintgatnak), valamint egy szeparált kurzor- és számblokk.

Egy ilyen számítógép esélyei, amely árban az 1000 márkás Amiga és a 300 márkás C64-es között lenne, óriásiak! Óriási a száma azoknak a géptulajdonosoknak, akik megszokott C64-es szoftvereikről és nehezen megszerzett programozói ismereteikről nem kívánnak lemondani. Ennek ellenére a C64 III egy igazi, 16 bites gép lesz, nagy sebességgel. Most már csak abban bízhatunk, hogy ez a projekt az eddigi üzletvezető, Winfried Hoffmann kiválása után sem kerül süllyesztőbe.

Ha ugyanis ezt a szupergépet mondjuk már jövőre megkaphatnánk a boltokban, az Amiga, az Atari ST és néhány PC is jó, ha fölkészül a hidegebb időkre...



## Commodore 64 — Felhasználói kézikönyv

Ára: 194 Ft  
Megjelent: 1988

A könyv a legújabb verziójú gépek nélkülözhetetlen felhasználói kézikönyve. Minden alapvető információ benne van: üzembe helyezés, BASIC nyelvű programozás stb. A Függelék a kódtáblázatokat, a hibaüzeneteket és még sok más fontos ismeretet gyűjt egybe.



# Bevezetés az MS DOS operációs rendszerbe

Sokan a Commodore PC10-es sorozatának használatán keresztül ismerték meg az MS DOS operációs rendszert. Ez az ismertetés nemcsak az ő számukra készült, hanem mindazoknak, akik szeretnének többet tudni az MS DOS-ról.

A gyakorlott felhasználó számára, aki a 64-es és 128-as gépet jól ismeri, az MS DOS félelmetes lehet, valójában azonban nagyon könnyű megérteni, ha néhány alapvető ismeretet megszerzünk róla. Először is, egyáltalán mi a DOS? A DOS szó a Disc Operating System (lemez-operációs rendszer) rövidítése. Egy operációs rendszer programok gyűjteménye, amely a számítógép működéséhez szükséges különböző feladatokat végez, például memória-kiosztást, bemeneti/kimeneti műveleteket, programok futtatását stb. Minden számítógépnek szüksége van valamilyen operációs rendszerre, hogy a külvilággal kapcsolat tudjon teremteni. A 64-es és a 128-as gépen az operációs rendszert eltakarja előlünk a BASIC, és a lemezekkel a kapcsolatot az 1541/1571/1581 jelű meghajtókban lévő ROM tartja. Az MS DOS használatakor mindez nem láthatatlan — az operációs rendszer segítségével mondjuk meg a számítógépnek, hogy mit csináljon.

Az MS DOS egyik legnagyobb előnye az, hogy a file-ok kezelése sokkal egyszerűbb, és nincs szükség hozzá további rendszerprogramokra. Például a DISC COPY utasítás egy teljes lemezt átmásol — mindehhez a 64-es vagy 128-as gépen egy másoló programra van szükség. Az MS DOS operációs rendszerrel működő számítógépek és meghajtók között párhuzamos az adatforgalom, így a bemeneti/kimeneti műveletek gyorsabbak, mint a 64-es vagy 128-as gépen. Mivel a rendszer nincs rejtve előttünk, közvetlen összeköttetésben állunk a géppel anélkül, hogy a BASIC-et megkerülő parancsokat kellene kiadnunk. (Ilyen például egy lemez formázása, ami egy parancscsatorna megnyitását igényli.)

Újabb különbség az, hogy bekapcsoláskor a 64-es és a 128-as gépben gyors rendszer-ellenőrzés történik, majd betöltődik a BASIC a READY üzenet megjelenésekor. Egy MS DOS operációs rendszerrel működő gépben bekapcsoláskor rendszerellenőrzés történik, majd egy RAM-teszt fut le, végül a gép az A jelű meghajtón megkeresi a DOS-t és lemezzel betölti. Három file-t tölt be a gép — először két rejtett rendszer-file-t, az IO.SYS és az MSDOS.SYS nevéket (ezeket az IBM gépeken IBMIO.COM és IBMDOS.COM néven használjuk), amelyek az aktuális operációs rendszert alkotják, majd egy COMMAND.COM nevű file-t, amely az MS DOS parancs-közvetítője. Ezen file-ok nélkül egy MS DOS gép egyáltalán nem tud működni.

Gyakran teszik fel a következő kérdést: „Mi a különbség az IBM DOS és az MS DOS között?” A válasz: nagyon kevés. A Microsoft Corporation 1981-ben fejlesztette ki a PC DOS-t az IBM PC számára, de fenntartotta a jogot, hogy a kompatibilis gépek számára egy másik változatot forgalmazzon. E jogot át-

engedte a PC-gyártóknak, amelyek egyike a Commodore. Az egyetlen említésre méltó alapvető különbség az, hogy az IBM a BASIC legnagyobb részét egy ROM-ban helyezi el, és a nyelv kiterjesztéseit lemezzel tölti a memóriába. A kompatibilis gépek a BASIC nyelv GWBASIC nevű változatát használják, amely minden tekintetben megegyezik az IBM BASICA nyelvvel, de teljesen lemezzel töltik a gépbe. Ezenkívül a két DOS csaknem felcserélhető. Valójában a kompatibilitás igazi próbája az, hogy egy PC-kompatibilis gép tud-e működni IBM DOS rendszerrel vagy sem (kivéve a BASIC-et). A PC10 és a COLT ezt a próbát nagyszerűen kiállja.

Egy másik, sokszor feltett kérdés a következő: „Futtatható-e a PC10 és COLT típusú gépeken a 64-es és 128-as gépekre írt program?” A válasz: nem. Ennek oka, hogy a 64-es és a 128-as gép nyolc bites és a MOS/Commodore 6502 CPU utasításkészleten alapul. Az MS DOS-ban működő gépek az Intel 8086/8088 utasításkészleten alapulnak és 16 bitesek. Mivel az utasításkészletek különbözőek, az egyik gépen futó program nem működik a másikon. Két kivétel van ez alól: egyrészt a BASIC programok átalakíthatók GWBASIC programokká, de ezt manuálisan kell megtenni, továbbá az MS DOS-ban nem lehet alakzatokat definiálni, és más a hangok előállítására is (nincs sprite és SID-chip benne). Másrészt vannak olyan rendszerprogramok, amelyekkel Commodore ASCII programok és szöveges file-ok átalakíthatók valódi ASCII-be, amely az MS DOS editorokkal és szövegszerkesztőkkel használható, de ezek a rendszerprogramok 1571 vagy 1581 típusú lemezmeghajtó használatát teszik szükségessé.

A Commodore 128 CP/M felhasználói az MS DOS-t nagyon hasonlóan fogják találni mindahhoz, amit már ismernek. A parancsok szerkezete általában hasonló, mint például az A> üzenet is a képernyőn (itt a merevlemez meghajtókat nem vesszük figyelembe). Csakúgy, mint a CP/M-ben egy futtatható program indítása a program nevének legépelésével és a RETURN lenyomásával történik. Sokan úgy fogalmazzák meg ezt, hogy a CP/M az MSDOS „nagyapája”, mert annyira hasonlítanak egymásra. Ez a hasonlóság könnyítette meg az MS DOS-ra való áttérést mind a CP/M felhasználói számára 1981–82-ben, mind a Commodore 128 CP/M felhasználói számára mostanában.

Térjünk rá a magától értetődően felvetődő kérdésre: „Mit tudunk megtenni az MS DOS segítségével?” A válasz: majdnem mindent. Az MS DOS felhasználóinak áll rendelkezésére a legnagyobb software-bázis, bármilyen gépen is dolgoznak. Egyre több szórakoztató program is készül MS DOS-ban. Az alkalmazható MS DOS programoknak már a mennyisége is elképesztő mind a kereskedelemben, mind a mindennapi életben — összefoglalva, határozottan állíthatjuk, hogy mindenki számára tartogat valamit az MS DOS.

# Tippek a C128-hoz



## A C 128-AS TÁROLÓJA

A C128-ast nem könnyű hatékonyan programozni, mert a hozzá való megfelelő szakirodalomnak igencsak híja van. Szeretnénk ezt egy kicsit alaposabban szemügyre venni.

### De mi az a nulláslap?

A nulláslap a számítógépek esetében a tárolóterület legalsó része. Azért hívják így, mert a két byte-os címezésnél a magasabb értékű byte a \$00. A nulláslap a tárolóban a \$0000—\$00FF, azaz a 0—255-ös címeket tartalmazza. A 6502-es processzorral rendelkező számítógépek esetében ez a lap azért olyan fontos, mivel vannak olyan speciális parancsok, amelyek az egy byte-os címzési lehetőséget kihasználva ezekre a tárolóterületekre hivatkoznak. Ezekre természetesen most nem tudunk kitérni.

### Mit tartalmaz a nulláslap?

A régebbi 6502-es számítógépeknél ide rakták le a legtöbb rendszerváltozót. Ide tartozik például a kurzorpozíció, vagy egy BASIC program kezdetének címe. Időközben azonban az operációs rendszerek egyre jobbak lettek, így a rendelkezésre álló 256 byte már nem volt elegendő. A C128-as esetében ezért úgy döntöttek, hogy a „nulláslap” fogalmát kiterjesztik. Ezt mi kibővített nulláslapnak fogjuk nevezni.

Kezdjük a vizsgálódást legalul. Adjuk be az alábbi utasításokat: POKE 0,111:POKE 1,51

A számítógép azonnal a beépített német (DIN) jelkészletre vált át anélkül, hogy meg kellett volna nyomni az ASCII/DIN gombot. A műveletet ráadásul ezzel a gombbal nem is lehet visszacsinálni. Hogyan történt mindez? Hát úgy, hogy a 0-ás cellába a 111-es értéket írtuk. Ezzel lekapcsoltuk az ASCII/DIN gombot. Vagyis azt hiába is nyomogatjuk. Az 1-es címre írt 51-gyel viszont azt „közöltük” a számítógéppel, hogy az adott gomb le van nyomva. A gép tehát engedelmesen a DIN jelkészletre vált. A POKE 1, 115-tel lehet visszaváltani az ASCII karakterkészletre. De az ASCII/DIN gomb továbbra sem használható. Azt csak a POKE 0,47 állítja helyre.

A 0 és az 1-es tárolócellák képezik a C128-as hardveréhez vezető kapukat. A 0 regiszter fölülírásával ezeket kifelé vagy befelé menő forgalomra állítjuk be. Az 1-es ceolánál lehet megadni, milyen értékek kiadásáról van szó, és lekérdezzük a vezetékeket. Itt lehet például a Datasette motorját vezérelni, vagy annak PLAY gombját, illetve az ASCII/DIN gombot is lekérdezni.

Hasonlóképpen érdekes funkciója van a 215-ös cellának. Adjuk be a PRINT PEEK (215) utasítást. A C128 kiad egy számot. Ha ez 128, akkor mi a 80 karakteres képernyőn dolgozunk. Ha nullát kapunk, úgy a 40 karakteres képernyő aktív. Erre mindjárt jöhet is egy ötlet: „Ha a dolog működött az amerikai és a német jelkészletekkel, akkor a két képernyő átkapcsolását is lehet valahogy manipulálni.”

Nos, az ötlet nem rossz. Írjuk a 215-ös cellába a 128-as értéket: POKE 215,128. A számítógép mindjárt a 80 karakteres képernyőre vált. A POKE 215,0 pedig értelemszerűen a 40-est hozza vissza.

Más, érdekes tárolókat a kibővített nulláslapon, a \$1000-tól találunk. Ott találjuk az F1—F8, RUN/STOP és HELP funkciók billentyű szövegeinek hosszát jelző értéket. A \$100A-tól áll a tíz funkciók billentyű szövege. Ezek törlésére az alábbi sort adjuk be:

```
FOR X=4096 TO 4105:POKE X,0:NEXT
```

Ezután hiába nyomjuk a funkciók gombokat, semmi nem történik. Ezzel egyébként az egyes funkciók billentyűket céltatosan törölhetjük. Szerézhethünk azonban meglepetést is: FOR X=4096 TO 4351:POKE X,65:NEXT

A RUN/STOP vagy a HELP funkciók gomb megnyomásakor a képernyőn egy sor betű jelenik meg. Hogy ez melyik, az az X értékétől függ. X=65-nél „a” betűket adunk ki.

## HOSSZÚ BASIC-PROGRAMOK MOZGATÁSA

A 128-as módban lehetőség van arra, hogy BASIC programokat egy meglévő RAM bővítőbe vigyük (1700/1764/1750). Erre a célra az alábbi utasítássorozat szolgál:

```
X=PEEK(4624)+256*PEEK(4625)-(PEEK(45)+PEEK(46)*256)
STASH X,7169,0,0:PRINT X
```

Azt az értéket, amelyet az X változóba raktunk le, feltétlenül meg kell jegyezni, mert a tárolt információ visszahozatala ennek segítségével történhet.

```
FETCH X, 7169,0,0
```

Ez a parancssor csak akkor működik, ha a BASIC program kikapcsolt grafikánál rövidebb mint kb. 46079 byte, grafikához lefoglalt terület esetén pedig 36863 byte. Ha a programunk ennél hosszabb, akkor egy ROM hiba miatt az 53248-as tárolócellától kezdve nem a BASIC programot, hanem a be/kiviteli tartományt mentjük ki a RAM bővítőbe. Ha megpróbáljuk ezt a „programot” visszahozni, a számítógép lemerevedik, mert a nevezett területet fölülírjuk. A mellékelt „reu debug” program (l. lista) kijavítva a C128-as ezen ROM hibáját.

A programot a BLOAD „REU DEBUG” utasítással kell betölteni, ezután már nem kell semmit csinálni. Ugyanis az a tárolóban elfoglalt helye miatt azonnal aktív. Ettől kezdve a hosszú BASIC programokat is hibamentesen vihetjük a RAM bővítőbe. Aki érti a gépi kódot, annak a magyarázatot is meg-

.003f0 sei	: az interrupt letiltása
.003f1 ldx \$ff00	: jegyezzük meg a tároló konfigurációt
.003f4 sty \$df01	: a dma controller fölülírása
.003f7 ora = \$01	: az i/o alatti ram bekapcsolása
.003f9 sta \$ff00	: a tároló konfiguráció beállítása

.003fc cli : az interrupt engedélyezése  
 .003fd jmp \$03ec : ugrás a program maradékához  
 .003ec stx \$ff00 : a régi konfigurációt visszaállítani  
 .003ef rts : vissza a főprogramba

A program trükkje az, hogy az interruptot bizonyos időre letiltjuk. Rendes körülmények között ugyanis a RAM bővítő DMA controllerje abban a pillanatban kezd dolgozni, amikor a \$DF01 regiszter fölülírása történik. Ha azonban letiltjuk az interruptot, akkor a controller csak a CLI utasítás után kezdhet neki a munkának. De addig is lehetőségünk van tetszőleges tárolókonfiguráció beállítására. Ettől kezdve a FETCH, a STASH és a SWAP parancsok hosszú programok esetében is rendesen működnek majd.

A reu debug program kijavítja a C128-as ROM egy hibáját.  
 Name: reu debug 03ec 0400

03ec : 8e 00 ff 60 78 ae 00 ff  
 03f4 : 8c 01 df 09 01 8d 00 ff  
 03fc : 58 4c ec 03 4e 01 0d 05

## PROGRAMOK ÖSSZE- HASONLÍTÁSA

Ha szeretnénk két programot összehasonlítani, akkor rendes körülmények között behívjuk a programot a gép tárolójába, majd a másodikat a DVERIFY "név"-vel, vagy VERIFY "név",8-cal leteszteljük.

Ezekkel a parancsokkal megállapíthatjuk, hogy a két program azonos-e vagy sem. Azt azonban nem találhatjuk meg, hol tér el a kettő adott esetben egymástól. A gépbe épített monitor C parancsát használva az alábbi módon végezhetjük el az összehasonlítást:

```
MONITOR
L"név1",8,1300
L "név2",8,11300
>ae
```

Az utolsó parancsnál a számítógép több hexadecimális számot ír a képernyőre. Ezek közül csak az első kettő fontos. Ha az például a 4E 36, akkor az összehasonlító parancsnak az alábbiak kell lenni:

```
C 1300 364E 11300
```

Ennek a parancsnak a beadása után kiadja a gép azokat a címeket, ahol a két program egymástól különbözik. Az M utasítással az eltérő helyeket megnézhetjük. Ha például az összevetésnél a gép a 0331F címet adta meg mint eltérő helyet, az alábbi két utasítással vethetjük össze a pozíciókat:

```
>331 F és >1331 F
```

A programjaink alaposabb összevetésének tehát nincs többé akadálya.

## MÁSFÉLE FUNKCIÓS BILLENTYŰ KIOSZTÁS

A C128-ason a funkciós billentyűket a KEY használatával könnyen átdefiniálhatjuk. A HELP és a RUN/STOP-hoz azonban már speciális program kell. A C128-as ROM listájának alaposabb tanulmányozásakor föltűnt egy rutin, ahol már nem történik meg a funkciós billentyű kódjának (számának) ellenőrzése. Így könnyen átírhatjuk a nevezett két billentyűt is. A fölhívás:

```
BANK 15 : SYS 24812,,X,,,a$
```

Az X változó adja meg a funkciós billentyű számát, az A\$ pedig a kívánt parancsszekvenciát. A mellékelt táblázat megadja a kódszám szerinti hozzárendelést:

F1	0	F5	4
F2	1	F6	5
F3	2	F7	6
F4	3	F8	7
F9	8 (run/stop)		
F10	9 (help)		

## SZÖVEG- NYOMTATÁS A TÁROLÓBÓL

A BASIC 7.0 BSAVE utasítása nemcsak egy tárolóterület lemezre mentésére szolgál. Ha például egy ASCII szöveg áll a gép tárolójában, azt előbb lemezre szoktuk írni a lentebb álló két módon, majd egy szövegszerkesztőt elővéve kinyomtattuk:

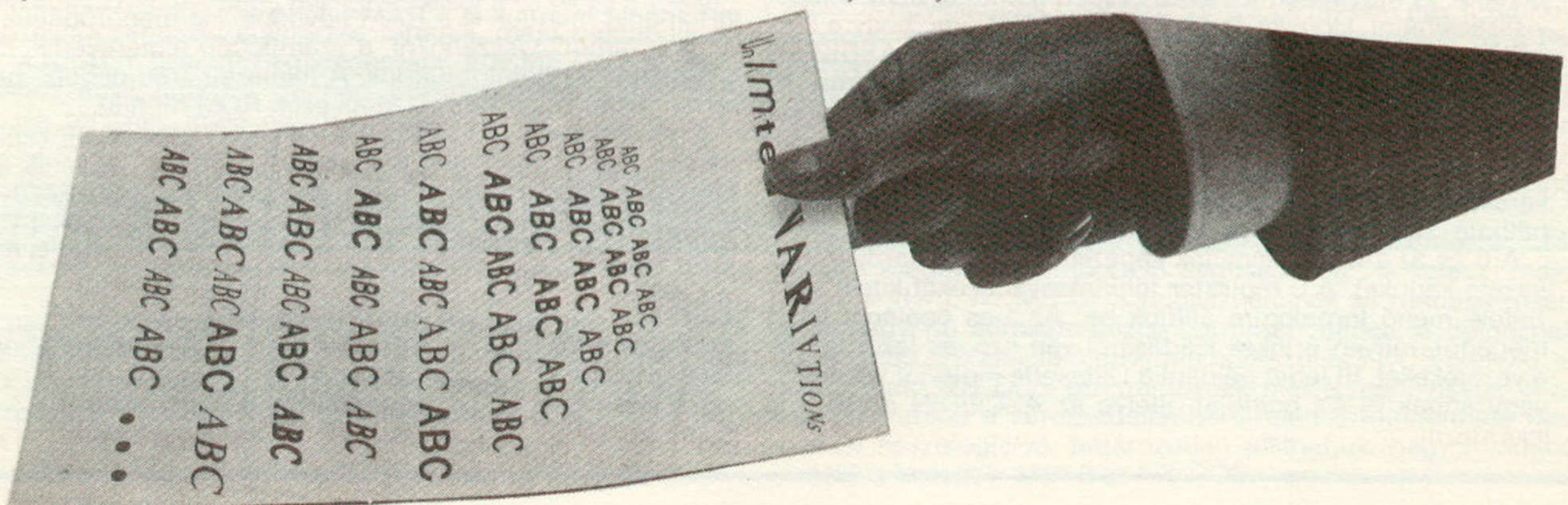
```
BSAVE "név" ON B0,P kezdet T0 vég vagy
```

```
S "név", 8, kezdet,vég
```

A dolog azonban megy egyszerűbben is. Tároljuk a szöveget egyszerűen a nyomtatóra! A dolog sokkal könnyebb, mint gondolnánk:

```
OPEN 4,4 : BSAVE "név",U4,On B0,Pkezdet TOPvég : CLOSE4
```

Ilyenkor az állománynevet is kinyomtattuk. Epson kompatibilis nyomtatóknál ezeknek a jeleknek a kiadását elnyomhatjuk, ha első karakterként a \$18-as (decimálisan a 24-es) értéket küldjük a nyomtatóra.



# EGYESÜLETI TAGOK FÓRUMA

## C 64

Eladó C64 + DATASETTE + 2 darab JOYSTICK + 500 program + szakkönyvek, 20 000 forintért. Mátyásiné, 13-30-560/177.

C64-re keresek megvételre (OXFORD-) PASCAL fordítót, ASSEMBLERT és monitor-programot. Szokop Imre, 1146 Budapest, Abonyi u. 14. Telefon: 14-27-652.

C64 + OC118N DRIVE + DATASETTE + 38 darab lemez, színvonalas játék- és felhasználói programok (150 darab) sürgősen eladó. Szabó Zoltán, 9028 Győr, Erfurt u. 30. III/12.

Eladó: újszerű állapotban C64 + 1541-II DRIVE + 1530 magnó + bőséges szakirodalom + FINAL CARTRIDGE + JOYSTICK + 15 kazetta + 80 lemez. Tóth Patony, 7150 Bonyhád, Perczel Mór u. 7.

Kezdő angol oktatóprogramot keresek kazettára. Ezenkívül érdekel bármely oktatóprogram C64-re. Brád György, 5641 Tarhos, Békési út 7/A.

C64 játék- és felhasználói programokat cserélek kazettán és lemezen. Listát kérek és küldök is. Szászvárosi Zsolt, 6725 Szeged, Tabán u. 10. Telefon: (62)-30-572.

C64 programokat cserélek kazettán. A válaszokat listával kérem. 600 darab programom van. Minden levélre válaszlok. Lázár Miklós, 4220 Hajdúböszörmény, Rudas László utca 11.

C64-re kazettán és lemezen 2200 játék- és felhasználói programból cserélek, eladok (10–20 Ft). Csere esetén listát kérek és küldök. Németh András, 9081 Győrújbarát, Veres Péter u. 23.

C64-hez 7 funkciós NEW LINE CARTRIDGE letilthatatlan RESET gombbal, reális áron eladó. Piller Krisztián, 1122 Budapest, Krisztina krt. 2/4. Telefon: 15-63-006.

Eladó új C64 + 1541-II drive + 2 darab joystick + játékmodul 45 000 forintért. Kérésre programokat is másolok. Szabó Zoltán, 9028 Győr, Erfurt u. 30.

Keresem a Gyűrűk ura programot C64-re. Török Zsolt, 18-31-111.

C64 programokat cserélek. Listát kérek és küldök. Csabai Kálmán, 6090 Kunszentmiklós, Kecskeméti út 11/A.

C64 + DRIVE + magnó + JOYSTICK + 130 lemez olcsón eladó együtt. Ugyanott programcsere lemezen. Csizmazia Ferenc, 9023 Győr, Bacsó Béla u. 87.

## PLUS/4

PLUS/4 játék-, felhasználói és oktatóprogramokat cserélek lemezen. Kálmán Albert, 3300 Eger, Rákóczi út 31. III/11. Telefon: (hétfőig 7–19 óra között) (36)-25-648.

PLUS 4 programokat cserélünk lemezen vagy kazettán. Listát kérünk. Cserealap 700 darab program. Sós Csaba & Attila, 6800 Hódmezővásárhely, Bajza u. 16.

PLUS/4 programokat cserélek kazettán! Keresem a SAVAGE ISLAND térképet! Listát kérek! Kiskovács Endre, 6090 Kunszentmiklós, Gaál Sándor út 39.

Plus/4 programokat cserélek kazettán. Listát kérek és küldök. Keresem a TIR NA NOG, ALIENS, MERCENARY II. című játékprogramokat. Domány István, 3200 Gyöngyös, Warga L. u. 11. I/5.

1000 darab program a cserealap. PLUS/4-es programcsere kazettán és lemezen. Solymosi Béla, 1204 Budapest, Dessewffy utca 41.

PLUS/4 programcsere! Listát kérek! Keresem: HULK, ON CUE, EUROPEAN GAMES, BALANCE című programokat. Kiskovács Endre, 6090 Kunszentmiklós, Gaál Sándor út 39.

## VEGYES

Számítógépek javítását vállalom. Török István, 1156 Nádastó u. 98. CB: AM 5 VIK/1. Telefon: (munkaidőben) 15-31-933/3184.

AMIGA tulajdonostársak jelentkezését várom. Kopácsy Vilmos. Telefon: 18-11-550/147 munkahely, 14-50-358 otthon.

Használt, jó állapotú JOYSTICK-et vennék. Keresem a STREET POKER című programot. Fehérvári Kornél, 2635 Vámosmikola, Ipoly út 17.

COMMODORE 1802-C színes monitor eladó. 64-er sonderhefteket cserélnék: adok 2., 3. játékhoz, kérek 1., 10., 22. C 128-hoz, vagy bármely egyéb felhasználóit C64/128-hoz. Korompay Attila, telefon: 14-86-165, 11-47-539.

Keresem a PROFI ASS V. 2.0 programot. Cserébe színvonalas 87-88-89-es programokat adok. Ugyanitt játékcseré (ez kazettán)! Császár Károly, 9028 Győr, Jereváni út 37. IV/4.

Eladó új CITIZEN 120D grafikus printer, C64 illesztővel, garanciális Fisher FVH-P20S indexkereséses video, SVHS vagy Panasonic NV-F70EG. Áregyeztetéses csere is érdekel. Ács Károly, 8111 Seregélyes, Münnich Ferenc u. 70.

Eladó egy alig használt BROTHER típusú nyomtató, sok programmal. Szatmár Tibor, 5830 Battonya, Lenin út 87.

COMMODORE AMIGA olcsón eladó. Érdeklődni lehet: 06/92/12 145/130 mellék (munkaidőben), Farkas András 3487.

# JÁTÉKPIACI ÚDONSÁGOK



## ITT A SPHERICAL!

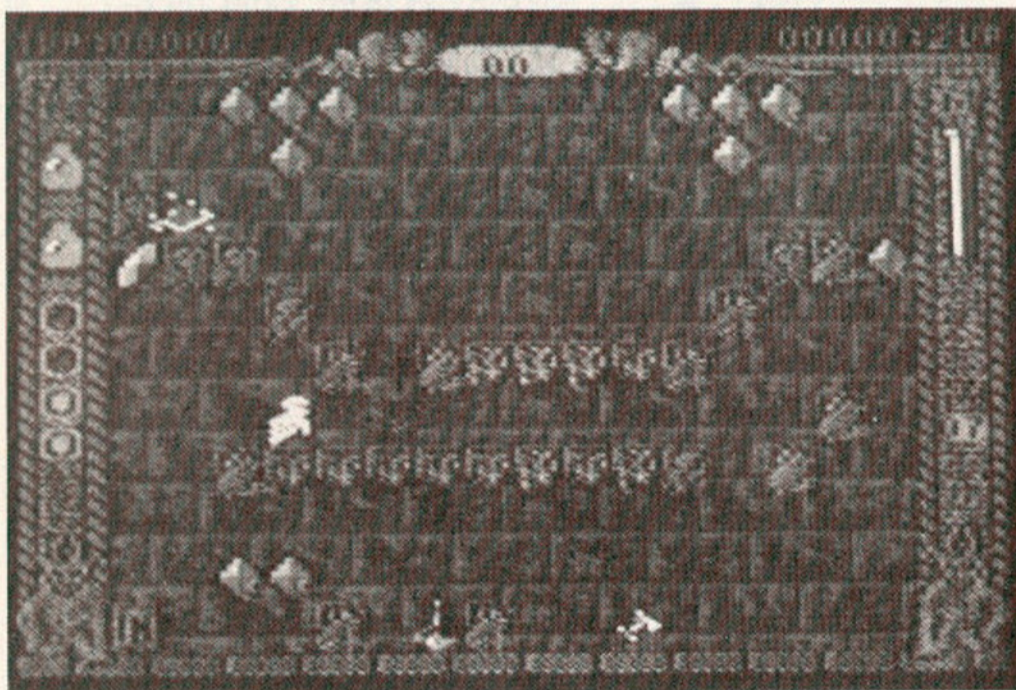
A Rainbow Arts korábban beharangozott játéka a Spherical, most végre megjelent. Ahogy azt vártuk, a nagyobb gépekről a C64-re készült átirat kiválóra sikerült. A grafika és a zene a legjobbak közül való.

Info.: Rushware, Bruchweg 128-132, D-4044 Kaarst 2.

kategóriájában a kitüntető „Év Szoftvere” díjat is, megverve sokkal rangosabb és komplexebb játékokat!

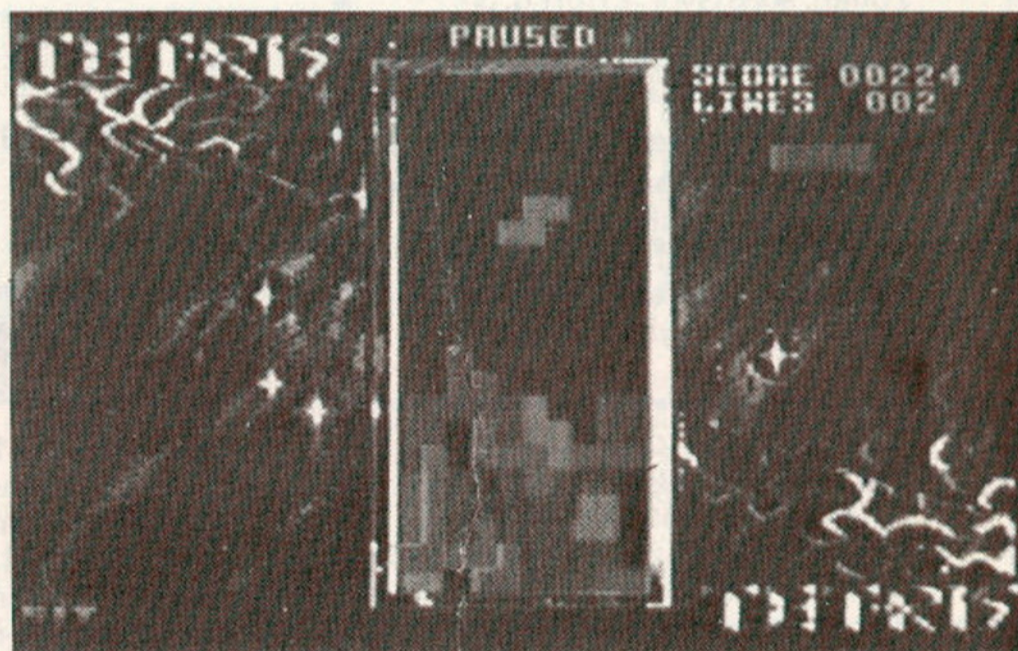
A Tetris zenéje jó, a grafikus megvalósítás kiváló. Akik szeretik teljesítményüket viszontlátni, azoknak egy nem tárolható highscore lista is a rendelkezésre áll.

Info.: Virgin Mastertronic Limited, 120 Campden Hill Road, London WB 7AR, Nagybritannia



## A TETRIS A C 64-RE

A Tetris programmal a brit szoftvergyártó, a Mastertronic az első és eddig egyetlen olyan programot dobta piacra, amely eredetileg szovjet programozóktól származik. Ebben az ügyességi játékokban az a cél, hogy a puzzle szerű blokk-elemekből összeállítsunk egy lyukaktól mentes sort. Ha ez sikerül, a sor eltűnik, a helyére újat rakhatunk. De nemcsak ügyes kézre, gyors döntésekre is szükség van a játékban. Ha ugyanis hibázunk, a lyukas sor megmarad, helyet foglal el, egyre magasabbra kerül, míg végül nem marad több hely, s a játéknak vége. Egy hiba után hamar jön a másik, s egyre nehezebb és nehezebb lesz ezeket korrigálni. A játékról egyébként tudni kell, hogy PC változatban került be először a köz tudatba, s mint ilyen a Chip Magazin által indított versengésben a maga egyszerűsége mellett egyszer mégis elnyerte a saját



## ÚJ KOSÁRLABDA-SZIMULÁCIÓ

Az Advanced Basketball Simulator a Mastertronic cég egy újabb sportszimulációja. Ez a játék igen vicces menüvezetéssel rendelkezik. Egy csarnokban vagyunk, ahol különböző tárgyak választásával különféle beállításokat tehetünk. Van egy videorekorder, amellyel a megtörtént eseményeket vissza lehet játszani. Különböző ajtók kiválasztásával gyakorló üzemmódba léphetünk, vagy változtathatjuk a játékerőt és a játékosok számát. Maga a szimuláció is igen jól sikerült, abban humoros elemek is visszaköszönnek. A nem éppen meggyőző zenét a kiváló minőségű grafika és az igen szép térbeli kép-görgetés felejteti.

Info.: Virgin Mastertronic Limited, 120 Campden Hill Road London W8 7AR, Nagybritannia

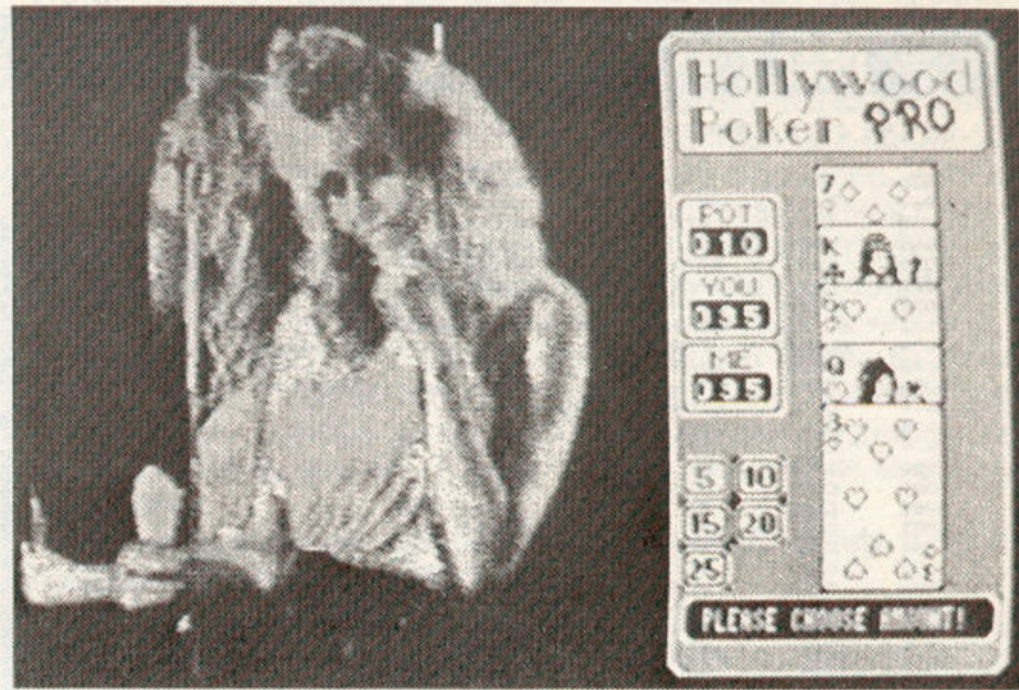


# PIKÁNS PÓKER

A Reline cég most mutatta be a C64-esre is megírt Hollywood Poker Pro programját. Ez a játék a klasszikusnak számító Strip Poker új változata. Mint ismeretes, abban a játékban a pénzen kívül más dolgokat is elfogadunk tétnek...

A Hollywood Poker Pro játékot azonban nemcsak a klassz intro és a szép grafikák teszik élvezetessé (most először találkozhatunk egy C64-es játékban az Amigánál ismert interlacing eljárással, amelyet a grafika fölbontás optikai megkettőzéséhez használnak), hanem ezenkívül egy igen intelligens pókeralgoritmussal is meg kell küzdenünk. Márpedig ez az algoritmus nemcsak hogy keményen játszik, de tanul is az ellenféltől. A „pókerpofára” és a jól időzített blöffökre szükség van tehát.

Info.: Reline Software GmbH, Königstr. 55. D — 3000 Hannover 1.



## JÁTÉKTIPPEK

### TRANTOR

A Trantor játék hősének végtelen energiát kölcsönözhetünk. Keressünk egy olyan szekrényt, amelyben hamburger van! Ha megvan, nyomjuk a joysticket lefelé, és gépeljük be a "PROBE" betűket.

### AFTERBRUNNER

Nyomjuk meg a játékban a címképnél kb. százszor a szünet-gombot! Ez aktiválja a continue-funkciót a 17. szintig. Azaz, előre a joystickkel, és tűz!!!

### DANGER FREAK

Ha a játékban beadjuk a "17,04.70" dátumot, a gyakorló üzemmódba jutunk.

### SPINDIZZY

Végtelen játékidőt nyerhetünk, ha a betöltés után közvetlenül begépeljük a "PAT" szót.

### VINDICATOR

A játék kódszavai a highscore listában állnak. A jelszó: "ENOLAGAY".

# FÉNYPONT

## VIDEO CENTER

ADÁS VÉTEL KÖLCSÖNZÉS SZERVIZ  
7624 PÉCS, RÓKUS U.3/A TEL.: 33-612  
CB 33-844/3 USB 36 CS.



# Tippek Trükkök



## EGÉR ANIMÁCIÓ

A program segítségével tetszőleges formára átírhatjuk az egér alakját, illetve a 8 fázisrajz megrajzolásával animációt végezhetünk vele. Az animálás sebességét is változtathatjuk, programunkba ezt az F10 billentyűvel csinálhatjuk.

A programot SEKA assembler fordítóba írjuk be, majd lefordítás után futtassuk le.

A program leírása:

A 14-től 21 sorig a rom rutinok ugráscímét adjuk meg. 23-tól 25-ig szubrutinok hívása, majd következik a rendszer nevű szubrutin, amelyben a Library-kat (könyvtárakat) nyitjuk meg. 28–31 grafik

Library, 33–36 intúció Library, 38–41 dos Library, 43–44 sorokban megállapítjuk az aktuális ablakot, és a 46–49-ben megnyitjuk. Ezután következik a Sprite nevű szubrutin. (Mint tudjuk, az egér is sprite-ként van megjelenítve.) A Sprite nevű szubrutinba ennek a sprite-nak keverjük ki a színeit és állítjuk be a különböző mutatóit. Az 53. sorban az aktuális ablakot adjuk meg, ahol az egérnek meg kell majd jelennie. Az 54. sorban a grafikus Library báziscímét adjuk meg (mivel a SetRGB4 nevű rom rutin a grafikus könyvtárban van), az 55. sorban egy indexet adunk meg a színpalettára vonatkozóan. Az 56–58. sorokba keverjük a 3 alapszínből (RGB) a 17-es paletta színét, 56. piros, 57. zöld, 58. kék, majd az 59-es sorban meghívjuk a SetRGB4 nevű rom rutint, amely beállítja a megfelelő regisztereket. Ugyanez történik a 60–65-ig és a 66–71-ig sorokban, csak a 18-as és a 19-es palettákkal.

A következő szubrutinunk a Forgasas nevű, amely a sprite megjelenítését és animálását végzi.

A SetPointer nevű rom rutin, melyet majd meghívunk az intúció könyvtárban (Library) van. Ezért a 74-es sorban megadjuk az intúció könyvtár báziscímét. A 75-ös sorban az aktuális ablakot, ahol az egér meg fog jelenni, a 76-os sorban a 0. fázis beolvasása történik, a 77-es sorban a sprite (egér) X-irányú méretét, a 78-asban az Y-irányút határozzuk meg. A 79–80-as sorokban a megjelenítés X és Y irányú koordinátáját adjuk meg. Majd a 81-es sorban meghívjuk a SetPointer nevű rom rutint, amely átírja az egér alakját és beállítja a különböző regisztereket és mutatókat. A 82-es sorban elküldi egy várakozó szubrutinra, a 83-ban megvizsgálja, hogy a bal oldali egérgomb le van-e nyomva, ha igen, akkor vége, ha nem, akkor ugrik a következő fázis beolvasására. Ezeket a műveleteket végrehajtja mind a nyolc fázisnál.

A várakozó szubrutin a 171-es sortól található. A 171-estől a 181-esig megvizsgálja, hogy az F10 billentyű le van-e nyomva, ha igen, akkor megváltoztatja az animáció sebességét és továbbmegy a várakozó rutinra. A 182-es sorban meghatározza a Dos könyvtár (Library) báziscímét, mert a Delay nevű rom rutin ebben a könyvtárban található. A 183-asban beállítja a várakozás idejét (s), és a 184-esben meghívja a Delay nevű várakozó rom rutint. 185–188-ig vár kétszer a következő képfelépítésig és továbbmegy. A 191–200-ig tartó „vége” szubrutinban lezárja a megnyitott könyvtárakat (Library) és visszaadja a vezérlést.

```

;*****
;
;      Eger animacio
;
;      Irta: Biro Zsolt es Bakos Gabor
;
;      C= Ujsag 1989
;*****
; Futtatas utan a kurzornyil kormozgast vegez, amely az F-10 b
; llentyu
; lenyomasaval felgyorsithato illetve lelassithato.
; A baloldali egérgombbal a program leallithato.
;*****

```

```

OpenLibrary = -30 -378
CloseLibrary = -$19e
ViewPortAdr = -30 -270
SetPointer = -30 -240
SetRGB4 = -30 -258
WaitTOF = -30 -240
Delay = -30 -168
ExecBase = 4

```

```

jsr Rendsz
jsr Sprite
jsr Forgasas
rts

```

```

Rendsz: move.l ExecBase, a6
        lea.l GfxName, a1
        jsr OpenLibrary(a6)
        move.l d0, GfxBase

```

AMIGA

```

        move.l ExecBase, a6
        lea.l IntName, a1
        jsr OpenLibrary(a6)
        move.l d0, IntBase

        move.l ExecBase, a6
        lea.l DosName, a1
        jsr OpenLibrary(a6)
        move.l d0, DosBase

        move.l IntBase, a0
        move.l 52(a0), AktWin
        move.l IntBase, a6
        move.l AktWin, a0
        jsr ViewPortAdr(a6)
        move.l d0, AktVP
        rts

```

```

Sprite:
move.l AktVp,a0
move.l GfxBase,a6
move.l #17,d0
move.l #0,d1
move.l #0,d2
move.l #0,d3
jsr SetRGB4(a6)
move.l AktVP,a0
move.l #18,d0
move.l #13,d1
move.l #2,d2
move.l #2,d3
jsr SetRGB4(a6)
move.l AktVP,a0
move.l #19,d0
move.l #15,d1
move.l #12,d2
move.l #12,d3
jsr SetRGB4(a6)
rts
Forgatas:
move.l IntBase,a6
move.l AktWin,a0
lea.l Image0,a1
move.l #14,d0
move.l #16,d1
move.l #-7,d2
move.l #-7,d3
jsr SetPointer(a6)
jsr Var
btst #6,$bfe001
bne cont1
bra vege
cont1: move.l IntBase,a6
move.l AktWin,a0
lea.l Image1,a1
move.l #14,d0
move.l #16,d1
move.l #-6,d2
move.l #-8,d3
jsr SetPointer(a6)
jsr Var
btst #6,$bfe001
bne cont2
bra vege
cont2: move.l IntBase,a6
move.l AktWin,a0
lea.l Image2,a1
move.l #14,d0
move.l #16,d1
move.l #-5,d2
move.l #-7,d3
jsr SetPointer(a6)
jsr Var
btst #6,$bfe001
bne cont3
bra vege
cont3: move.l IntBase,a6
move.l AktWin,a0
lea.l Image3,a1
move.l #14,d0
move.l #16,d1
move.l #-4,d2
move.l #-6,d3
jsr SetPointer(a6)
jsr Var
btst #6,$bfe001
bne cont4
bra vege
cont4: move.l IntBase,a6
move.l AktWin,a0
lea.l Image4,a1
move.l #14,d0
move.l #16,d1
move.l #-5,d2
move.l #-5,d3
jsr SetPointer(a6)
jsr Var
btst #6,$bfe001
bne cont5
bra vege

```

```

cont5: move.l IntBase,a6
move.l AktWin,a0
lea.l Image5,a1
move.l #14,d0
move.l #16,d1
move.l #-6,d2
move.l #-4,d3
jsr SetPointer(a6)
jsr Var
btst #6,$bfe001
bne cont6
bra vege
cont6: move.l IntBase,a6
move.l AktWin,a0
lea.l Image6,a1
move.l #14,d0
move.l #16,d1
move.l #-7,d2
move.l #-5,d3
jsr SetPointer(a6)
jsr Var
btst #6,$bfe001
bne cont7
bra vege
cont7: move.l IntBase,a6
move.l AktWin,a0
lea.l Image7,a1
move.l #14,d0
move.l #16,d1
move.l #-8,d2
move.l #-6,d3
jsr SetPointer(a6)
jsr Var
btst #6,$bfe001
bne Forgatas
bra vege

Var: move.b $bfec01,d0
not.b d0
ror.b #1,d0
cmpi.b #$59,d0
bne tovabb
move.l seb,d0
cmpi.b #$0,d0
bne gyors
move.l #$2,seb
jmp tovabb
gyors: move.l #$0,seb
tovabb: move.l DosBase,a6
move.l seb,d1
jsr Delay(a6)
move.l GfxBase,a6
jsr WaitTOF(a6)
move.l GfxBase,a6
jsr WaitTOF(a6)
rts

vege: move.l ExecBase,a6
move.l GfxBase,a1
jsr CloseLibrary(a6)

move.l IntBase,a1
jsr CloseLibrary(a6)

move.l DosBase,a1
jsr CloseLibrary(a6)
rts
even
GfxName:dc.b 'graphics.library',0
IntName:dc.b 'intuition.library',0
DosName:dc.b 'dos.library',0
even
GfxBase:blk.l 1,0
IntBase:blk.l 1,0
DosBase:blk.l 1,0
AktWin: blk.l 1,0
AktVP: blk.l 1,0
seb: blk.l 1,2

```

even  
 Image0: dc.1 0  
 dc.w %0000000000000000,%0000000000000000  
 dc.w %0111111000000000,%0000000000000000  
 dc.w %0111111000000000,%0011111000000000  
 dc.w %0100011000000000,%0011111000000000  
 dc.w %0100011000000000,%0011110000000000  
 dc.w %0100011000000000,%0011111000000000  
 dc.w %0100100110000000,%0011011100000000  
 dc.w %0011010011000000,%0000001110000000  
 dc.w %0000001001100000,%0000000111000000  
 dc.w %0000000100110000,%0000000011100000  
 dc.w %0000000010100000,%0000000001000000  
 dc.w %0000000001000000,%0000000000000000  
 dc.w %0000000000000000,%0000000000000000  
 dc.w %0000000000000000,%0000000000000000  
 dc.1 0

even  
 Image1: dc.1 0  
 dc.w %0000001100000000,%0000000000000000  
 dc.w %0000010110000000,%0000000110000000  
 dc.w %0000100011000000,%0000001111000000  
 dc.w %0001000001100000,%0000111111000000  
 dc.w %0001000111100000,%0000111111000000  
 dc.w %0000110111000000,%0000000110000000  
 dc.w %0000010110000000,%0000000110000000  
 dc.w %0000010110000000,%0000000110000000  
 dc.w %0000010110000000,%0000000110000000  
 dc.w %0000010110000000,%0000000110000000  
 dc.w %0000010110000000,%0000000110000000  
 dc.w %0000010110000000,%0000000110000000  
 dc.w %0000010110000000,%0000000110000000  
 dc.w %0000011110000000,%0000000000000000  
 dc.w %0000000000000000,%0000000000000000  
 dc.1 0

even  
 Image2: dc.1 0  
 dc.w %0000000000000000,%0000000000000000  
 dc.w %0000000111111000,%0000000000000000  
 dc.w %0000001000011000,%0000000011111000  
 dc.w %0000001000011000,%0000000011111000  
 dc.w %0000000100011000,%0000000011110000  
 dc.w %0000001001011000,%0000000011111000  
 dc.w %0000010011111000,%0000000111011000  
 dc.w %0000100110110000,%0000001110000000  
 dc.w %0001001100000000,%0000111000000000  
 dc.w %0010011000000000,%0001110000000000  
 dc.w %0001110000000000,%0000100000000000  
 dc.w %0000100000000000,%0000000000000000  
 dc.w %0000000000000000,%0000000000000000  
 dc.w %0000000000000000,%0000000000000000  
 dc.1 0

even  
 Image3: dc.1 0  
 dc.w %0000000000000000,%0000000000000000  
 dc.w %0000000000000000,%0000000000000000  
 dc.w %0000000000000000,%0000000000000000  
 dc.w %0000000001100000,%0000000000000000  
 dc.w %0000000010010000,%0000000001100000  
 dc.w %0111111110001000,%0000000001110000  
 dc.w %01000000000000100,%0011111111111000  
 dc.w %0111111111001100,%0011111111111000  
 dc.w %0111111111011000,%0000000001110000  
 dc.w %0000000011110000,%0000000001100000  
 dc.w %0000000001100000,%0000000000000000  
 dc.w %0000000000000000,%0000000000000000  
 dc.w %0000000000000000,%0000000000000000  
 dc.w %0000000000000000,%0000000000000000  
 dc.1 0

even  
 Image4: dc.1 0  
 dc.w %0000000000000000,%0000000000000000  
 dc.w %0000000000000000,%0000000000000000  
 dc.w %0000100000000000,%0000000000000000  
 dc.w %0001010000000000,%0000100000000000  
 dc.w %0011001000000000,%0001110000000000  
 dc.w %0001100100000000,%0000111000000000  
 dc.w %0000110010110000,%0000011100000000  
 dc.w %0000011001001000,%0000001110110000  
 dc.w %0000001100001000,%0000000111110000  
 dc.w %0000000110001000,%0000000011110000  
 dc.w %00000001100001000,%0000000111110000  
 dc.w %0000000111111000,%0000000111110000  
 dc.w %0000000111111000,%0000000000000000  
 dc.w %0000000000000000,%0000000000000000  
 dc.1 0

even  
 Image5: dc.1 0  
 dc.w %0000000000000000,%0000000000000000  
 dc.w %0000011110000000,%0000000000000000  
 dc.w %0000011010000000,%0000000110000000  
 dc.w %0000011010000000,%0000000110000000  
 dc.w %0000011010000000,%0000000110000000  
 dc.w %0000011010000000,%0000000110000000  
 dc.w %0000011010000000,%0000000110000000  
 dc.w %0000011010000000,%0000000110000000  
 dc.w %0000011010000000,%0000000110000000  
 dc.w %0000011010000000,%0000000110000000  
 dc.w %0000111011000000,%0000000110000000  
 dc.w %0001111000100000,%0000111111000000  
 dc.w %0001100000100000,%0000111111000000  
 dc.w %0000110001000000,%0000001111000000  
 dc.w %0000011010000000,%0000000110000000  
 dc.w %0000001100000000,%0000000000000000  
 dc.1 0

even  
 Image6: dc.1 0  
 dc.w %0000000000000000,%0000000000000000  
 dc.w %0000000000000000,%0000000000000000  
 dc.w %0000000001000000,%0000000000000000  
 dc.w %0000000011100000,%0000000001000000  
 dc.w %00000000110010000,%0000000011100000  
 dc.w %0000001100100000,%0000000011100000  
 dc.w %0011011001000000,%0000000111000000  
 dc.w %0111110010000000,%0011011100000000  
 dc.w %0110100100000000,%0011111000000000  
 dc.w %0110001000000000,%0011110000000000  
 dc.w %0110000100000000,%0011111000000000  
 dc.w %0110000100000000,%0011111000000000  
 dc.w %0111111000000000,%0000000000000000  
 dc.w %0000000000000000,%0000000000000000  
 dc.1 0

even  
 Image7: dc.1 0  
 dc.w %0000000000000000,%0000000000000000  
 dc.w %0000000000000000,%0000000000000000  
 dc.w %0000000000000000,%0000000000000000  
 dc.w %0001100000000000,%0000000000000000  
 dc.w %0011110000000000,%0001100000000000  
 dc.w %0110111111111000,%0011100000000000  
 dc.w %1100111111111000,%011111111110000  
 dc.w %10000000000001000,%011111111110000  
 dc.w %0100011111111000,%0011100000000000  
 dc.w %0010010000000000,%0001100000000000  
 dc.w %0001100000000000,%0000000000000000  
 dc.w %0000000000000000,%0000000000000000  
 dc.w %0000000000000000,%0000000000000000  
 dc.w %0000000000000000,%0000000000000000  
 dc.1 0





## LÉPEGETŐ

Az alábbi program a C/+4 (C-16) gépi kódú programjait (pl. az interpreter rutinokat) utasításonként, egyeslépés üzemmódban hajtja végre. Eközben a képernyő alsó két sorában láthatók: a PC jelenlegi értéke (a soron következő utasításra mutat), az éppen végrehajtott utasítás PC értéke, az utasítás eredményeként előállított regiszterértékek, továbbá két tetszőleges memóriacím pillanatnyi tartalma.

A SHIFT billentyű lenyomására a vizsgált program egy utasításnyit lép előre. A SHIFT LOCK lenyomásával az előrelépés szemmel követhető lassúsággal, folyamatosan történik. Egy példa: a BASIC melegstart rutinja ekkor több mint egy óráig tart.

Ha a 3-as billentyűt megnyomjuk (nyíl jelenik meg), a végrehajtás felgyorsul, újabb lenyomásra (a nyíl villog) még inkább felgyorsul — ekkor pl. az előbbi rutin kb. 5 perc alatt zajlik le.

A programot \$0E60-tól \$0FFF-ig kell bevinni a képernyőmemória területére (ESC + B használata!), és innen kell kimenteni MONITOR üzemmódban.

A \$0EA8...\$0EEF és a \$0FCB...\$0FCF között ASCII — szöveg van, ezt M paranccsal célszerű bevinni.

Újrabetöltés (MONITORBAN) után a G 0E60 paranccsal indítható. Ekkor először széthelyezi önmagát a rendszer — RAM szabad területeire (\$060C...\$06EB és \$0BE8...\$0BFF és \$0FE8...\$0FFF), majd a képernyőn a következő üzenet jelenik meg:

```

BYTES LO HI
>0626 00 00...
>0629 00 00...
KEY?= 11 DB
>063A FF 0F...
START LO HI
>06E5 03 87...
G 06A8

```

A LO — HI alá beírható az a két memóriacím, amelyet figyelni akarunk. Ha olyan rutint vizsgálunk, amely a billentyűzetről adatot vár, akkor az FF0F helyére 11 DB-t kell írni. A vizsgálandó rutin kezdőcímét a START LO — HI alá kell beírni (ez jelenleg \$8703 = READY).

Végül G 06A8 paranccsal elindul az egyes — lépés program. Leállítani RUN/STOP + RESET segítségével lehet.

Ezután a HELP lenyomására újra az előbbi üzenet jelenik meg a képernyőn, és a címeket újra módosítani lehet.

A program \$ 1000...8000 között RAM-ban lévő rutinokat futtat, \$ 8000 felett pedig a ROM rutinjait.

Ha a vizsgált programban SEI van, akkor a következő CLI utasításig a program a gép eredeti sebességével hajtódik végre.

A disassembler-listát mellékelem.

MARGITAI JÁNOS

	FC	SR	AC	XR	YR	SP	
MONITOR							199
B000	00	00	00	00	F8		
0E60	78				SEI		
0E61	A2	F7			LDX ##F7		
0E63	A0	17			LDY ##17		
0E65	BD	F0	0E		LDA \$0EF0,X		
0E68	99	E8	0B		STA \$0BE8,Y		
0E6B	CA				DEX		
0E6C	88				DEY		
0E6D	10	F6			BPL \$0E65		
0E6F	EA				NOP		
0E70	A0	E0			LDY ##E0		
0E72	BD	F0	0E		LDA \$0EF0,X		
0E75	99	0B	06		STA \$060B,Y		
0E78	CA				DEX		
0E79	88				DEY		
0E7A	D0	F6			BNE \$0E72		
0E7C	EA				NOP		
0E7D	EA				NOP		
0E7E	EA				NOP		
0E7F	EA				NOP		
0E80	A9	0E			LDA ##0E		
0E82	85	23			STA \$23		
0E84	A9	A8			LDA ##A8		
0E86	85	22			STA \$22		
0E88	A9	07			LDA ##07		
0E8A	85	76			STA \$76		
0E8C	A9	46			LDA ##46		
0E8E	20	C2	B7		JSR \$B7C2		
0E91	EA				NOP		
0E92	EA				NOP		
0E93	EA				NOP		
0E94	A2	07			LDX ##07		
0E96	20	2C	DC		JSR \$DC2C		
0E99	EA				NOP		
0E9A	20	70	DE		JSR \$DE70		
0E9D	EA				NOP		
0E9E	EA				NOP		
0E9F	EA				NOP		
0EA0	58				CLI		
0EA1	4C	45	F4		JMP \$F445		
0EA4	00				BRK		
0EA5	00				BRK		
0EA6	00				BRK		
0EA7	00				BRK		
0EAB	93				???		
0EA9	11	11			ORA (\$11),Y		
0EAB	42				???		
0EAC	59	54	45		EOR \$4554,Y		
0EAF	53				???		
0EB0	20	4C	4F		JSR \$4F4C		
0EB3	20	48	49		JSR \$4948		
0EB6	8D	3E	30		STA \$303E		
0EB9	36	32			ROL \$32,X		
0EBB	36	0D			ROL \$0D,X		

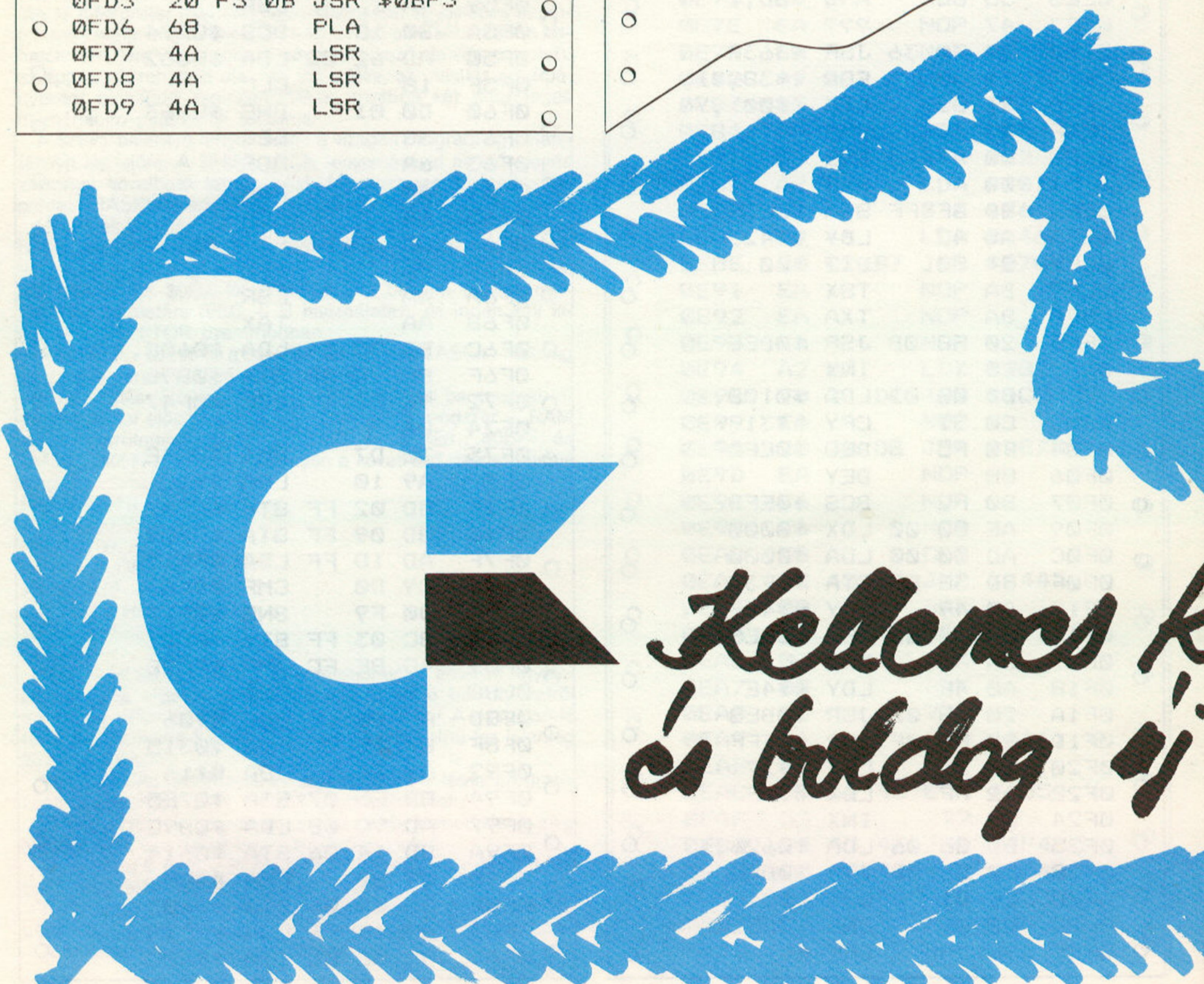


○	0EBD	3E	30	36	ROL	\$3630, X	○
	0EC0	32			???		
○	0EC1	39	0D	4B	AND	\$4B0D, Y	○
	0EC4	45	59		EOR	\$59	
○	0EC6	3F			???		○
	0EC7	3D	3E	31	AND	\$313E, X	
○	0ECA	31	20		AND	(\$20), Y	○
	0ECC	44			???		
○	0ECD	42			???		○
	0ECE	8D	3E	30	STA	\$303E	
○	0ED1	36	33		ROL	\$33, X	○
	0ED3	41	0D		EOR	(\$0D, X)	
○	0ED5	53			???		○
	0ED6	54			???		
○	0ED7	41	52		EOR	(\$52, X)	○
	0ED9	54			???		
○	0EDA	20	4C	4F	JSR	\$4F4C	○
	0EDD	20	48	49	JSR	\$4948	
○	0EE0	8D	3E	30	STA	\$303E	○
	0EE3	36	45		ROL	\$45, X	
○	0EE5	35	0D		AND	\$0D, X	○
	0EE7	47			???		
○	0EE8	20	30	36	JSR	\$3630	○
	0EEB	41	38		EOR	(\$38, X)	
○	0EED	91	00		STA	(\$00), Y	○
	0EEF	00			BRK		
○	0EF0	00			BRK		○
	0EF1	00			BRK		
○	0EF2	8D	3F	FF	STA	\$FF3F	○
	0EF5	A0	42		LDY	##42	
○	0EF7	24	00		BIT	\$00	○
	0EF9	BA			TSX		
○	0EFA	8A			TXA		○
	0EFB	20	E8	0B	JSR	\$0BE8	
○	0EFE	E8			INX		○
	0EFF	BD	00	01	LDA	\$0100, X	
○	0F02	C0	31		CPY	##31	○
	0F04	F0	F5		BEO	\$0EFB	
○	0F06	88			DEY		○
	0F07	B0	F2		BCS	\$0EFB	
○	0F09	AE	00	00	LDX	\$0000	○
	0F0C	AD	00	00	LDA	\$0000	
○	0F0F	8D	3E	FF	STA	\$FF3E	○
	0F12	A0	49		LDY	##49	
○	0F14	20	E8	0B	JSR	\$0BE8	○
	0F17	8A			TXA		
○	0F18	A0	4E		LDY	##4E	○
	0F1A	20	E8	0B	JSR	\$0BE8	
○	0F1D	20	FF	0F	JSR	\$0FFF	○
	0F20	A0	27		LDY	##27	
○	0F22	A2	FF		LDX	##FF	○
	0F24	E8			INX		
○	0F25	BD	26	06	LDA	\$0626, X	○
	0F28	20	E8	0B	JSR	\$0BE8	
○	0F2B	E0	01		CPX	##01	○
	0F2D	D0	02		BNE	\$0F31	
○	0F2F	88			DEY		○

○	0F30	E8			INX		○
	0F31	E0	04		CPX	##04	
○	0F33	D0	EF		BNE	\$0F24	○
	0F35	BC	C6	0F	LDY	\$0FC6, X	
○	0F38	BD	C0	0F	LDA	\$0FC0, X	○
	0F3B	9D	C6	0F	STA	\$0FC6, X	
○	0F3E	98			TYA		○
	0F3F	9D	C0	0F	STA	\$0FC0, X	
○	0F42	CA			DEX		○
	0F43	D0	F0		BNE	\$0F35	
○	0F45	A0	00		LDY	##00	○
	0F47	EA			NOP		
○	0F48	CA			DEX		○
	0F49	D0	FC		BNE	\$0F47	
○	0F4B	88			DEY		○
	0F4C	D0	F9		BNE	\$0F47	
○	0F4E	A9	FD		LDA	##FD	○
	0F50	8D	30	FD	STA	\$FD30	
○	0F53	8D	08	FF	STA	\$FF08	○
	0F56	AD	08	FF	LDA	\$FF08	
○	0F59	4A			LSR		○
	0F5A	B0	18		BCS	\$0F74	
○	0F5C	AD	62	06	LDA	\$0662	○
	0F5F	18			CLC		
○	0F60	D0	01		BNE	\$0F63	○
	0F62	38			SEC		
○	0F63	6A			ROR		○
	0F64	6A			ROR		
○	0F65	6A			ROR		○
	0F66	8D	62	06	STA	\$0662	
○	0F69	4A			LSR		○
	0F6A	4A			LSR		
○	0F6B	AA			TAX		○
	0F6C	BD	0C	06	LDA	\$060C, X	
○	0F6F	8D	9C	0B	STA	\$0B9C	○
	0F72	90	D3		BCC	\$0F47	
○	0F74	0A			ASL		○
	0F75	30	D7		BMI	\$0F4E	
○	0F77	A9	10		LDA	##10	○
	0F79	8D	02	FF	STA	\$FF02	
○	0F7C	8D	09	FF	STA	\$FF09	○
	0F7F	AD	1D	FF	LDA	\$FF1D	
○	0F82	C9	D0		CMP	##D0	○
	0F84	D0	F9		BNE	\$0F7F	
○	0F86	8C	03	FF	STY	\$FF03	○
	0F89	4C	BE	FC	JMP	\$FCBE	
○	0F8C	78			SEI		○
	0F8D	A9	06		LDA	##06	
○	0F8F	8D	15	03	STA	\$0315	○
	0F92	A9	16		LDA	##16	
○	0F94	8D	E5	07	STA	\$07E5	○
	0F97	AD	9C	0B	LDA	\$0B9C	
○	0F9A	8D	14	06	STA	\$0614	○
	0F9D	09	80		ORA	##80	
○	0F9F	8D	0D	06	STA	\$060D	○
	0FA2	AD	15	FF	LDA	\$FF15	
○	0FA5	8D	0C	06	STA	\$060C	○

0	0FAB	8D 9C 0B	STA	\$0B9C	0
0	0FAB	A9 20	LDA	##20	0
0	0FAD	20 E8 0F	JSR	\$0FEB	0
0	0FB0	4A	LSR		0
0	0FB1	8D 0A FF	STA	\$FF0A	0
0	0FB4	8C 62 06	STY	\$0662	0
0	0FB7	84 CA	STY	\$CA	0
0	0FB9	84 CD	STY	\$CD	0
0	0FBB	A2 02	LDX	##02	0
0	0FBD	8E 02 FF	STX	\$FF02	0
0	0FC0	8D 09 FF	STA	\$FF09	0
0	0FC3	8C 03 FF	STY	\$FF03	0
0	0FC6	EA	NOP		0
0	0FC7	58	CLI		0
0	0FC8	4C 03 87	JMP	\$8703	0
0	0FCB	20 2B 10	JSR	\$102B	0
0	0FCE	03	???		0
0	0FCF	1F	???		0
0	0FD0	48	PHA		0
0	0FD1	29 0F	AND	##0F	0
0	0FD3	20 F3 0B	JSR	\$0BF3	0
0	0FD6	68	PLA		0
0	0FD7	4A	LSR		0
0	0FDB	4A	LSR		0
0	0FD9	4A	LSR		0

0	0FDA	4A	LSR		0
0	0FDB	C9 0A	CMP	##0A	0
0	0FDD	90 02	BCC	\$0FE1	0
0	0FDF	69 C6	ADC	##C6	0
0	0FE1	69 30	ADC	##30	0
0	0FE3	99 97 0F	STA	\$0F97,Y	0
0	0FE6	88	DEY		0
0	0FE7	60	RTS		0
0	0FE8	A0 4F	LDY	##4F	0
0	0FEA	C0 1B	CPY	##1B	0
0	0FEC	B0 0C	BCS	\$0FFA	0
0	0FEE	B9 40 CF	LDA	\$CF40,Y	0
0	0FF1	29 3F	AND	##3F	0
0	0FF3	C0 06	CPY	##06	0
0	0FF5	B0 03	BCS	\$0FFA	0
0	0FF7	B9 E6 06	LDA	\$06E6,Y	0
0	0FFA	20 FB 0B	JSR	\$0BFB	0
0	0FFD	D0 EB	BNE	\$0FEA	0
0	0FFF	60	RTS		0



*Kelencsés K  
és boldog új*



# KERESZTREJTVÉNY

VÍZSZINTES: 1. A függőleges 46 folytatása. 11. Ferde. 12. Fás szárú növény. 13. Mutatószó. 14. Aktínium vegyjele. 17. Rubidium vegyjele. 18. Gyarapodik. 20. Budapest, XIII. kerület egyik utcájának neve. 25. Állóvízre. 27. Van ilyen csősz is. 28. Ilyen ház is létezik. 30. Svájci kanton. 32. E község fölött kezdődik a Mátra. 34. Hosszú idővel ezelőtt. 35. Egyforma betűk. 36. Borsod-Abaúj-Zemplén megyei ipari község. 38. Tejtermék. 40. Idegen szavak előképzője. 41. Csoportosítás. 43. Római számok összege: 555. 44. Építészeti stílus. 46. Megfagyott víz. 48. Juttass. 50. Bibliai alak. 52. Eszes. 53. Háziszárnyasok. 54. Szamáreleség. 55. ... Mátyás (1684–1749): evangélikus lelkész volt. 57. ...es-Salaam, Tanganyika fővárosa. 59. Van ilyen költség is. 60. Bródy Sándor névjele. 61. Székesegyház. 63. Hajó far. 65. Névelő. 66. Pest megyei helység. 68. Ugyancsak Pest megyei helység.

FÜGGŐLEGES: 1. Latin ügynökség. 2. Kettős betű. 3. Értéke. 4. ... Károly (1936– ): színész. 5. Határrag. 6. Kémiai elem. 7. Létezik. 8. Női név becézve. 9. Testrész. 10. Betű kimondva. 15. Női név. 16. Pest megyei község. 19. Azonos a 30. vízszintessel. 21. Szem közepe. 22. A számítástechnikával kapcsolatos megállapítás. 23. Diszprózium vegyjele. 24. Szélesre nyit. 26. Arc része. 29. Óv. 31. Vágott. 33. Énekel. 36. Azonos a vízszintessel. 37. Azonos a 43. vízszintessel. 38. Ilyen irat is létezik. 39. Jön-megy. 42. Egyik megyénk. 45. Állam. 46. A függőleges 22. folytatása. 47. Labda a hálóban. 48. Vissza: állófilm. 49. Azonos a 39. függőlegessel. 50. Csen. 51. Csonka szobrok. 56. Júliusban volt a nevenapja. 58. Van ilyen lexikon is. 61. Pinceszag. 62. Nem azonos. 63. Évszak. 64. Ötórai ital. 66. Görög abc egyik betűje. 67. Azonos betűk. 68. Kisgyerek köszönés. 69. Elté (I).

Mokos István

1		2	3	4	5		6		7	8	9	10		E
	○	11				○		○	12				○	
13		○	14		○	15	○	16	○	17		○	18	
	○	19	○	20	21		22		23		○	24	○	
25	26			○	27					○	28		29	
30			○	31	○	32			○	33	○	34		
35		○	36		37	○		○	38		39	○	40	
	○	41											○	I
○	42	○	43			○	0	○	44			○	45	○
46		47	○		○	48		49	○		○	50		51
52				○	53					○	54			
55			○	56	○	57			○	58	○	59		
60		○	61		62	○		○	63		64	○	65	
	○	66				67	○	68				69	○	
N							N							

Magyarországi újságírók
   
 szövet kiadása O.C.E.



# kedvezmények

A kedvezmény a megjelenéstől számított egy hónapig érvényes.



A NOVOTRADE RT. 2C Áruházában az Egyesület PLUSZ- és SZUPER PÁHOLYÁNAK tagjai kedvezményel vásárolhatják meg a következő programokat:

## Kedvezmények

a NOVOTRADE 2C Áruházában:

### DEÁKPÁHOLY

Basic Logo pascal	149 Ft
Fizikomp	129 Ft
Első könyvem a chipről	99 Ft
Sakkprogramozás mindenkinek	293 Ft
Robotok és a C-64	249 Ft
Mesterséges intelligencia	349 Ft
MPS 1000 felhasználói kézikönyv	180 Ft

### PLUSZ ÉS SZUPERPÁHOLY

Enterprise 128 K	
Speak Easy	3750 Ft

#### Fényceruza

A legújabb, 1.3-as GEOS-sal is működő, külön nyomógombbal rendelkező fényceruza, megtekinthető, megrendelhető és megvásárolható a 2C Áruházban. **Ára: 2500 Ft**

KEDVEZMÉNYEK a PC-Szalontól a Szuperpáholy tagok részére!

IMPOSSIBLE MISSION

II. 15% **781,- ÁFÁVAL**  
játékprogram

AMBASSADOR PC szövegszerkesztő  
10% **5950,- ÁFÁVAL**



## decemberi 60 forintos vásárlási utalvány

Beváltható készpénzes vásárlás esetén az ÁPISZ szaküzleteiben  
XI., Budafoki út 7.  
VIII., Szigony u. 15.

Érvényes: 1990. március 31-ig



## decemberi 60 forintos vásárlási utalvány

Beváltható készpénzes vásárlás esetén a 2C áruházban XIII., Balzac u. 35. és a Művelt Nép vidéki boltjaiban működő 2C sarkokban.

Érvényes: 1990. január 31-ig

A Newline számítástechnikai vállalkozás 10% kedvezményt ad az egyesület tagjainak:

C 16 beépíthető 64 KByte memóriabővítő

16-64-es átkapcsoló	1990,- Ft
beépítés munkadíja	150,- Ft
ROMTURBO 16	490,- Ft
együttes megrendelés esetén	770,- Ft
árengedménnyel:	3400,- Ft
	3060,- Ft

Jogosultak: a Plusz- és a Szuperpáholy tagjai  
Igazolás: ennek a tikketnek postai elküldésével  
Cím: Newline, 1014 Budapest, Tárnok u. 26. 1/5.



A Fotoelektronik-Novotrade-Alfa KFT az alább felsorolt szervezeteiben mindenféle szervizszolgáltatás munkadíjából és az 1989. évben megkötött átalánydíjas szerződés összegéből 10% kedvezményt ad az egyesületi tagoknak.  
Határidő: nincs, illetve átalánydíjas szerződésnél 1989. 12. 01.

A kedvezményt nyújtó szervezeteink:

1053 Budapest, Magyar u. 12-14.	Telefon: 173-551
1083 Budapest, Szigony u. 9.	Telefon: 343-153
1191 Budapest, Gábor Á. sétány 3.	Telefon: 274-763
3100 Salgótarján, Arany J. u. 3.	Telefon: 32-14-007
3525 Miskolc, Fazekas u. 1-3.	Telefon: 46-17-011
4034 Debrecen, Holló L. u. 14.	Telefon: 52-32-863
5600 Békéscsaba, Bartók B. u. 37.	Telefon: 66-27-195
6726 Szeged, Csongrádi sugárút 76.	Telefon: 62-13-377
7400 Kaposvár, Füredi u. 24.	Telefon: 82-16-307
7624 Pécs, Jurisics M. u. 17.	Telefon: 72-11-812
8000 Székesfehérvár, Széchenyi u. 15/a.	Telefon: 22-12-711

9024 Győr, Bem J. tér 1.	Telefon: 96-12-802
9700 Szombathely, Szalonok u. 31.	Telefon: 94-13-419

Igazolás: a javítandó berendezés leadásakor egyesületi igazolvánnyal.  
A kedvezmény többször is igénybe vehető.





**KOGINFORM**

# KOGINFORM

**MŰSZAKI FEJLESZTŐ KISSZÖVETKEZET**



**KOGINFORM**

Levél cím: 1325 Újpest 1. Pf.: 159. Tel.: 18-94-642, 18-94-756

## *Felhasználói rendszerek IBM XT és AT számítógépekre*

Vezetésszolgáltatási Információs Rendszer	120 000 Ft + ÁFA
Anyag- és Félkésztermék Nyilvántartási Rendszer	200 000 Ft + ÁFA
Pénzügyi és Főkönyvi Könyvelési Rendszer	250 000 Ft + ÁFA
Teljesítményelszámolási Rendszer	90 000 Ft + ÁFA
Munkaügyi és Személyzeti Nyilvántartó Rendszer	150 000 Ft + ÁFA
Bérszámfejtési Rendszer	180 000 Ft + ÁFA
Termelésirányítási Rendszer	250 000 Ft + ÁFA
Értékesítési, Készáru-nyilvántartási és Számlázási Rendszer	150 000 Ft + ÁFA

### **Szolgáltatásaink**

- A rendszert a Megrendelő igényeihez adaptáljuk — egy szakemberünk egy havi munkájával térítésmentesen, azon felül meg egyezés szerint.
- Betanítjuk a rendszert ingyenesen.
- A rendszer beüzemelését szolgáló próbaüzem időtartama alatt — kívánságra — programozói, ill. operátori ügyeletet biztosítunk.
- A rendszer átadását követő 5 éven keresztül térítésmentesen elvégezzük a szabályozórendszer változásainak következtében felmerülő módosításokat évi egy ember/hónap munkaigényig.
- Számítógéppel nem rendelkező ügyfeleink részére programrendszereket az üzemeltetésükhöz szükséges számítógép-konfigurációval együtt biztosítjuk:
  - az eszközöket eladjuk,
  - az eszközöket bérbe adjuk,
  - hálózatot telepítünk.

A legtöbb helyen saját gépeinket kihelyezzük a felhasználóhoz, és azokon ún. gépi adatfeldolgozó kapacitásslolgáltatást nyújtunk.

Áraink munkahelyenként értendőek, a megvásárolt darabszámtól függően árendedményt adunk.

Ez a mennyiségi árendedmény érvényes akkor is, ha több, különböző Rendszerünket vásárolják meg.

Oktatási és egészségügyi intézmények részére további 50%-os árendedményt adunk.

# A KOGINFORM Műszaki Fejlesztő Kisszövetkezet hardver ajánlata

## IBM kompatibilis számítógépek:

### XT turbo alapgép

- 8088 CPU
- 6/8 MHz órajel
- 640 KByte RAM
- 360 KByte 5.25" floppy drive
- monochrom monitor 73 200 Ft + ÁFA

### AT turbo alapgép

- 80286 CPU
- 8/12 MHz órajel
- 640 KByte RAM
- 1.2 MByte 5.25" floppy drive
- monochrom monitor 94 060 Ft + ÁFA

### AT-386 alapgép

- 80386 CPU
- 16/20 MHz órajel
- 2 MByte RAM
- 1.2 MByte 5.25" floppy drive
- 1.44 MByte 3.5" floppy drive
- monochrom monitor 159 300 Ft + ÁFA

## Bővítések:

- színes monitorral +20 700 Ft + ÁFA
- EGA monitorral +49 200 Ft + ÁFA
- 20 MByte Winchester (kártyával) +40 500 Ft + ÁFA
- 40 MByte Winchester (kártyával) +65 550 Ft + ÁFA
- 80 MByte Winchester (kártyával) +97 800 Ft + ÁFA
- ArcNet kártya +12 300 Ft + ÁFA

## Nyomtatók:

- Epson FX-1050 69 750 Ft + ÁFA
- Epson DFX-5000 280 000 Ft + ÁFA
- Sharp laser printer 255 000 Ft + ÁFA

## Szünetmentes

- tápegység (300 W) 65 000 Ft + ÁFA

**Áraink 1 év garanciát tartalmaznak.**



**KOGINFORM**

**KOGINFORM**  
**MŰSZAKI FEJLESZTŐ KISSZÖVETKEZET**  
**SZÁMÍTÁSTECHNIKAI ÁGAZAT**

Telefon: 18-94-120. Telefax: 18-94-597



**KOGINFORM**