



MIKROSZÁMÍTÓGÉP
MAGAZIN

A NEUMANN JÁNOS
SZÁMÍTÓGÉP
TUDOMÁNYI
TÁRSASÁG
LAPJA

Ára: 30 Ft

1986
május





A NEUMANN JÁNOS SZÁMÍTÓGÉP- TUDOMÁNYI TÁRSASÁG LAPJA

A kiadvány
a Tudományszervezési
és Informatikai
Intézettel
együttműködve készül

A szerkesztőbizottság
vezetője:
Kovács Győző

E számunkat
szerkesztették:
Bakos Tamás
(programozástechnika)

Broczkó Péter
(hírek)

Kovács Győző
(levelezés)

Lindner László
(sakkprogramozás)

Petróczy Judit
(könyvek)

Simonyi Endre
(klub)

Vadkerti János
(µprogramok)

Varga András
(iskola — számítógép)

A szerkesztőség
munkatársa:
Kardos Zsuzsa

Felelős szerkesztő:
Könyves Tóth Pál
Szerkesztőség:
1027 Budapest II., Fő u. 68.
Telefon: 154-250

Kiadja a Delta Szaklapkiadó
és Műszaki Szolgáltató
Leányvállalat
Felelős kiadó:
Faklen Pál igazgató
1142 Budapest VII., Garay u. 5.
Telefon: 415-583, 215-440

Terjeszti a Magyar Posta
Előfizethető
bármely postahivatalban,
a kézbesítőknél,
a Posta hírlapüzleteiben
és a Hírlapelőfizetési
és Lapellátási Irodánál
(Budapest V., József nádor tér 1.
Postacím: 1900 Budapest)
közvetlenül
vagy postautalványon,
valamint átutalással
a HELIR 215—96162
pénzforgalmi jelzőszámmra.
Megjelenik havonta
Példányonkénti ára 30,— Ft
Előfizetési díj:
egy évre 360,— Ft
fél évre 180,— Ft



Szikra Lapnyomda
Budapest (86-3144)
Felelős vezető:
CsönDES Zoltán vezérigazgató

INDEX: 25629
ISSN 0236-6088

Címképván:
Az iskolaszámítógép-pályázat
nyertes gépe
a középiskolás kategóriában:
a PRO/PRIMO,
FDU 1109 Floppy
és DATACOOP nyomtató

Tartalom

Házunk tája	2
Adok — veszek — cserélek	15
A minőségügy közügy	26
Assemblerek, cross-assemblerek	28

ISKOLA — SZÁMÍTÓGÉP

A D 100-as nyomtató illesztése HT—1080Z számítógéphez	3
Negyedfokú egyenlet megoldása	5
Gép a gépen III.	6

PROGRAMOZÁSTECHNIKA

Strukturált programtervezés	8
-----------------------------	---

PIAC

Melyiket vegyem?	10
Kereskedni is tudni kell!	11
Egy bemutató margójára	15

TERMÉKISMERTETŐ

Ez már nem játék!	14
-------------------	----

ALKALMAZÁSOK

Meteorológiai program C64-re	16
------------------------------	----

FÓRUM

µPROGRAMOK	17
------------	----

Öt gép vizsgálata egy programmal	19
----------------------------------	----

µKLUB

PTA-4000. LH 5801 utasításkészlet	21
Módosított hűtőlemez ZX81-hez	23

DIÁKROVAT

Kétszeres nagyságú, technikai típusú karakterek	30
A számítógép anatómiája	32
Nagyító	34
Matematika	35

JÁTÉKPROGRAMOK

AZ OLVASÓ ÍRJA	42
----------------	----

SAKKPROGRAMOZÁS

Bitek és figurák	45
------------------	----

KÖNYVEK

HÍREK, ÉRDEKESSÉGEK	47
---------------------	----

Házunk tája

„Ős van, — gurul az átlami szó s telek
igérete száll távol hegyek ormán.

Ülök, a nedves szél arcon simogat,
a rendre gondolok s szemem mögött
igáslovak üznek vad kocsisokat.”

(RADNÓTI MIKLÓS: PARTON)

Egyik nagyon kedves barátom, maga is és főleg gyermekei hűséges olvasói a μ M-nak, immár másodszer tette szövé: tegyünk már valamit a lapban közölt programok hibátlansága érdekében. Elismerte, hogy a Program(ön)kritika — amelynek létrejötté részben az ő kritikájának volt köszönhető — már valami, de az igazi mégiscsak az lenne — mondta —, ha hibátlan programokat közlünk. Erre törekszünk — mondtam én —, és talán értünk is el némi eredményt az elmúlt években, de hibátlan programokat majd csak akkor tudunk hozni, ha minden, az országban használt számítógépből lesz egy-egy példány a szerkesztőségben. Mindegyik géphez lesz természetesen nyomtató is, hogy a programlistákat mi magunk ki tudjuk nyomtatni. Ma mindehhez kevesen vagyunk, a programteszteléshez kölcsöngépeket használunk, ezért — főleg a nagyobb programokat — a szerzők által küldött listákban közlőjük, amelyek sajnos nemegyszer hibásak.

Főleg ifjabb olvasóink vették észre, hogy a Százlábú, amelyet diákok szerkesztettek, és egyik ifjú levelezőm szerint többet ér, mint az egész lap, már nagyon régen nem jelentkezett. Ennek az az oka, hogy a diákszerkesztők érettségiztek, felvették őket az egyetemre, behívták katonának, és így nem volt lehetőségük, hogy a rovatot szerkesszék. Nem gondoskodtunk időben igazi utánpótlásról, a szerkesztőség ifjabb munkatársai vezetés nélkül elszeledtek, az anyag elfogyott, a Százlábú elaludt. Most ismét van diákszerkesztőségünk, van friss anyag, készülő a folytatás. Az új szerkesztőségnek már nemcsak érettségiző tagjai vannak, de ifjabb és főleg aktív munkatársai is, akik az érettségi idején átvesszik majd a szerkesztést, és tovább írják a lapot. A szerkesztőség nyitott, a diákszerkesztőség is. Akik tehetséget éreznek a szak-újságíráshoz, jelentkezzenek Kardos Zsuzsa szerkesztőnél, szelreteltel várjuk az ifúságot.

A havi megjelenéssel szorosabbra szeretnénk fűzni kapcsolatunkat az országban működő klubokkal. A mozgalom-

ról — ez föltétlenül a mi hibánk — újabban nagyon keveset lehet hallani. Szeretnénk, ha lapunkban folyamatosan tájékoztatást adhatnánk a klubok munkájáról, fejlesztési és oktatási eredményeiről. Szeretnénk a μ M-t a klubok fórumává fejleszteni. Rendezni kell a klubok és a Neumann Társaság közötti kapcsolatot is. Itt volna az ideje, hogy a klubok például szakosztályi szinten vagy a budapesti, vagy a területi szervezetek keretében szervezettel is közvetlenül a Társasághoz kapcsolódjanak.

A Magyar Televízióval igen jó munkakapcsolatunk alakult ki, amelyet szeretnénk továbbfejleszteni. A tv „Mi és a számítógép” műsora voltaképpen tévés magazin, tartalmában, célkitűzéseiben nagyon közel áll a μ M-hoz. Már nagyon régen tervezzük, hogy kölcsönösen rovatot indítsunk: a μ M-ban előzetesen bemutatjuk a „Mi és a számítógép” riportjait, közlőjük azokat a programokat, amelyeket a nézők a műsorokban láthatnak. És fordítva: a „Mi és a számítógép” képes és hangos riportban az előzetest a μ M következő számából, felhívásokat, pályázatokat közöl. Egyszerűen közösen próbáljuk meg népszerűsíteni az informatikát. Beszéltünk közösen rendezett vetélkedőről és versenyekről is, amelyeknek mind lapunkban, mind a képernyőn igyekszünk megfelelő publicitást adni.

Most kezdjük az együttműködést a Magyar Rádióval is, hiszen a két média az informatika oktatásában nagyon jól együtt tud működni. Egyetlen példa az MTV, a Rádió és a μ M együttműködésére, hogy a nem is távoli jövőben megkezdjük kazettára felvett programok sugárzását az ultrarövidhullámon. Ezekről a kazettákról nemcsak kész játé- és más programokat szeretnénk sugározni, de tanfolyami anyagot is, sőt a μ M rovatból is szemelvényeket, vagyis rádió — képet.

Sajnos Magyarországon még nincs nyilvános postai videotext-hálózat — mindent megteszünk annak érdekében, hogy legyen —, így számítógépterminál-hálózatot a tömegoktatásban felhasználni ma még nem lehet. Marad a tévé, a rádió és az újság. Az együttműködés mindig hasznos: a társadalmi méretű informatizálási programot ennek hiányában biztosan nem lehet eredményesen végrehajtani.

KOVÁCS GYÖZŐ

Az iskolaszámítógép-pályázat eredménye

Páris György igazgató (Tudományos Szervezési és Informatikai Intézet) sajtótájékoztatóján elmondta, hogy a pályázat általános iskolai kategóriájában az első helyre két gép került: a C16 (PIÉRT/Novotrade) és a TVC kis kiépítésű változata (Videoon). A harmadik helyezett a Pro/Primo kis kiépítésű változata (MTA SZTAKI—Mikrokey) lett. A középiskolai pályázat első három helyezette a Pro/Primo bővített változata, a TVC bővített változata és a HT—3080Z (Híradástechnika Szövetkezet). A gépek ára rendre: 8775, 11 000, 13 500 illetve 17 000, 13 000, 20 000 forint (nagykereskedelmi ár). Rényi Gábor, a Novotrade igazgatója ehhez hozzátette, hogy a C16 ára a 64 k-ra bővített változatra vonatkozik, és az iskoláknak andandó dokumentációt is tartalmazza.

Az iskolák a kategóriájuk helyezettjei közül szabadon választhatnak, a TII az igényeket őszig begyűjti, összesíti és továbbítja a szállítóknak. Páris György megjegyezte, hogy az iskolák arra is kapnak lehetőséget, hogy esetleg más típusú gépszerzőseiket folytathassák. Közölte, hogy a jelenlegi öt éves tervben mindkét iskolatípus 25—25 000 gépet kap. Ehhez a megyei tanácsoknak kell a pénz biztosítaniuk.

Rényi Gábor hangsúlyozta, hogy pusztán a gépgyártás nem elég, biztosítani kell az alkalmazáshoz szükséges teljes hátteret, amiről cége a C16 esetében gondoskodik is. A többi géppel kapcsolatos kérdéseinkre nem kaptunk választ a gyártóktól, de Rényi Gábor felajánlotta cége segítségét, Páris György pedig javasolta a tanároknak és másoknak is, hogy minél több, az oktatásban használható programot dolgozzanak ki, amit a TII terjesztésre meg is vásárol.

Rényi Gábor azt is kijelentette, hogy a C16 alkatrészellátása megoldott. A Commodore céggel kötött megállapodás értelmében Magyarországon konszignációs raktár létesül. Ehhez a PIÉRT képviselője megjegyezte, hogy az ellátás köteleességük is 5 évig, a belkereskedelmi törvény alapján. Ez a bejelentés természetesen nemcsak a majdan iskolákba kerülő gépek tulajdonosainak jó hír, hanem az eddigi vevőknek is, hiszen így mentesülnek a két kritikus — és csak a Commodore cég által gyártott, kereskedelmi forgalomba nem került, igen borsos árú (170 DM!) — alkatrész feketepiaci beszerzésétől.

Az iskolák választásának előlegésztése céljából termékismertető rovatunkban a *Melyiket vegyem?* sorozatban értékelni fogjuk a jelölteket, amire az olvasók leírt tapasztalatait is kérjük. Elsőnek, következő számunkban a TVC típust ismertetjük.

A D 100-as nyomtató illesztése HT-1080Z számítógéphez

A közelmúltban több iskola vásárolt a SZÁMALK-tól lengyel gyártmányú D 100-as típusú mozaiknyomtatót. Ennek működéséről itt nem kívánok szólni. Cikkemet azoknak szánom, akiknek már van ilyen nyomtatójuk, és azt HT-1080Z iskolaszámítógéphez használják.

A SZÁMALK adott hozzá egy illesztőegységet és egy kazettát, amelyen egy rövid SYSTEM program illeszti a géphez a nyomtatót. Ez annyit tesz, hogy ha 13-at (ODH=CR) küldünk ki, akkor azt megoldja egy 10-es kóddal (soremelés). A nyomtatót — kérésünkre — úgy kapjuk, hogy az eredeti ASCII, lengyel és cirill jelkészleten kívül EPROM-jába égették a 144—161-es kódokkal a magyar ékezetes kis- és nagybetűket is.

A program segítségével illeszthetjük a nyomtatót a HT-1080Z/64 számítógéphez (1. ábra). Ahhoz a változathoz, amely a „BASIC ROM EXTENSION 2.51” szöveggel jelentkezik be a bővítés meghívása után.

A program önnindító (3—4-es sor). Kezdőcíme a START címével egyezik meg. Beolvasás után védi a programot, és a 4026H címen található rutincímét felírja a saját címével. Ennek következtében a program „átlátszó” lesz a felhasználó számára, a nyomtatót működtető rutinokat változtatlanul úgy kell használni, mint a program beolvasása előtt.

A 14—122-es sorokban felismeri a program, hogy most a PRINTSCREEN funkciót kell-e megvalósítania vagy sem. Az előbbi esetben soronként kiviszi a képernyő tartalmát a nyomtatóra megfelelő átkódolással úgy, hogy a 80H-nál nem kisebb kódok helyett szöközt viszi ki, majd, ha ilyen előfordult, akkor a sor elejére visszatérve újra kiviszi a szemigrafikus kódokat úgy, hogy most a 80H-nál kisebb kódok helyett 80H-t viszi ki. Erre azért

van szükség, mert a kódok folyamatos kivitelek a kevert karakter- és grafikus kódok típusváltásokról fel sor emelését idézik elő a nyomtatón.

A teljes képernyő kivitele után még hat sort emel, majd visszaugrik abba a rutinba, amelyből meghívtuk a PRINTSCREEN rutint (23—69-es sor).

Mivel a magyar ékezetes karaktereknek a VIDEORAM-ban a 0 és 31 közötti kódok felelnek meg, amelyek közvetlenül a nyomtatóra kerülve, ott különféle vezérléseket okoznak, ezeket a kódokat a 70—81-es sorokban át kell konvertálni. Ha olyan karakter jelenik meg a képernyőn, amelynek nincs megfelelője a nyomtatón, akkor az ISM jelű (5-ös sor) kódot viszi ki helyette. Mivel a nyomtatói kódjai több helyen különböznek a számítógép kódjaitól, itt is kell konvertálni (82—96-os sor).

Ha LPRINT vagy egyéb módon viszünk ki karaktert a nyomtatóra, akkor a 99-es sorra ugrik a fentiek helyett. Itt lekezelet a CR kódját (99—101-es sor), majd egy flag függvényben latin betűs vagy cirill betűs üzemmódba vált (102—111-es sor). A cirill betűs üzemmódban (2. ábra) tetszőleges szöveg vihető ki a papírra. A latin betűssel akár karakterenként változathoz az üzemmód.

Az ábra alsó sorában látható karaktereket beírva, ebben az üzemmódban a felettük látható karakterek jelennek meg a SHIFT billentyű állásától függően. Ha olyan billentyűt nyomunk meg, amely nem szerepel az alsó sorban, akkor az annak megfelelő karaktert írja ki a nyomtató. Ha a cirill betűs szöveget számokkal akarjuk keverni, akkor célszerű latin betűs üzemmódba visszaváltani (112—126-os sor).

A 127—149-es sorokban egy táblázat segítségével (154—158-as sor) átkódolja a magyar ékezetes karaktereket, ha a megfelelő üzemmódban van a gép. Ezt a SHIFT

1	:	D-100 NYOMTÁTO ILLESZTÉSE HT 1080Z/64-HEZ
2	:	KESZITETTE LANYI TAMAS, 1985.11.12.
3		ORG 41E2H
4		JFJ (HL)
5		EGU 127
6		ORG 0FEB0H
7	ISM:	HL+DCBFR ; ISMERETLEN KOD JELE
8	START:	LD HL+SP ; INICIALIZALAS
9		LD (4026H),HL
10		DEC HL
11		LD (40B1H),HL
12		LD A+0C9H
13		LD (41E2H),A
14		JFJ 5000H
15	DCBFR:	LD HL+10 ; UZEMMOD KIVALASZTAS
16		ADD HL+SP
17		LD A+50H
18		CP (HL)
19		JR NZ+D1
20		INC HL
21		LD A+31H
22		CP (HL)
23	D1:	JFJ NZ+LPRINT
24		LD SP+HL
25		LD HL+3C00H
26		RES 1,(1X*4)
27		CALL CR
28		BIT 6,AH
29		JR NZ+K0
30		LD A+(HL)
31		INC HL
32		CP 00H
33		JR C+0A
34		SET 1,(1X*4)
35		LD A+20H
36		LD C,A
37		CALL DT
38		LD A+AL
39		AND 5FH
40		JR NZ+W2
41		BIT 1,(1X*4)
42		JR Z+W1
43		LD A+13
44		CALL 5B4H
45		LD DE+0-64
46		ADD HL+DE
47		LD A+1BH
48		CALL 5B4H
49		LD A+50H
50		CALL 5B4H
51		LD B+64
52		LD A+(HL)
53		INC
54		CP 00H
55		JR NC+W2
56		LD A+00H
57		AND 0FH
58		CALL 5B4H
59		DJNZ W6
60		LD A+1BH
61		CALL 5B4H
62		LD A+35H
63		CALL 5B4H
64		JR W1
65		LD A+6
66		CALL CR
67		DJNZ W9
68		XOR A
69		RET
70	DT:	CP 20H ; KARAKTEREK KIVITELE
71		JR NC+T00
72		ADD A+144 ; 0 - 31 KODOK KEZELESE
73		CP 153
74		JR C+D4
75		SUBB 7
76		CP 153
77		JR C+D6
78		CP 162
79		JR C+D4
80		LD A+EM
81		JFJ 5B4H
82		OR 20H ; A NYOMTÁTO ILLESZTÉSE
83		CP 60H ; 0
84		JR NZ+D2
85		LD A+106
86		DS 3
87		JR ZBH
88		JR C+5B0H
89		CP 7FH
90		JR NC+5B0H
91		LD A+00
92		ADD A

1. ábra (folytatása a 4. oldalon)

93 FF66	FEAF	CP	175	
94 FF68	3802	JR	CAD7	
95 FF6A	D610	SUB	16	
96 FF6C	C38405	D7:	JP	58DH
97 FF6F	3E59	DB:	LD	A:91
98 FF71	18F9		JK	D7
99 FF73	79	LPRINT:	LD	A+C
100 FF74	FE00		CP	13
101 FF76	2657		JK	Z:CK
102 FF78	FE82		CP	2
103 FF7A	2005		JR	NZ+T2
104 FF7C	DECB0456		SET	0:(IX+4)
105 FF80	CY		RET	
106 FF81	FE03	T2:	CP	3
107 FF83	2005		JR	NZ+T3
108 FF85	DECB0456		RES	0:(IX+4)
109 FF89	CY		RET	
110 FF8A	DECB0446	T3:	BIT	0:(IX+4)
111 FF8E	281E		JR	Z:LP2
112 FF90	F610		OR	10H
113 FF93	2005		CP	33A
114 FF94	380E		JR	G:LP4
115 FF96	FE38		CP	36H
116 FF98	308A		JR	NC:LP4
117 FF9A	79		LD	A+C
118 FF9C	C650		ADD	A:5BH
119 FF9D	FE90		CP	6BH
120 FF9F	3802		JR	C:LP5
121 FFA1	D630		SUB	50H
122 FFA3	4F	LP5:	LD	C+A
123 FFA4	FE48	LP4:	CP	40H
124 FFA6	10A805		JP	C:58DH
125 FFA9	CBF9		SET	Z:C
126 FFAE	C38005		JP	58DH
127 FFAE	213640	LP2:	LD	HL+A:30H
128 FFB1	CB56		BIT	Z:(HL)
129 FFB3	2810		JR	Z:TOV
130 FFB5	CB5E		BIT	4:(HL)
131 FFB7	281E		JR	Z:00
132 FFB9	CBDE		RES	3:(HL)
133 FFBF	FE40		CP	60H
134 FFC0	CADFFF		JP	Z:TOV
135 FFC0	21E9FF		LD	HL+TABL
136 FFC3	011400		LD	BC+20
137 FFC6	EDB1		CP:IR	
138 FFC8	C0		RET	NZ
139 FFC9	3CA3		LD	A:163
140 FFCB	91		SUB	C
141 FFCE	FEA2		CP	162
142 FFCE	DAB405		JP	C:584H
143 FFD1	289C		JR	Z:DB
144 FFD3	3E7D		LD	A:123
145 FFD5	1895		JR	DF
146 FFD7	FE40	OD:	CP	60H
147 FFD9	C24FFF		JP	NZ+TOV
148 FFD0	CBDE		SET	3:(HL)
149 FFDE	CY		RET	
150 FFE0	3E80	CR:	LD	A:13
151 FFE1	CDB405		CALL	584H
152 FFE4	3E0A		LD	A:10
153 FFE6	C38405		JP	584H
154 FFE9	4145494F	TABL:	DB	41H+45H+49H+4FH
155 FFE9	33452535		DB	33H+34H+35H+35H
156 FFE1	36616669		DB	36H+61H+65H+69H
157 FFE5	6F232475		DB	6FH+23H+24H+25H
158 FFE9	25263727		DB	25H+26H+37H+27H
159			END	

Az 1. ábra folytatása

2. ábra

A D-100 nyomtatón hívható cirill betűs karakterek

Az üzemmód LPRINT CHR\$(2) kóddal hívható.
Az LPRINT CHR\$(3) kóddal visszaállítható a latin betűs írásmód.

A megfelelő latin betűs karakterek jelennek meg a képernyőn.
A SHIFT billentyű leengedésekor íráshők a negybetűk.

А В С D E F G H I J K L M N O P Q R S T U V X Y Z а б в г д е ж з и к л м н о п қ р с т у в х y z 0 9
А Б В Г Д Е Ж З И Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я
А В М G D E V Z I J K L M N O P R S T U F H C A 3 5 7 Y X 4 9 0

CLEAR és „I” billentyűk egyidejű megnyomásával érhetőek el. Itt az omega karakter helyett az Á és ä karaktereket jelenteti meg a papíron, míg a többi, csak bővítésben szereplő jel helyett nem visz ki semmit, még szőkzött sem (3. ábra). Ebben az üzemmódban az Á és ä kivételével a nyomtató teljes jelkészletét kihasználhatjuk a

CHR\$ segítségével. Mivel a 7FH-nál nagyobb kódokat változatlanul viszi ki a nyomtatóra, ezért a grafikus vagy szemigrafikus funkciókat is ebben az üzemmódban használhatjuk változatlanul.

Az iskolák több lépcsőben kaptak HT—1080D számítógépet, ezért előfordul, hogy más változathoz kívánják illeszteni

A D-100 NYOMTATÓ KARAKTERKÖDJEI

32		08	X	144	0	2000	X
33	I	09	X	145	0	2001	X
34	-	0A	Z	146	1	2002	H
35	\$	0B	Z	147	0	2003	X
36	~	0C	Z	148	0	2004	X
37	X	0D	Z	149	0	2005	X
38	Δ	0E	-	150	0	2006	H
39	∞	0F	-	151	0	2007	X
40	∞	10	-	152	0	2008	X
41	∞	11	-	153	0	2009	X
42	∞	12	-	154	0	2010	X
43	+	13	-	155	0	2011	X
44	-	14	0	156	0	2012	T
45	-	15	0	157	0	2013	Y
46	-	16	0	158	0	2014	X
47	∞	17	0	159	0	2015	0
48	∞	18	0	160	0	2016	0
49	∞	19	0	161	0	2017	H
50	∞	10	1	162	0	2018	0
51	∞	11	1	163	0	2019	0
52	∞	12	1	164	0	2020	X
53	∞	13	1	165	0	2021	H
54	∞	14	1	166	0	2022	X
55	∞	15	1	167	0	2023	X
56	∞	16	1	168	0	2024	H
57	∞	17	1	169	0	2025	X
58	∞	18	1	170	0	2026	X
59	∞	19	1	171	1	2027	H
60	∞	1	1	172	1	2028	X
61	∞	2	1	173	1	2029	X
62	∞	3	1	174	1	2030	0
63	∞	4	1	175	0	2031	T
64	∞	5	1	176	0	2032	X
65	∞	6	1	177	0	2033	H
66	∞	7	1	178	0	2034	X
67	∞	8	1	179	0	2035	H
68	∞	9	1	180	0	2036	X
69	∞	10	1	181	2	2037	H
70	∞	11	1	182	2	2038	H
71	∞	12	1	183	2	2039	X
72	∞	13	1	184	5	2040	H
73	∞	14	1	185	5	2041	X
74	∞	15	1	186	0	2042	X
75	∞	16	1	187	0	2043	X
76	∞	17	1	188	1	2044	T
77	∞	18	1	189	1	2045	X
78	∞	19	1	190	1	2046	X
79	∞	20	1	191	1	2047	H
80	∞	21	1	192	1	2048	H
81	∞	22	1	193	1	2049	H
82	∞	23	1	194	1	2050	X
83	∞	24	1	195	1	2051	H
84	∞	25	1	196	1	2052	X
85	∞	26	1	197	1	2053	H
86	∞	27	1	198	1	2054	X
87	∞	28	1	199	1	2055	X

A „JEL” (SHIFT 0) UTÁNI KÖDOK HATÁSA

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	X	Y	Z	
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5

3. ábra

a nyomtatót. Ehhez a programot módosítani kell az alábbiak szerint.

A régi, „mutatós” változat, amely a bővítés meghívása után a NEW KEYBOARD ROUTINE ENABLE szöveggel rendelkezik, csak 16 kb-ig, és hasonlóképpen a második sorozat is, amelyen a kivezérlésmérő LED és a bővítés meghívása után nem ír ki bejelentőző szöveget. Azokat a változtatásokat, amelyek mindkét régi változatra érvényesek, felsorolom:

- A 6-os sorban ORG 7FOOH
- A 16-os sorban LD A,85H
- A 26, 34, 41, 104, 108, 110-es sorokban a zárójelben helyesen az IX+3 szerepel
- Beirandó az alábbi két sor az END elé (159-es sor):
H58D: LD A,C
JP 5B4H
- A 124-es sorban JP C, H58D

- A 126-os sorban JP H58D
- A 127—149-es sorokat ki kell törölni
- A 154—158-as sorokat is ki kell hagyni

A „mutatós” változatban a fentiekben kívül még:
— A 37-es sorban CALL TOV
— A 111-es sorban JR Z, TOV
— A 88 és 90-es sorokban 58DH helyett H58D irandó
— A 70—81-es és a 97—98-as sorokat ki kell törölni

A LED-es változatban:
— A 37-es sorban CALL 5B4H
— A 111-es sorban JP Z, 5B4H
— A70—98-as sorokat ki kell hagyni

FIGYELEM! Előfordul, hogy egy BASIC programban gépi kódú betét van. Ha interferál a fenti programmal, annak beláthatatlan következményei lesznek. Az ilyen betéteket át kell írni, ha együtt kívánjuk használni a programmal.
LÁNYI TAMÁS

Negyedfokú egyenlet megoldása

Az $x^4 + a \cdot x^3 + b \cdot x^2 + c \cdot x + d = 0$ negyedfokú egyenlet Lodovico Ferraritól (1522—1565) származó módszer szerinti megoldása két másodfokú egyenlet megoldására vezethető vissza. Előbb azonban meg kell oldani egy harmadfokú egyenletet, melynek eredményét a másodfokú egyenletek együtthatóinak képzésékor fogjuk felhasználni.

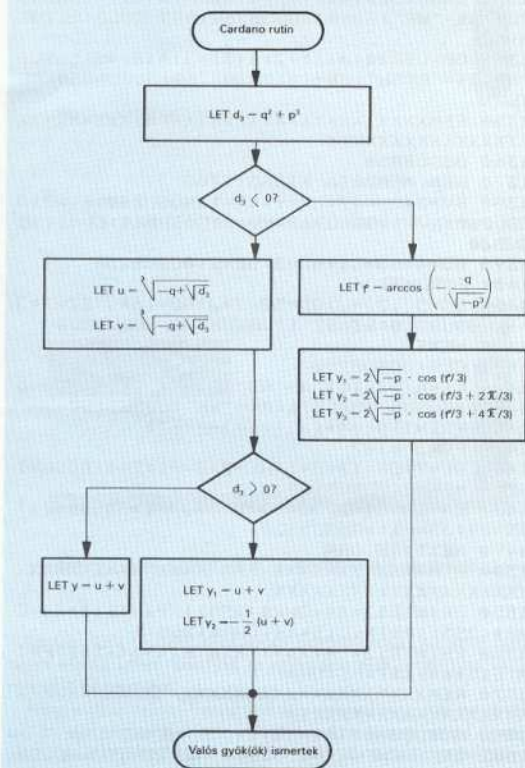
A harmadfokú egyenlet: $y^3 + 3 \cdot p \cdot y + 2 \cdot q = 0$, ahol

$$3 \cdot p = \frac{a \cdot c}{4} - \frac{b^2}{12} - d \text{ és}$$

$$2 \cdot q = \frac{a \cdot b \cdot c}{24} - \frac{a^2 \cdot d}{8} - \frac{b^3}{108} + \frac{b \cdot d}{3} - \frac{c^2}{8}.$$

Megoldása a Cardano-képlettel történik, melynek folyamatábráját bemutatjuk. z-t úgy kapjuk meg, hogy a harmadfokú egyenlet legnagyobb értékű y megoldásához $\frac{b}{6}$ -ot hozzáadjuk.

A Cardano-képlet folyamatábrája



```

3 REM *****
5 REM * FERRARI *
7 REM *****
9 REM "A" A HATVANYOZAS JELE
10 CLS:PRINT"ADD MEG AZ
    X4 + A*X3 + B*X2 + C*X + D = 0"
15 PRINT"EGYENLET EGYÜTTHATÓIT"
20 INPUT "A=";A; INPUT "B=";B;
    INPUT "C=";C; INPUT "D=";D
30 F=(A*C/4-B*B/12)/3;
Q=(A*B*C/24-A*A*D/8-B*B*B/108+B*D/3-C*C/B)/2
40 D3=Q*Q+P*P
50 IF D3<0 THEN GOTO 100
60 U3=-Q+SQR(D3); IF U3<0
    THEN U=-U3*(1/3) ELSE U=U3*(1/3)
65 V3=-Q-SQR(D3); IF V3<0
    THEN V=-V3*(1/3) ELSE V=V3*(1/3)
70 IF D3>0 THEN Z=U+V+B/6; GOTO 200
80 Z1=U+V+B/6; Z2=-.5*(U+V)+B/6;
    IF Z1<Z2 THEN Z=Z2 ELSE Z=Z1
90 GOTO 200
100 X=-Q/SQR((-P)*A3)
110 F=-ATN(X/SQR(1-X*X))+1.5708
120 Z1=2*SQR(-P)*COS(F/3)+B/6
130 Z2=2*SQR(-P)*COS(F/3+2.0944)+B/6
140 Z3=2*SQR(-P)*COS(F/3+4.1888)+B/6
150 IF Z1<Z2 THEN Z=Z2 ELSE Z=Z1
160 IF Z<Z3 THEN Z=Z3
200 IF A*Z-C<0 THEN M=-SQR(ABS(Z*Z-D))
    ELSE M=SQR(ABS(Z*Z-D))
210 P1=A/2+SQR(ABS(A*A/4-B+2*Z)); Q1=Z+M
220 P2=A/2-SQR(ABS(A*A/4-B+2*Z)); Q2=Z-M
230 D1=P1*P1/4-Q1; D2=P2*P2/4-Q2
240 IF D1<0 THEN X1=-P1/2; Y1=SQR(-D1);
    X2=-P1/2; Y2=-SQR(-D1); GOTO 250
245 X1=-P1/2+SQR(D1); Y1=0;
    X2=-P1/2-SQR(D1); Y2=0
250 IF D2<0 THEN X3=-P2/2; Y3=SQR(-D2);
    X4=-P2/2; Y4=-SQR(-D2); GOTO 260
255 X3=-P2/2+SQR(D2); Y3=0;
    X4=-P2/2-SQR(D2); Y4=0
260 IF Y1<0 THEN IR="-" ELSE IR="+"
265 PRINT "X1=", X1, IR, ABS(Y1)
270 IF Y2<0 THEN IR="-" ELSE IR="+"
275 PRINT "X2=", X2, IR, ABS(Y2)
280 IF Y3<0 THEN IR="-" ELSE IR="+"
285 PRINT "X3=", X3, IR, ABS(Y3)
290 IF Y4<0 THEN IR="-" ELSE IR="+"
295 PRINT "X4=", X4, IR, ABS(Y4)
300 INPUT "AKARSZ UJABB EGYENLETET
    MEGOLDANI? (I/N)";KR
310 IF KR="I" THEN GOTO 10
  
```

Az egyenletmegoldó program

A másodfokú egyenletek:

$$x^2 + \left(\frac{a}{2} + \sqrt{\frac{a^2}{4} - b + 2 \cdot z}\right) x + z \pm \sqrt{z^2 - d} = 0$$

$$x^2 + \left(\frac{a}{2} - \sqrt{\frac{a^2}{4} - b + 2 \cdot z}\right) x + z \pm \sqrt{z^2 - d} = 0$$

Kettős műveleti jelnél az alsót akkor kell használni, ha $a \cdot z - c < 0$.

A program nem igényli a számítógéptől az $\arccos(x)$ függvény ismeretét:

$$\arccos(x) = -\operatorname{atn} \frac{x}{\sqrt{1-x \cdot x}} + \frac{\pi}{2}$$

valamint π többszöröse helyett négy tizedesre kerekített értékűeket írta, ez tetszés szerint pontosítható.

A program HT—1080Z-re készült. Ha viszont a használandó gépen nincs ELSE utasítás, úgy újabb, IF-fel kezdődő sorokat kell írni helyette. A hatványozás jele a használt nyomtató, az Epson sajátossága miatt lett á.

Gép a gépen III.

Ebben a részben úgy szervezzük programunkat, hogy a már elkészített szubrutinokat kelljen meghívni. Most az ábrák gyakran megjelenő részleteinek rajzát készítjük el. Ezek a gépen fontos szerepet töltenek be, a működés lényeges részletét valósítják meg, ezért funkcionális egységnek neveztük el őket.

Az első részben közölt ábráról már ismert három doboz, valamint a BUS-vonalak a nyílakkal és a vezérlővonalak. Ezeknek a funkcionális egységeknek a rajzolásához a szubrutinok már megvannak, megfelelő működtetésük csak szervezés kérdése.

```

870 REM*****
*****
880 REM FUNKCIONÁLIS EGYS
EGEK!
890 GOSUB820
900 REMXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXX
910 REMXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXX
920 REM A HARDM DOBOZ
930 T=0:SD=1:OD=11:SZ=5:HD=10:GOSUB190:
A$="M. PROC":SI=3:OI=13:GOSUB710
940 SD=0:OD=35:SZ=3:HD=8:GOSUB190:A$="DE
KOPER":SI=1:OI=36:GOSUB710
950 SD=1:OD=46:SZ=10:HD=6:GOSUB190:A$="M
E M":SI=2:OI=51:GOSUB710
960 A$="----"----:OI=39:FORSI=1TO7:PRINTE
SI*64+OI,A$;:NEXT
970 RETURN
980 REMXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXX
990 REM ADDRESS BUS
1000 GOSUB820
1010 A(1)=9:A(3)=13:A(5)=10:A(6)=2:A(12)
=10:A(13)=1:GOSUB780
1020 SD=2:OD=17:HD=2:SZ=17:ST=1:GOSUB310

1030 SN=10:OI=60:D=4:JB=1:T=-1:GOSUB510
1040 PRINTE64+19,"ADDR. BUS":GOSUB850:R
ETURN
1050 REMXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXX
1060 REM VIZSINTES DATA BUS
1070 GOSUB820:A(1)=5:A(3)=14:GOSUB780
1080 G=0:SD=10:OD=17:HD=0:SZ=41:ST=1:GOS
UB310
1090 SN=31:OI=50:D=2:JB=1:T=-1:GOSUB510
1100 PRINTE9*64+19,"DATA BUS":RETURN
1110 REMXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXX
1120 REM FUGGOLEGES DATA BUS 1
1130 SD=10:OD=59:HD=10:SZ=3:ST=-1:GOSUB4
00
1140 IFTTHENA$=" "ELSEA$="<<<<<<<<
"
1150 SI=1:OI=57:GOSUB710
1160 SN=20:OI=121:D=2:JB=-1:T=-1:GOSUB660
:GOSUB850:RETURN
1170 REMXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXX
1180 REM DATA BUS LE 2

```

```

1190 GOSUB820:A(1)=9:A(3)=11:A(5)=8:GOSU
B780
1200 SD=0:OD=59:HD=10:SZ=3:ST=1:GOSUB400

1210 IFTTHENA$=" "ELSEA$=">>>>>>>>"

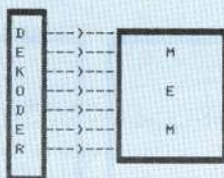
1220 SI=1:OI=58:GOSUB710
1230 SN=20:OI=121:D=2:JB=-1:T=-1:GOSUB66
0:RETURN
1240 REMXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXX
1250 REM DATA BUS VISSZA
1260 SD=10:OD=58:HD=0:SZ=41:ST=-1:GOSUB3
10
1270 SN=31:OI=40:D=1:JB=-1:T=-1:GOSUB510
:GOTO1100
1280 REMXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXX
1290 REM VEZERLO VONALAK
1300 GOSUB820:A(1)=2:A(3)=3:A(5)=1:GOSUB
780:A$="M1":SD=2:GOSUB760:GOSUB850:RETUR
N
1310 GOSUB820:A(1)=2:A(3)=3:A(5)=2:GOSUB
780:A$="MREQ":SD=4:GOSUB760:GOSUB850:RET
URN
1320 GOSUB820:A(1)=2:A(3)=3:A(5)=3:GOSUB
780:A$="RD":SD=6:GOSUB760:GOSUB850:RETUR
N
1330 GOSUB820:A(1)=2:A(3)=3:A(5)=4:GOSUB
780:A$="WR":SD=8:GOSUB760:GOSUB850:RETUR
N
1340 GOSUB820:A(1)=2:A(3)=3:A(5)=5:GOSUB
780:A$="RFSH":SD=10:GOSUB760:GOSUB850:RE
TURN
1350 REMXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXX
1360 GOSUB850
1370 REM MEMORIA KINAGYITAS
1380 SV=5:OV=47:HV=9:KV=1:GOSUB60:KV=7:G
OSUB60:KV=1:GOSUB60:KV=7:GOSUB60:KV=1:GO
SUB60
1390 SD=4:OD=51:HD=3:SZ=2:GOSUB250
1400 FORJ=3TO6
1410 SD=5-.22*J:OD=52-7*J:HD=.6*J:SZ=7*J
+1:GOSUB250:SZ=SZ-1:GOSUB200:GOSUB250
1420 NEXT
1430 CLS:GOSUB200
1440 K1=4:SZ=10:A$="RD":B$="<":SD=4:OD=6
3:T=0:GOSUB750:SD=6:A$="WR":GOSUB750
1450 PRINTE5*64+1,"-----"----";
1460 FORJ=0TO7
1470 SF=7:OF=13+J*5:HF=2:G=-1:T=0:GOSUB1
60:REMSF=0:GOSUB150
1480 SN=22:OI=27+J*10:D=2:JB=1:GOSUB620:
SN=29:JB=-1:GOSUB620
1490 NEXT:RETURN
1500 REMXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXX
1550 PRINTE13*64,CHR$(255):PRINTE14*64,C
HR$(255):PRINTE15*64,CHR$(255);
1560 PRINTE13*64,X$;PRINTE14*64,Y$;PRI
NTE15*64,Z$;RETURN
1570 REMXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXX
1580 GOSUB860:IFINKEY$("<")THENRETURN
1582 IFFTHENFORI=0TO5000:NEXT:RETURNELSE
1580

```


NOVOTRADE

A **2C** Számítástechnikai Áruházunk új szolgáltatással jelentkezik!

M
-
P
R
D
C



5000 CLS:GOSUB930



5000 CLS:GOSUB1120

ADDR. BUS



DATA BUS



5000 CLS:GOSUB990:T=0:GOSUB1060:T=0:GOSUB1120

Itt sem kell tesztelő programot bemutatnunk. Helyette néhány rajzot adunk közre, amelyek az egyes részletek meghívása után kerülnek a képernyőre.

Nem tudjuk rajzon bemutatni a memóriakinagyítás nevű részletet. Itt egy memóriarekesz először csak felvillan, majd egyre nagyobb lesz, végül betölti az egész képernyőt. Ezután rajzolódnak ki a WR-, az RD-vezetékek, a BUS-vonalak és egy címezővezeték.

NYIRATI LÁSZLÓ

Megkezdjük IBM PC XT-vel kompatibilis
COMMODORE PC 20 lízingbe adását.
Központi egysége 256 kbájt,
10 Mbájt Winchester tárral
és 360 kbájt floppyegységgel,
monokromatikus display-vel.
Dokumentáció: GW-Basic, MS DOS
Garancia és vevőszolgálat.
Széles körű szoftverválaszték.
Nyomtató lízingjére is van lehetőség.

Szoftverek:

- dACCESS relációs adatbázis-kezelő rendszer,
- FWINDOW képernyőkezelő rendszer,
- OCTOPUS relációs adatbázis-kezelő rendszer,
- Aszinkron kommunikációs adapter,
- VT 52 emulációs szoftver (PC és COMMODORE 64 között),
- Komplex főkönyvi könyvelési rendszer,
- Állóeszköz-gazdálkodási rendszer,
- Anyagnyilvántartó és számlázó rendszer,
- Bérszámfejtési, munkaügyi feldolgozási rendszer,
- Mezőgazdasági jellegű szoftverek.

Üzletünkben vásárolhat szoftvercsomagokat, könyveket, hardverkiegészítőket és mindazt, ami a számítógép üzemeltetéséhez szükséges.

Vállaljuk továbbá:

- komplett rendszerek kulcsrakész átadását, betanítását,
- meglévő szoftverek átírását,
- oktatást és szaktanácsadást,
- alkalmazói szoftverek igény szerinti kialakítását.

Az üzlet nyitva tartási ideje:

hétfőtől péntekig
9—18 óráig.

Címünk: 1136 Budapest, Balzac u. 35.
Telefon: 402-954
Telex: 22-5959, 22-7673

Strukturált programtervezés

A sorozat előző részei bemutatták, hogy a programok által feldolgozott adatok szerkezetének elemzése hogyan segíthet a program szerkezetének kialakításában. Előfordulhat azonban, hogy programunk szempontjából az adatok egymással össze nem egyeztethető szerkezetekkel rendelkeznek, azaz az adatok szerkezetei nem teszik lehetővé, hogy belőlük a cikksorozat második részében ismertetett módon egy egységes programszerkezetet alakítsunk ki.

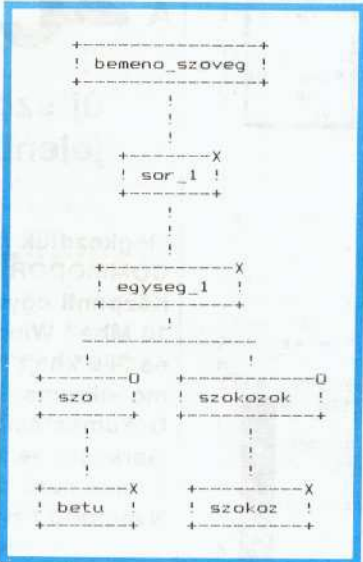
5. feladat. Készítsük el azt a programot, amely olyan különböző hosszúságú sorokat olvas be, amelyek szavakból állnak (a szavakat egy vagy több szóköz választja el egymástól, a teljes szöveg végét pedig egy üres sor jelzi). A program az így beolvasott szöveget lehetőleg egyetlen, legfeljebb 30 karakter hosszú sorokra tördeli, a sorokat egy szövegváltozóban helyezi el, ahol a sorokat egymástól „kocsi vissza” karakterek választják el. A sorok a szavakat nem törhetik el, és a szavakat a sorokban egy szóköz választja el egymástól.

Az 1. ábra a bemenő adat, a 2. ábra pedig a kimenő adat szerkezetét ábrázolja. A két adatszerkezetből nem tudunk egy egységes programszerkezetet kialakítani, mert a feldolgozást két különböző „ritmus” szerint kell elvégezni, hiszen az ábrákon lévő sor_1 és sor_2 a szavakat különböző módon csoportosítják. Első lépésként oldjuk meg a feladatot két menetben (a menetekre bontást a 3. ábra tartalmazza). Az első menet a bemenő szöveget közbenső formában tárolja. A közbenső formában az adatok egy szövegváltozóban és egy numerikus tömbben tárolódnak. A szövegváltozó a bemenő szöveg betűit tartalmazza folytonosan, szóközök nélkül. A tömb elemei az egyes sorok hosszait tartalmazzák, a tömb végét egy nulla értékű elem jelzi. A második menet ebből a közbenső formából készíti el a végleges szöveget. Az egyes menetek már probléma nélkül megtervezhetőek, hiszen az 1. ábrán levő szerkezet megfelel az első menet programszerkezetének, a második menet szerkezetét pedig a 2. ábrából kapjuk. A 4. ábra az első menet számára, az 5. ábra pedig a második menet számára tartalmazza az elemi tevékenységekkel és feltételekkel kiegészített programszerkezetet. Az elemi tevékenységek és feltételek az alábbiak (programozási nyelvként az IBM PC BASIC nyelvét használjuk):

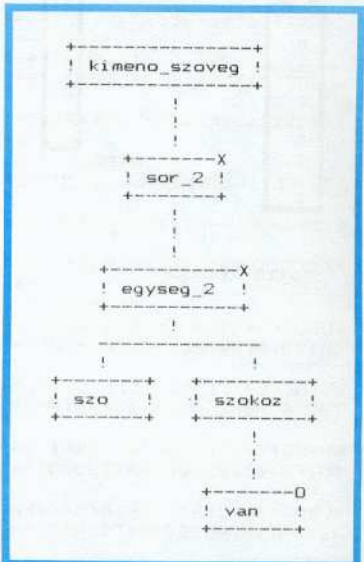
ELSŐ MENET:

Elemi tevékenységek

- 1 INPUT \$S
- 2 K=0
- 3 K=K+1:CS=MID\$(S\$,K,1)
- 4 L=1
- 5 A(L)=0
- 6 A(L)=A(L)+1
- 7 FS=FS+CS



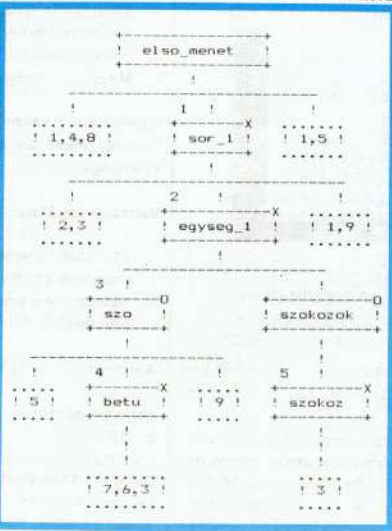
1. ábra



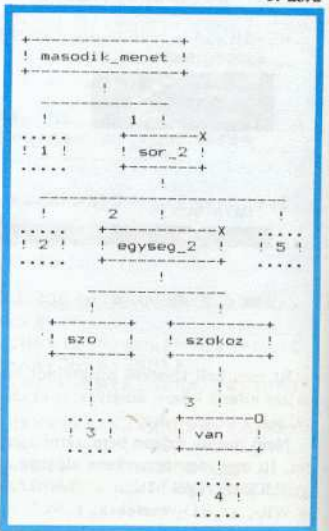
2. ábra



3. ábra



4. ábra



5. ábra

```

1 DIM A(300)
2 GOSUB 10:GOSUB 200:STOP
10 REM első menet
20 INPUT S$:L=1:F$=""
30 IF LEN(S$)=0 THEN A(L)=0:RETURN
40 K=0:K=K+1:C#=MID$(S$,K,1)
50 IF K>LEN(S$) GOTO 130
60 IF C$="" GOTO 110
70 A(L)=0
80 IF C$="" OR K>LEN(S$) GOTO 100
90 F$=F$+C$:A(L)=A(L)+1:K=K+1:C#=MID$(S$,K,1):GOTO 80
100 L=L+1:GOTO 50
110 IF C$<>" " OR K>LEN(S$) GOTO 50
120 K=K+1:C#=MID$(S$,K,1):GOTO 110
130 INPUT S$:GOTO 30
200 REM második menet
210 H=1:E$="":J=1
220 IF A(J)=0 THEN RETURN
230 H=0
240 IF H+A(J)>30 OR A(J)=0 GOTO 280
250 E$=E$+MID$(F$,M,A(J)):H=H+A(J):J=J+1
260 IF H=30 GOTO 240
270 E$=E$+" "+H:H=H+1:GOTO 240
280 E$=E$+CHR$(13):GOTO 220

```

6. ábra

```

2 GOTO 200
10 REM korutin
15 IF I=2 GOTO 50
20 INPUT S$
30 IF LEN(S$)=0 THEN A=0:RETURN
40 K=0:K=K+1:C#=MID$(S$,K,1)
50 IF K>LEN(S$) GOTO 130
60 IF C$="" GOTO 110
70 A=0:F$=""
80 IF C$="" OR K>LEN(S$) GOTO 100
90 F$=F$+C$:A=A+1:K=K+1:C#=MID$(S$,K,1):GOTO 80
100 I=2:RETURN
110 IF C$<>" " OR K>LEN(S$) GOTO 50
120 K=K+1:C#=MID$(S$,K,1):GOTO 110
130 INPUT S$:GOTO 30
200 REM főprogram
210 E$="":I=20:GOSUB 10
220 IF A=0 THEN STOP
230 H=0
240 IF H+A>30 OR A=0 GOTO 280
250 E$=E$+F$:H=H+A:GOSUB 10
260 IF H=30 GOTO 240
270 E$=E$+" "+H:H=H+1:GOTO 240
280 E$=E$+CHR$(13):GOTO 220

```

7. ábra

```

8 F$=""
9 L=L+1

```

Feltételek

```

1 LEN(S$)<>0
2 K=<LEN(S$)
3 C$<>" "
4 C$<>" " AND K=<LEN(S$)
5 C$="" AND K=<LEN(S$)

```

MÁSODIK MENET:

Elemi tevékenységek

```

1 M=1:E$="":J=1
2 H=0
3 E$=E$+MID$(S$,M,A)
4 J:=M+1:A(J):H=H+A(J):J:=J+1
4 E$=E$+" ":H=H+1
5 E$=E$+CHR$(13)

```

Feltételek

```

1 A(J)<>0
2 H+A(J)=<30 AND A(J)<>0
3 H<30

```

A két menetet eljárás-ként írhatjuk meg, amelyeket egy főprogramból GOSUB utasításokkal hívhatunk meg. A teljes BASIC szöveget a 6. ábra tartalmazza.

Megoldásunk a tárolóhely kihasználása szempontjából nem előnyös, hiszen az átmeneti forma tárolása hosszú szöveg esetében sok helyet igényel. Valójában a második menetenek nincs szüksége arra, hogy egy szó feldolgozásakor az egész szöveg rendelkezésére álljon. Az első menet úgy is dolgozhatna, hogy amikor egy szó beolvasása befejeződik, elhelyezi a szót egy szöveg-változóba, a szó hosszát egy numerikus változóba, és felfüggeszti működését. A második menet feldolgozza ezt az egy szót, majd visszaadja a vezérlést az első menetenek, amely ott folytatja működését, ahol előzőleg abbahagyta, ismét egy új szót olvas, visszatér a második menetre s. i. t. Ezt az üzemmódot úgy lehetne megvalósítani, hogy a második menetet mint főprogram eljárás-ként hívna meg az első menetet, és az első menet egy szó feldolgozása után visszaadná a vezérlést, azaz a szó és szöveg kiírását a főprogramba való visszatérés követné.

Esetünkben azonban a főprogramba való visszatérésre a BASIC RETURN utasítása nem alkalmas, mert ezt alkalmazva a következő híváskor az eljárás teljesen előlről kezdődne, ahelyett, hogy a végrehajtást ott folytatná, ahol előzőleg befejeződött (más nyelvek esetében még az eljárás változó is megsemmisülne a RETURN utasítás kiadásakor). Az első menetek nem hagyományos értelemben vett szubrutinok kell működnie, hanem egy olyan eljárás-ként, amely a belőle való kilépés után is tovább „él”. Az ilyen eljárásokat szubrutinoktól való megkülönböztetés érdekében korutinoknak nevezik. Valójában a korutin fogalma általánosabb, mert két korutin kölcsönösen is hívhatja egymást, itt viszont megmarad az első menet alárendeltsége. Az viszont igaz, hogy az az előzővel szimmetrikus megoldás is lehetséges, amelyben az első menetből lesz a főprogram és a második menetből az eljárás. A RETURN utasítások helyett a korutinokban a DETACH utasításokat használják. A DETACH utasítás hatására a vezérlés visszaadódik a hívó programba, és a kilépés után a korutin tovább él, és a következő híváskor működését az utáni a DETACH utasítás után folytatja, amelynél az előző híváskor működését befejezte.

A programozási nyelvek, és így a BASIC is általában nem tartalmaznak a fenti DETACH utasításhoz hasonló utasítást, ezért ezt csak több utasítás és egyéb nyelvi elemek együttes alkalmazásával lehet megvalósítani. A BASIC esetében például az alábbi konstrukció használható:

```

100 REM korutin két DETACH utasítással
110 IF I=1 THEN GOTO 120
111 IF I=2 THEN GOTO 220
112 IF I=3 THEN GOTO 330
120 REM a rutin algoritmusának kezdete

```

```

190 REM az első DETACH utasítás
200 LET I=2

```

```

210 RETURN
220 REM az első DETACH utasítás utáni rész

```

```

300 REM a második DETACH utasítás

```

```

310 LET I=3

```

```

320 RETURN

```

```

330 REM a második DETACH utáni rész

```

```

400 REM a korutin igazi vége

```

```

410 RETURN

```

A fentiekben vázolt korutin a GOSUB 100 utasítással aktivizálható, az első hívás előtt azonban a hívó programrészen végre kell hajtani az $i=1$ értékadást.

Az a programozási technikát, amelyben egy önálló menetenként elkészített programot korutinná alakítunk, M. A. Jackson után programinverzióznak nevezük. Feladatunknak az inverzió után kapott szövegét a 7. ábra tartalmazza. A 6. ábrán látható szövegből a 7. ábra szövege szinte mechanikusan elkészíthető, csupán az első menetenben a szó és szóhossz kiírását kell módosítani, a kiírás után a DETACH utasítást beilleszteni, illetve a második menetenben a közbenső formából való olvasást kell GOSUB utasításokkal helyettesíteni. A két szöveget összehasonlítva megállapíthatjuk, hogy az inverzió utáni szöveg az egyszerűbb, hiszen az A tömbből egy egyszerű változó lett, az F\$ szöveg-változót pedig már nem az egész szöveg, hanem csak egy szó tárolására használjuk. Ennek következtében a második menetből eltűntek a teljes szöveg és az A tömb kezeléséhez szükséges műveletek és változók.

A fentiekben ismertetett tervezési fogások az a lényege, hogy a bonyolultabb feladatokat több, egyszerűbb menetre bontjuk, az egyszerűbb meneteket önállóan megtervezzük, majd a több menetet miatt előálló hatékonysági problémákat a programinverzió alkalmazásával kiküszöböljük.

MOLNÁR MÁTÉ

Melyiket vegyem?

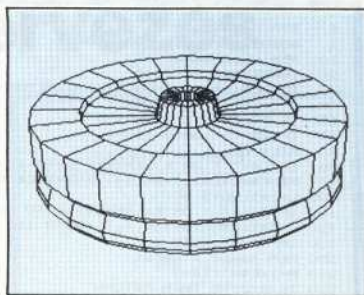
Az 1985/6. számban a legolcsóbb, ezért honfitársaink által leginkább vásárolt számítógépekről adtam egy rövid, összefoglaló értékelést.

A CHIP című, legnagyobb példányszámú NSZK-szaklap 1985. novemberi száma az ún. háziszámítógép-kategória tíz gépét értékeli (nem a legnagyobb példányszám-ban eladottakat, bár hat ezek közül való). Az értékelést, mivel a néhány gépnél kiegészítő információt ad, táblázatos formában, átlagértékkel, és az ugyanott megadott

árral együtt közöljük. Az ár alapján látható, hogy öt nem tartozik az általunk vizsgált kategóriába, ezeket csak a teljesség kedvéért szerepeltetjük itt. A táblázatban szereplő osztályzatok, az itthoni iskolai osztályzatokkal megegyezően, 1–5 számmal megadott értékek.

A DRAGON BASIC „rajztudásának” szemléltetésére bemutatok még egy alkotórésznevezetről (kollégám által készített) rajzolóprogram által felrajzolt nézetrajzpárt.

D. S. E.



ÉRTÉKELÉS	ATARI 130XE	LASER 500/700	C16	C64	C128	SCHNEIDER CPC 464/664/6128	SINCLAIR QL	SINCLAIR SPECTRUM +	SPECTRA VIDEO SVI 728	TRIUMPH-ADLER ALPHATRONIC PC
TELJESÍTMÉNY	4	4	3	3	4	4	5	3	3	3
BERENDEZÉS	3	3	2	2	4	3	3	2	3	4
SZOLGÁLTATÁSOK	3	3	2	2	3	3	3	2	3	3
DOKUMENTÁCIÓ	3	3	3	2	3	4	2	3	3	3
SZOFTVERAJÁNLAT	4	2	2	5	4	2	1	2	2	2
ÁR/TELJESÍTMÉNY	5	4	3	4	3	4	4	3	3	3
ÁTLAG	3,7	3,2	2,5	3,0	3,5	3,3	3,0	2,5	2,8	3,0
ÁR	600	400	330	600	1000	900–2100	1700	550	700	1000



ORTEKON TERVEZÉSI ÉS FEJLESZTÉSI gazdasági munkaközösség
1036 Bp., Árpád fejedelem útja 69.

Telefon: 154-250, 887-861
Bányai

Vállaljuk:

— elektronikus (digitális) vezérlőegységek, szabályozókészülékek és -rendszerek tervezését és gyártását;

— CNC-vezérlés komplett felújítását;

— műszerek, villamos és elektronikus ipari termékek konstrukciós és technológiai tervezését, komplex felszerszámozását a sikeres „0” sorozatig;

— a gyártás-előkészítés és elszámolás irányítási és adatfeldolgozási feladatainak megvalósítását több munkahelyes számítógéprendszer alkalmazásával.

A megfelelő referenciákról személyesen adunk tájékoztatást.

MIKRÓK A CENTRUMBAN

Kereskedni is tudni kell!



— Mit adtak a gépekért?

— Ez most nem fontos, de az ügylet ilusztrálására elmondok egy igaz történetet. A legutóbbi BNV-n egy nyugati üzletember öt tonna lóéletet vett volna tőlünk. Körülnéztünk, nem volt hazai kínálat, csak lóval együtt, így viszont partnerünknek nem kellett, helyette ezért almalet kínáltnak, ám az sem nyerte meg tetszését, viszont szívesen vett volna nagy tételben olcsó ébresztőórát.

— A lényeg tehát: jönnék a gépek. Áruk?
— Az említett négy egység ára összesen 141 500 forint.

— Igen, hazai viszonylatban meglehetősen olcsó. Egy-egy boltban ugyanezt 304 ezerért láttuk, aki pedig lízingbe vette, az 840 ezret fizet ki érte. Altruista cselekedetnek látszik tehát ilyen olcsón kitenni ezt a gépet a pultra.

— Szó sincs róla, így is van rajta hasznunk szépen, de nem ezen akarunk megzavargodni. Egyébként a gép nem kerül a pultra, mert elsősorban a kereskedelem területét akarjuk előltni.

— Mi ez? Diszkrimináció?

— Szó sincs róla! Ám a vásárlást a tárcia is támogatta, s ez volt a kikötése, másrészt mindenkinek nem tudunk alkalmazási programokat kínálni, mertelenül eladni a gépet pedig belföldi szakmai partnerünk tanácsára nem akartuk.

— Nocsak!

— Igen, mert mi nem értünk a számítástechnika-alkalmazáshoz, s a jövőben is megmaradunk a mi kaptafánkánál. Belföldi partnerünk, a Kereskedelmi Szervezési Intézet, közismert nevén Kerszi dolgozott ki belkereskedelmi alkalmazási programokat, amelyeket a géppel együtt kínálnak. Ugyancsak a Kerszinél lehet megtekinteni a gépet működés közben is.

— Mi lesz a szervizzel, alkatrész-utánpótlással, garanciával?

— A géphez egyéves garancia jár, mely külön megállapodással fél évvel meghosszabbítható. A garanciális javításokat egy kiszívókezet végzi, s vállalkozik a későbbi karbantartásra és javításra is. Az alkatrészt a szállító folyamatosan biztosítja, de a Commodore cég képviselője is ügy nyilatkozott, hogy — bár a gépek nem közvetlenül tőlük származnak — a javításhoz szükséges alkatrészeket tőlük is beszerezhetjük.

— Milyen az érdeklődés a gépek iránt?

— Meglehetősen élénk, s ez annak is köszönhető, hogy a konfiguráció darabonként is megvásárolható.

— De mit kezdjek egy nyomtatóval, csak úgy magában?

— No, nem is erre gondoltunk. Természetesen a legtöbb felhasználónak mind a négy egységre szüksége lesz, ám ezek egyike az ára sem haladja meg az ötvenezer forintot, így — s erre egy ezzel kapcsolatos pénzügyminisztériumi leirat a garancia — azok tartós fogyóeszköznek minősülnek, az érdeklétési állapot nem terhelik, vagyis a vállalatok „olcsó pénzből” juthatnak a géphez.

— Mégis: hétszázötven gép! Kimondani is sok! Tudják ezt majd használni is?

— Minden vevőt betanítunk. Egy-egy géphez egy fő oktatása minden további térítés nélkül jár. A tanfolyamokat a PM Továbbképző Intézet tartja, időtartamuk negyven óra, s a résztvevők gépkezelői oklevelet is kapnak.

— Mire használják majd a gépet a Centrumban?

— Ó, rengeteg lehetőség van, egyet azért konkrétan is kiemelek. Az utóbbi években megszorodott az áruházi lopások száma, különösen a műszaki osztályokon. Ott vannak ugyanis olyan drága, viszonylag kis fizikai terjedelmű, elektronikus készülékek, amelyekre egyeseknek felettsébb fáj a foga. Hi-Fi-tornyok, magnók, videorekordersek, számítógépek. Ezek szám szerinti egyedi nyilatvántartása óriási segítség az ellopott készülékek azonosításában. Ezt eddig kézzel nem tudtuk megoldani.

— Egyéb tervek? Mit várhat az egyszerű vásárló?

— Gondolkozunk azon is, hogy a lakosság részére is árusítsunk számítástechnikai termékeket. Forr a világ, „bézikül” tanul az ország, a fiatalság issza a technikai újdonságokat. Ezzel nekünk is számolni kell!

— Ezzel is, meg a számítógépekkel is. Sok szerencsét hozzá a Centrumnak, s a vásárlóknak is.

Ha valakinek ma azt mondják, hogy centrum, akkor először a Centrum Áruházak valamelyike jut az eszébe, aztán erről mindjárt a Centrum hétfo, vagy a legnagyobb Centrum, a Corvin, ahol „minden szinten szinte minden” — s legkevésbé sem a mikroszámítógépek. Bár az utóbbi Szolnokra nem biztos hogy áll, mert ott már tavaly óta folyik a személyi számítógépek bizományi értékesítése. Sőt, figyelmes olvasók már a napilapokban is felfigyelhettek arra a hirdetésre, mely szerint a kispesti Centrum különleges műszaki cikkek — így személyi számítógépek — bizományosi adásvételét kezdte meg. Újabb viszont azt csiripeltek a verebek, hogy a Centrum nagyobb mennyiségű Commodore 610-est vásárolt. Erről az üzletről kértünk bővebb tájékoztatást Kaszás Kálmántól, a Centrum Áruházak Vállalat vezérigazgatójától.

— A Commodore cég neve Magyarországon évek óta összefonódott a Novotrade-ével, a tavaly karácsonyi C-16-os vásárt az APISZ tette népszerűvé, s most a Centrum is bekapcsolódik a versenyfutásba. Van ennek értelme?

— Már hogyan lenne. Előrebocsátom, hogy egyáltalán nem kerestük a Commodore cég kegyeit, a gépeket nem is tőlük vásároltuk, magyar részről viszont az arra illetékes és — remélhetőleg — hivatott kereskedelmi vállalat bonyolította le az üzletet, az ösznapi magyar érdekek így minden bizonnyal hivatalból érvényesültek.

— Mégis, mi adta az ötletet?

— Nevetni fog, elsősorban saját belső adatefeldolgozási igényeinkre kerestük a megoldást. A Centrumnak ugyanis harmincnegyevenyolc áruháza van, köztük az évi két-milliárdos forgalmú Corvin és a csupán 170 milliósi Székesfehérvár. S a harmincnegyevenyolc áruház mindegyikében sok-sok osztály! S ezek mindegyike már 1966 óta gép nélkül, hegyes ceruzával készíti el a mérleget! A vállalati szintű adatefeldolgozást ugyan bér munkában végteztük, s nagykereskedelmi részlegünk számára újabban egy nagyobb számítógépet is vásároltunk, amelytől rengeteget várunk, a napi forgalom részletekbe menő feldolgozását csak az egységekre telepített eszközökkel lehet megoldani.

— Miért nem vásároltak a hazai piacról?

— Nézzel! Egy Proper nekünk túl drága, egy C-64-es pedig ...

— Inkább játékszer, amelyből termelőszekét csináltak. Hiába, ügyes menedzserek egy akadémia elnökét is meg tudják buktatni, s a egy paprikajancsiból is tudnak VALAKIT csinálni. Nem ez a helyzet az Önök által választott hasztávítással is?

— Valahol el kellett kezdeni! Úgy döntöttünk, hogy negyven személyi számítógépet állítunk be saját részlegeinkbe, s miközben körülnéztünk a piacon, egy nyaratémet cég dőmping áron felkínált 750 darab C-610-est, képernyővel, egymegás floppyval és margarétakeresek nyomtatóval együtt, méghozzá számunkra előnyös barter ügylet keretében.

— ???

— Egyszerű árucseréről van szó.



Termékei:

1. 3 1/2 és 4 1/2 digit kijelzésű multiméterek általános használatra.



2. Különböző laboratóriumi tápegységek 220 V bemenetről 0—40 V egyenfeszültség és 0—10 A terhelhetőséggel különböző változatban.

3. Rajzdigitalizáló berendezés: olyan feladatok megoldására alkalmazható, amelyekben grafikus, vagy képi információt kell átalakítani számítástechnikai feldolgozásra alkalmas formába.



Műszaki adatok:

Munkafelület (mm) 1050 × 675

Felbontóképesség (mm) 0,1 ± 0,02

Munkafelület méretének hőfokfüggése $2 \cdot 10^{-5}$ mm/°C

Letapogatási sebesség max. 3,25 m/s

Számkijelzés X és Y irányban: mm-es mértékegység esetén ± 4 egész szám és egy tizedesjegy, inch mértékegység esetén ± 2 egész és 3 tizedesjegy

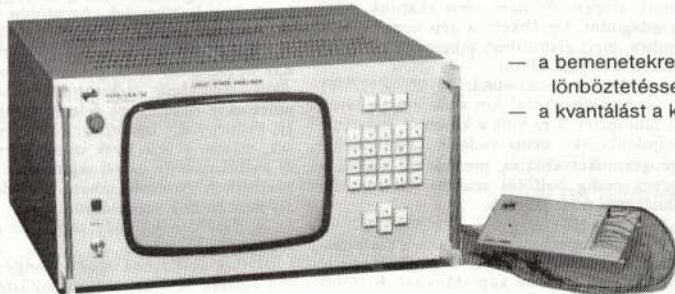
Digitalizálható anyag

max. 1 mm vastag nem mágneses anyag

Pozicionáló eszköz

szabadon mozgatható tekerecs, középen hajszálkereszttel.

4. Logikaiállapot-analizátor: mikroprocesszoros és szinkronrendszerek funkcionális vizsgáló műszere, felhasználható bármely digitális rendszer, kombinációs hálózat, szekvenciális hálózat, sínrendszer működésének tesztelésére.



A készülék szolgáltatásai az alábbiak:

- a bemenetekre kapcsolt jeleket logikai „0” és „1” megkülönböztetéssel kvantálja max. 10 MHz órajelfrekvenciával;
- a kvantálást a készülék külső (EXT), vagy belső órajel hatására végzi el, az órajel homlokleténeli;

— a 32 bemeneti csatornán érkező TTL, ECL stb. szintű impulzussorozatból a készülék a kiválasztott logikaiállapot-sorozat szakaszt tárolja, saját display egységén megjeleníti.

A jelfelvételi tár hossza 1024 bit csatornánként.

EGY JÓ DÖNTÉS

AZ IBM KOMPATIBILIS

PROPER—16

professzionális személyi számítógép

BŐVEBB PROPER-VÁLASZTÉK!

KORÁBBAN AZ SZKI-TÓL VÁSÁROLT GÉPEIBŐL SZEMÉLYI SZÁMÍTÓGÉPES HÁLÓZATOT ALAKÍTHAT KI.

RENDKÍVÜL JÓ ÁR/TELJESÍTMÉNY VISZONY!

A PROPER—16-OS HÁLÓZAT MEGTARTJA SZEMÉLYI JELLEGÉT, DE BIZTOSÍTJA A KITERJESZTETT ERŐFORRÁSOK KÖZÖS HASZNÁLATÁT. 12 TERMINÁLIG ALACSONYABB ÁRON NYÚJT NAGYOBB TELJESÍTMÉNYT, MINT EGY MINISZÁMÍTÓGÉP.

NINCS SZÜKSÉG KÜLÖN SZEMÉLYZETRE, KÜLÖN GÉPTEREMRE, KLIMATIZÁLÁSRA!

NAGYOBB KATEGÓRIÁJÚ GÉPEKKEL ON LINE, ILLETVE OFF LINE KAPCSOLAT LEHETSÉGES.



**SZÁMÍTÁSTECHNIKAI KUTATÓ INTÉZET
ÉS INNOVÁCIÓS KÖZPONT
1015
Budapest I., Donáti u. 35—45.**

**Információ: SZÁMÍTÁSTECHNIKAI INFORMATIKAI
FEJLESZTŐ
LEÁNYVÁLLALAT (SCI—L)
1011 Budapest, Iskola u. 10.
Telefon: 350-180**

A COMMODORE 610



Ez már nem játék!

A belkereskedelmi vállalatok számítástechnika-alkalmazási kultúrájának fejlesztésére rendkívül kedvező áron, nagy számban értékesítették CBM 610 típusú mikrogépeket. Lássuk a géptípus főbb jellemzőit.

Az alapgép egybeépítve tartalmazza a mikrogépet, a tasztatúrát és a tápegységet. Az alkalmazott mikroprocesszor a MOS 6509 (6502 kompatibilis) 2 MHz órárfrekvenciával. Ez 64 kb-át memória címzésére alkalmas. A gép alapkonfigurációban 128 kb-át RAM memóriát, valamint 24 kb-át ROM memóriát tartalmaz. A ROM-ban helyezkedik el a Kernal, 4 kb-ajnyi karakterterület és a BASIC interpreter.

Az alkalmazott BASIC 4.0 verzió és a Kernal úgy van felkészítve, hogy a felhasználható RAM területet 64 kb-ajtos ún. BANK-ba szervezi. BASIC program csak az 1. BANK-ban futhat, a 2-3-4-es BANK-okba a változók és az adatok kerülnek. A 128 kb-ajtos verzió csak az 1-es és 2-es BANK-ot használja. A tár 256 kb-ajtra bővíthető. A 3-as és 4-es BANK POKE, illetve PEEK utasítással érhető el. BASIC verzió cserével megoldható, hogy maga az interpreter is felismerje a további BANK-okat. Mivel a ROM-ban csak 4 kb-ajnyi helyet foglal el a karakterkészlet, alternatív karakterkészlet választható egyetlen utasítással. Ehhez természetesen előzőleg el kell helyezni a karakterkészletet. A karakterek 8x8-as pontmátrixban jelennek meg, soronként 80 karakter és 25 sor képernyőnként.

A billentyűzet tartalmazza a teljes ASCII készletet, 10 (20) programozható funkcióbillentyűt, az ESC billentyűt és különválasztva egy plusz numerikus billentyűmezőt az adatbevitel megkönnyítésére. A jelszót kiegészíti a Commodore grafikus jelszótáblával. A beépített EDITOR az ESC billentyű segítségével „ablak” definiálást, sorkövetést, beszúrás és egyéb szerkesztési könnyítéseket tesz lehetővé.

A Commodore 64-hez hasonlóan itt is beépítettek egy 6581-es hanggenerátor IC-t, erre azonban ritkán van szükség, mivel ez a gép nem játékgépek futtatására készült.

Az interfészek

A modulcsatlakozó 24 kb-ajt RAM/ROM vagy egyéb periféria csatlakoztatására alkalmas.

A **monitorkimenet** csak monokróm videójel-t szolgáltat.

Az **audiokimenet**: a hangszintetizátor azon kívül, hogy egy kis hangszórót működtet, további felhasználásra hozzáférhető.

A **zsettás magnetofon port** ugyanolyan, mint a C64-nél, de csak a fizikai csatlakoztatást teszi lehetővé, a működtető rutinokat nem építették be a Kernalba.

Az **RS232C (V. 24.)**: a C64-tól eltérően itt „igazi” soros illesztőt alakították ki a 6551 típusú aszinkron kommunikációs adapter segítségével. Így megfelelő szoftverrel modem vezérlésre is lehetőség van.

Az **IEEE-488 busz**: ez a csatlakozó felület szolgál a legtöbb periféria csatlakoztatására. A rendszerhez kapható háttértárolók (SFD 1001, 8250) és a nyomtatók is (1361, 8023, 8028) mind IEEE-488 illesztővel rendelkeznek. Mivel ez a HP által definiált buszrendszer terjedt el a mérés-technikában is, ez a gép felhasználható az olyan rendszerekben, ahol alapkövetelmény a mérés-adatgyűjtés-kiértékelés számítógépes megvalósítása.

Mivel a gép kisebb Commodore-típusokon alkalmazott soros IEC busszal nem rendelkezik, és az SFD 1001 floppy jelrögzítése más, mint az 1541-é, ezért a már meglévő programok közvetlenül nem vihetők át erre a rendszerre.

A szoftver

A Commodore 610-es alapszoftverről elmondhatjuk, hogy a legkorszerűbb alapszoftverekhez tartozik az operációs rendszer nélkül működő személyi számítógépek között. Fejlettsége a monitorprogramnak és a BASIC 4.0 verziójának köszönhető. Ezek fontosabb jellemzői a következők:

A **monitorprogram** segítségével közvetlenül assembler mnemonikus kódban programozható a gép, így nincs szükség az egyes gépi kódú utasítások ismeretére. Megjegyezzük, hogy közvetlenül is írhatjuk gépi kódban a programokat. A monitor a sys 60950-nel hívható. Meghívásakor automatikusan megjelenik a szokásos processzorregiszterek tartalma (PC, SR, AC, XR, YR, SP), így megfelelő szintaktikákkal elmenthető és később visszahozható a leírt program.

A **BASIC 4.0 verziója** felülről a C64 BASIC 2.0 verziójával kompatibilis. Bővítés-ként több utasítás és parancs mellett a CP/M lehetőségeit foglalja magába. Mivel a gépet nem játékra, hanem adatfeldolgozásra szánták, a C64-gyel ellentétben a Commodore 610-es nem ismeri a színkezelést. A MOS 6581 processzor segítségével viszont hangokat generálhatunk.

A BASIC 4.0 fontosabb utasításbeli újdonságai a következők:

A **DISPOZE** (a verem tisztítása) utasítással a veremben már nem használatos információit töröljük. Főleg a hibakezelés során használjuk.

A **PRINT USING** utasításnál a négy leggyakrabban használt jel menet közbeni átírására a PUFDEF utasítás szolgál.

A **RESUME** (folytatás) utasítás hatására a program a hiba lekezelése után folytatódik. Ez az utasítás a TRAP utasítással együtt működőképes.

A **TRAP** (hibakezelés) utasítás (más számítógépen ON ERROR-nak felel meg) lehetővé teszi, hogy a programfutás során fellépő hibákat magunk kezeljük.

A **WAIT** (várakozás) utasítás segítségével pontosan követhető a számítógép működése, mert ez az utasítás egy memóriapozíció tartalmát vizsgálja.

Egy bemutató margójára

Az új parancsok közül a következők fontosak még:

A BACKUP (másolás) hatására egyik lemezzel a másikra készíthetünk teljes másolatot.

A BANK alkalmazásával kiválaszthatjuk a felhasználó memóriaszegmenst.

A BLOAD a lemezen levő információt betölti a kiválasztott BANK-ba.

A BSAVE a BANK-ban levő információt kiment a lemezre.

A CATALOG parancs a lemez tartalomjegyzékének kiírására szolgál.

A COLLECT segítségével felszabadítjuk azokat a blokkokat, amelyeket foglaltként jelez a rendszer. Röviden: ez a parancs átrendezi a lemez blokkjait.

A CONCAT két adatfájlt fűz össze, hasonlóan a MERGE utasításhoz, amely két programfájlt fűz össze.

A DCLEAR parancs a nyitott fájlokat automatikusan lezárja.

A HEADER a lemez formálására szolgál. Összesen 4133 blokk áll rendelkezésre.

A KEY (kulcs) parancs segítségével húsz funkcióbillentyűt tetszőlegesen programozhatunk. Addig maradnak érvényben a parancsok, amíg át nem írjuk őket. Természetesen bekapcsoláskor minden esetben a gyárilag generált parancsok, utasítások élnek.

A RECORD (beállítás) segítségével relatív fájlnál bármikor beállhatunk bármelyik adata és bájtpozícióra.

A programozás

A CBM 610-en BASIC-ben és Assembly nyelven írt programok futtathatók. A BASIC — 32767 és 32768 közötti egész, valamint 10E+38 és 10E-39 közötti lebegőpontos számjegyeket kezel. Sajnos a duplapontosság változó használata nincs lehetőség. Legfeljebb 3 dimenziós numerikus vagy karakterláncot használhatók. Az OPEN utasításban a fájlok sorszáma 1—255-ig terjedhet. Arra, hogy egyszerre hány láncot nyithatunk meg, nincs utalás. Elvileg például 1 relatív és 2 szekvenciális fájl is megnyitható egyidőben. Az IEEE—488 busz alkalmazásokról csak a 4—31 egyéjszámokat használhatjuk.

A dokumentáció

A gép kézikönyve tartalmasabb, mint a C64-é. A gépre vonatkozó fejezetek felépítése a szokásos. Ezt igen részletes BASIC 4.0 leírás követi, amely önálló BASIC könyvként is megállja a helyét (lásd a Novotrade RT. által kiadott magyar nyelvű kézikönyvet). Említésre méltók a gyári szoftverek: a CP/M, a CP/M—86, az U.C.S.D. Pascal; a KERSZI-nél kifejlesztett felhasználói programcsomagok közül a C610 felhasználását biztosító segédprogramok, valamint az adott géptípus lehetőségeit optimálisan kihasználó Quick-sort (gyorsrendező) program.

Nagy sikerű bemutatót tartott az OPTIMUM Szervezési, Számítástechnikai és Innovációs Kiszöveket és az SG—2 Pénzügyi Informatikai Kft. 1986. január 29-én a Béke Szállóban.

A bemutatón működésük közben láthatuk azokat a szoftvertermékeket, melyeket a két számítástechnikai szervezet közösen, vagy külön-külön fejlesztett ki és értékesített.

Feltűnő érdeklődést keltek a takarékoszövetkezetek számára készített programrendszerek, melyek lefedik az eddig manuálisan vagy középgepeken végzett munkafolyamatokat. Látnyoson és gyorsan bonyolódott le a kolléganők által kezelt gépeken a betétforgalom minden fázisa, új betétkönyv nyitása, forgalombonyolítás, kamatszámítás, tőkésítés, leltárak készítése.

A hitelrendszer a bírálattól kezdődően a folyósításra keresztül végigkíséri a hitellezés folyamatát egészen a hitelek lejártáig. Figyelemmel kíséri és listát készít a hátralékosokról. Az év végi tevékenységeket könnyíti meg azáltal, hogy elkészíti a hitelleltárt, valamint a tőkésítést. A rendszerek bemutatóját nagy érdeklődéssel nézte végig az OTP magas rangú képviselője is...

Nem fogyott ki a hallgatóságából a „GESZTOR” főkönyvi könyvelési rendszert bemutató munkatárs sem. „Fogadom van rá — mondja az OPTIMUM kiszöveket elnöke —, hogy közepes képességű betanított munkatárs, aki még életében nem látott számítógépet, 30 perc alatt kezeli a rendszert”. A tájékoztatás szerint a „GESZTOR” eddig 17 gazdálkodó szervnél került bevezetésre.

A FER számára készített havidíjas bérszámfejtési rendszer — mely üzemszerűen működik — olyan részletettségű dolgozza fel az adatokat, ahogy azt a vállalati bérfőnök (havidíjas, órabéres, nyugdíjas, egyéni és csoportos jutalékos, tanuló, állományon kívüli) megkívánja.

(Elgondolkodtat, hogy hazánkban a bérfizetés módozatai számtalan variációban lehetők fel. Ily módon egységessé válhat még álmodni sem lehet.)

Örömmel tapasztaltuk, hogy a bemutató játékosprogramokat futtató gépeket nem

látunk. Oktatóprogram viszont volt, mégpedig egy topográfiai készséget fejlesztő földrajzi program.

A bemutató — jellemzően a magyar számítógéppiacra — különféle számítógépeket láthatunk (COMPUT—80 két db 1 Mbájtos floppyval, COMPUT—80 2 Mbájtos Winchesterrel, 4 munkahellyel, MXT gépek a Műszertechnikatól, Commodore 64 1541-es és 1001-es floppyval, TEXAS INSTRUMENT PC, SIEMENS PC, CAF Kollega PC, PC—420, IBM—XT, Sinclair Spectrum).

Oldalak lehetne írni az itt nem részletezett működő rendszerekről (konszignációs raktárelszámoló, vezetői információs program, hálótervezési program, termelésirányítási program, kereskedelmi vállalat teljes vertikumát feldolgozó programrendszer stb.), de dr. Kádár Iván, az SG—2 igazgatója és Herskovits György, az OPTIMUM elnöke tájékoztatása szerint az elkészült rendszerek bármikor megtekinthetők a vállalatok telephelyein, és rendszeresen tartanak kevés számú meghívott részére szakmai bemutatókat.

„Várjuk Önöket” — zárja a bemutató „tárlatvezetését” a két vezető, s mi pedig gratulálunk a színvonalas, jól szervezett és szemmel láthatóan nagy érdeklődést kiváltó eseményhez.

SG—2: A Magyar Kűterkereskedelmi Bank (MKB) és a francia Société Generale de Service et Gestion (SG—2) közös vállalta, amely 1983 októberében alakult. Tevékenységi köre: pénzügyi információs rendszerek szervezése, az informatika és az üzemszervezés körébe tartozó tanácsadás, pénzügyi alkalmazási programcsomagok készítése, a felhasználók képzése. Cím: Budapest XI., Bartók Béla út 7. Budapest 4. Pf: 146, 1364. Telefon: 868-999, 868-940, 868-182. Telex: 7744.

OPTIMUM: Szervezési, Számítástechnikai és Innovációs Kiszöveket, 1983 februárjában alakult. Profiljába tartozik bármely népgazdasági ágba tartozó gazdálkodó szervezetek számítástechnikai rendszerterveinek, programjainak elkészítése, szükség szerint azok üzemeltetése. Cím: Budapest XIII., Kárpát u. 54. Budapest 4. Pf: 206, 1364. Telefon: 496-706, 495-312.

ADOK — VESZEK — CSERÉLEK

Ebben a rovatban rövid, szöveges, a mikro-számitógépekkel kapcsolatos hirdetések közlünk. A díjszabás: közületeknek gépelt soronként (60 karakter) 100,— Ft, magánszemélyeknek az első sor 50,— Ft, minden további sor 20,— Ft. Az NJSZT tagjainak az első három sor ingyenes. Hirdetéseiket a szerkesztőség címére várjuk.

● **C-128 PROGRAMOKAT** és dokumentációt szeretnék cserélni. Drahovszky Zoltán, 1066 Bp., Jókai u. 18. II. 11. Telefon napközben: 667-029

● **KERESEM** a Mikroszámítógép Magazin 1984/5. számát, cserébe az 1985/1. számát tudnám adni, vagy az újság többszörös árát. Járóka László, 1063 Bp., Sziv u. 3—5. fszt./5.

● **C 16-OS** játék- és egyéb programokat cserélnék. Barta István, Eger, Kodály u. 10. 3300

● **1 K-S ZX 81** magyar nyelvű gépkönyvet, sok programmal 700 Ft-ért eladó. Telefon: 496-294, este.

Meteorológiai program C64-re

A program Magyarország húsz nagyvárosában mért hőmérsékletadatokat tárolja és dolgozza fel. Többféle üzemmód közül választhatunk. Valamennyi esetében — külön kérésre — a képernyőtartalmat nyomtatóra is küldhetjük. A program futtatásához C64-es alappépre, VC—1541 lemezegységére és egy MPS—801 vagy azzal kompatibilis nyomtatóra van szükség.

A választható üzemmódok:

1. Egy adott hónapban és időpontban belül — ez lehet 7 vagy 13 óra — egy tetszőlegesen kiválasztott városban mért adatok képernyőre iratása. Kiíródnak a megadott időpontban mért átlag-, minimum- és maximumhőmérsékletek is (1. ábra).

2. Kiírathatjuk az egy konkrét időpontban mért hőmérsékleti adatokat a városok neve mellé (2. ábra).

3. Az egyik leglátványosabb üzemmód a térképes forma, bár csak 7 helység: Budapest, Sopron, Nagykanizsa, Pécs, Debrecen, Békéscsaba és Kékestető hőmérsékletét írja ki (3. ábra).

4. Összeadhatjuk az egy városban egy hónap alatt mért adatokat, és ebből átlagot

számitathatunk. Ugyanaz, mint az első üzemmód, csak itt az összes átlag egyszerre látható.

5. Oszlopdiagramot is rajzoltathatunk a géppel (4. ábra).

6. Ez az üzemmód ad lehetőséget arra, hogy a legfrissebb adatokat — amennyiben valahonnan beszereztük vagy hallgattuk a rádióban — mi magunk is lemeze írhatjuk. Ha mindezt szabályosan írjuk és egy teljes hónap adatait betápláltuk, akkor az általunk létrehozott adatállomány is feldolgozható.

A térkép és az oszlopdiagram a speciális karakterek segítségével készült, tehát nem nagy felbontású grafika! Ugyanis ha azt alkalmazzánk, a program rendkívül bonyolulttá válna és lelassulna. A térképnel minden kiírási ponthoz külön meg kellett adni a kezdő koordinátákat, hogy a hőmérsékleti adatok a képernyőn a megfelelő helyre kerüljenek.

Miután a C64-en nincsen PRINT AT utasítás, erre külön szubrutint kell írunk. Ezt megírhatjuk BASIC-ben, de gépi kódban is. A BASIC szubrutin:

```
1000 PRINT"R"
1001 IFY=0THEN1005
1002 FORI=0TOY-1
1003 PRINT
1004 NEXT
1005 PRINT"!"TAB(X);VAL TOZOK
1006 RETURN
```

A szubrutin meghívása előtt meg kell adnunk x és y változó értékét. Ez 0-tól 39-ig, illetve 0-tól 24-ig terjedhet.

A nyomtaton minden kiírást duplaszélés karakterekkel jelenik meg; így sokkal olvashatóbb, és a papírt teljes szélességében kihasználjuk. A duplaszélés nyomtatást a következőképpen érhetjük el:
OPEN 4:PRINT #4,CHR\$(14)

A hőmérsékleti adatok a programban belül egy DIM T(20,31) tömbben vannak tárolva. A 20 a városok számát, a 31 a napok számát jelenti. Ez esetenként lehet 30, 28 vagy 29. Az adatok a lemezen szekvenciális állományokban vannak. Egy teljes hónap két állományban van, külön a 7 és 13 óra.

1. ábra

1985. MARCHIUS 07 ORA SZEGED										
1	2	3	4	5	6	7	8	9	10	
-4	-2	0	1	0	2	0	1	1	0	

11	12	13	14	15	16	17	18	19	20	
0	-2	-1	-1	1	2	2	4	5	5	

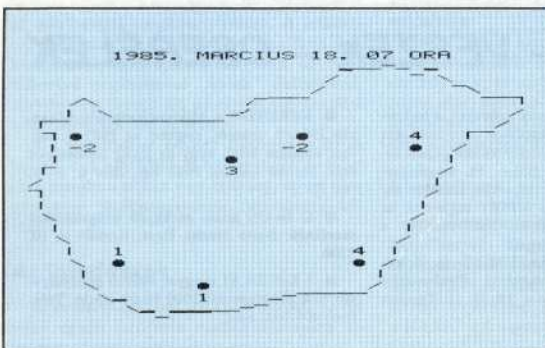
21	22	23	24	25	26	27	28	29	30	31
3	4	4	4	3	2	5	8	1	2	1

ÁTLAG: 1.64516129 CELSIUS FOK										
A MINIMUM: -4 CELSIUS FOK VOLT										
MARCHIUS 1. NAPJAN										
A MAXIMUM: 8 CELSIUS FOK VOLT										
MARCHIUS 28. NAPJAN										

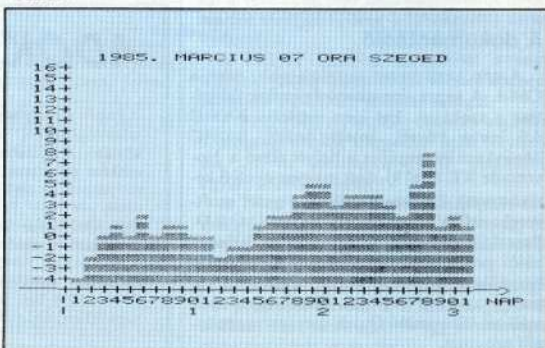
2. ábra

1985. MARCHIUS 18. 13 ORA	
VÁROS	CELSIUS FOK
BUDAPEST	1
GYOR	1
SOPRON	1
SZOMBATHELY	1
FAPA	1
ZALAEGERSZEG	1
KESZTHELY	1
NAGYKANIZSA	1
SIOFOK	1
PECS	1
PAKS	1
BAJA	1
KECSKEMET	1
MISKOLC	1
SZOLNOK	1
DEBRECEN	1
NYÍREGYHÁZA	1
SZEGED	1
BEKÉSCSABA	1
KEKESTETO	1

3. ábra



4. ábra



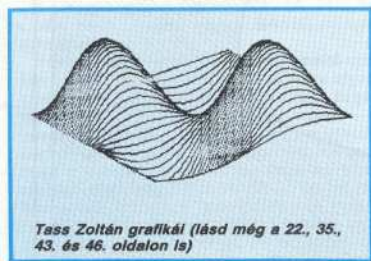
Le kellett tiltani a RUN/STOP billentyűt, mert ha olyankor állítjuk le a programot, amikor épp egy fájl nyitva van, akkor ez az adatok elvesztéséhez vezethet. A letiltás a POKE 788,52 utasítással érhető el, mely a C64 IRQ vonalának kezdőcímet EA31-ről EA34-re állítja. Ez kikerüli a RUN/STOP billentyű vizsgálatát. A STOP/RESTORE együttes leütése a bájttérteként az eredeti (49) értékre állítja vissza.

A program felépítése

- 2— 120 adatbeolvasás a szekvenciális állományokból
- 120— 400 egy városos havi hőmérsékletek + minimum, maximum és átlaghőmérséklet nyomtatási szubrutinok (gépi kódú hcopy meghívása)
- 6000— 6002 a gép alaphelyzetbe állítása (kilépés a programból)
- 6999— 8000 Magyarország-térkép a printer számára
- 8000— 9000 a program alaphelyzetbe állítása (visszatérés a kezdéshez)
- 9000— 9050 táblázatmagyarázat szubrutin
- 9050— 9100 minimum-, maximumszámítás
- 9100— 9200 1. üzemmódváltás
- 9200— 9300 egy konkrét időpontban mért adatok kiírása (a 20 város mellé az adatok)
- 9300— 9450 térképes üzemmód
- 9450— 9600 2. üzemmódváltás
- 9600— 9630 a hús város átlaga egyszerű
- 9630— 9700 egy város átlaga
- 9700— 9900 oszlopdiagram üzemmód
- 9900— 10000 lemezre írás üzemmód
- 50000— 50012 diszkhibra esetén meghívott szubrutin

A program még két gépi kódú szubrutint is tartalmaz C000 memóriától kezdve. Az egyik egy hcopy (képernyőtartalom ki-nyomtatása nyomtatóra), a másik egy PRINT AT utasítást helyettesítő rutin. A két program együtt sem több, mint 1 blokk a lemezen. A teljes program terjedelmét meghaladja a 10 kb-át nagyságot, ezért listáját nem közöljük. Érdeklődők a HCC Commodore Szekciónál saját lemezükre átmásolhatják.

KOCSIS TAMÁS



Tess Zoltán grafikai (lásd még a 22., 35., 43. és 46. oldalon is)

Micsoda kiszolgálás! 2.

Élvezettel olvastam a Mikroszámítógép Magazin 1985. évi 6. számában e címen megjelent derűs kis cikküket, amelyből megtudhattuk, hogy milyen csodás körülmények között vásárolt egy vékony pénzü magyar turista München egyik áruházában kistia számára egy olcsó, sőt leértékelt számítógépet. Ha írással az említett cikk karikatúrájaként sikeredne, arról nem én tehetek, hanem a tények, amelyekhez igyekszem szigorúan ragaszkodni.

1985. április. Örömmel értesültem az egyik volt diákiomtól, hogy a Videoton elkészült olcsó személyi számítógépével, és TV-Computer néven hamarosan forgalomba is hozza.

1985. május. A BNV-n, a Videoton standjánál már sok számítógép-rajongó gyönyörködhetett a TV-Computer néhány példányában. A Mikroszámítógép Magazin, a Számítástechnika, sőt még a napilapok is írtak róla vásári beszámolóikkal.

1985. június. Lelkes diákjaimmal úgy búcsúztunk a számítástechnikai szempontból is sikeres tanévétől, hogy az új tanév kezdetére már bizonyára ezzel az új, sokat ígérő géppel is gyarapodni fog iskolánk géppalómánya.

1985. szeptember. Diákjaim megjöttek, de a gép sajnos nem. Helyette hozta az egyik diákom a géphez ígért könyvsorozat első két könyvét: a TV-Computer kezelési útmutatóját és BASIC programozási segédletét. A két csinos kiállítású könyv még jobban felcsigázta érdeklődésünket. Tartalmuk azt mutatta, hogy a gép sokat tud és sok mindenre használható. Örültünk, hogy a gépnek már megjelenésekor lesz használható irodalma. Most már egyre türelmetlenebbül vártunk.

1985. november. A SW '85 szoftvervásáron az egyik diákom dolgozhatott a gépen. Nagyon tetszett neki. Reméltem, hogy karácsonyra mindannyian örülhetünk a gépnek. Alátámasztott ezt a reményt híreink is. A Mikroszámítógép Magazin 1985. évi 4. számában Videoton-újdságok címen megjelent a TV-Computer fényképe és a hír, hogy a gyár az év második felében 2000 darabot szeretne belőle piacra hozni. A Számítástechnika novemberi számában pedig egy rövid híren belül olvashattuk, hogy a TV-Computerek is Tabon készülnek. Tanítványaimnak kiadtam, hogy aki a karácsonyi vásáron lát valahol ilyen gépet, azonnal jelentse, hogy megvehessük.

1985. december. Az Ötletben teljes oldalt kitöltő hirdetés, hogy a Videoton forgalomba hozta a TV-Computert. Alatta telefonszámok, ahelyett, hogy a vásárlási helyek és a gép ára lennének felsorolva. (Nálunk még nincs gyakorlatban a telefon útján való rendelés!)

1986. január. Elmúlt a karácsony, de gépünk nincs. No de se baj! Az Ötlet január 2-i számában ismét egy egész oldalas hirdetés — telefonszámokkal.

1986. január 5. A tv vasárnapi szép színes reklámjaiban kétszer is felvillant a TV-Computer hirdetése.

1986. január 8. Mivel diákjaim még mindig nem hoztak hírt arról, hogy valamelyik kirakatban látták volna a TV-Computert, a telefonra fanyalodtam. Felhívtam a hirdetésekben megadott telefonszámot. A vevőszolgálatnál barátságos férfihang fogadta kérdéseimet. Néhány dicséret szót is mondott a gépről, majd kapcsolta az illetékes csoportot. Itt megkaptam a kért információkat: az egyik művelődési házban van ilyen gép, ott megnézhetem, és az egyik nagyruházban érdeklődhetem a vétellel kapcsolatban.

Másnap kora délelőtti elmentem az áruházba. Kicsit reménykedtem, hogy ebben az előkelő, háromszintes épületben valami olyat látok, amit az előző cikk írója abban a müncheni külvárosi diszkontárúházból.

A karácsony utáni ismert kép fogadott. Alig lézengtet néhány érdeklődő, de a polcok bőséget mutattak. Örültem, hogy kevés a vásárló, mert így figyelmesebb kiszolgálásra számíthatam. Végignéztem a műszaki osztályt, de a TV-Computerre vonatkozó, általam elképzelt nagy reklámokból semmit sem láttam, még magát a gépet sem. Az egyik polcon felfedeztem a Primót: egyedül szomorokodott, tévé és rivális nélkül. Éz felbátorított, és az arra sétáló eladótól érdeklődtem a Videoton Computer után. Kijelentette, hogy ilyenről még nem is hallott, de a biztonság kedvéért megkérdezte a kollégánóját, tud-e róla. A hölgy végignézett rajtam, és azt mondta, hogy ha itt van nálam a számítógép ára, 20 ezer forint, akkor kihozza és megmutatja. Meglepetésemben alig tudtam megszólalni, de látszottat az arcomon, hogy valóban nincs a zsemben ennyi pénz, mert a hölgy diadalmasan elvonult a csatáról. Így nemcsak bemutatásból, de még megmutatásból sem lett semmi.

Szégyenkezve kullogtam ki az áruházból. Resteltem, hogy nyugdíjas létemre nem 20-30 ezresrel a zsembem indulok nyomozásra egy olyan számítógép után, amelynek csak a reklámjára már legalább 100 ezer forintot költöttem.

Optimista vagyok, és remélem, hogy mire írásom megjelenik, minden áruházban és Kérvillban lehet két-három fajta hazai gyártmányú számítógépet kapni. Ha így lesz, akkor szívesen elfelejtem mostani bosszankodásomat. De a propagandá, a vevőszolgálat és a kereskedelem tehetne valamit azért, hogy a vásárlás a vevő számára ne nyomozás, bosszúság és megaláztatás, hanem öröm legyen!

KOVÁCS MIHÁLY



INFORMÁCIÓTECHNIKAI VÁLLALAT

Tisztelt jövőbeni felhasználónk!

Kereskedelmi áruajánlatunk választékából az alábbiak forgalmazására szeretnénk szíves figyelmét felhívni!

- ROBOTRON A 6402-es típusú kisszámítógépek
- FORM-o-TRONIC leporellóvágó és -válogató berendezések
- MEOPTA mikrofiche olvasó- és mikrofilmolvasó berendezések
- PENTAKTA mikrofiche olvasó- és visszanyagító készülékek és komplett laborok
- Hazai gyártású iratmegsemmisítő gépek
- TZ—80 kisszámítógéppel vezérelt munkaidő-rögzítő rendszerek

Vállaljuk továbbá egyedi igények alapján egyéb számítástechnikai berendezések beszerzését, szállítását.

Részletes tanácsadással, tájékoztatással,
KERESKEDELMI FŐOSZTÁLYUNK
készséggel áll felhasználóink rendelkezésére.
Telefon: 803-294

PROGRAMOK

ÖT GÉP

VIZSGÁLATA

Egy programmal

1985/6. számunkban cikket írtam „Melyiket vegyem?” címmel, melyben a legalacsonyabb árkategóriába tartozó 10 gépet összehasonlító táblázatokat közöltem. Ezekhez készítettés és egyúttal két változatot programot ad ez a cikk.

Különböző gépek — igazában azok fordítóprogramjai, de mert ezek gyakran azok tartózkéki, tehát mindkettő — összehasonlítására, adott programok futási sebességei közli a szakirodalom. (Mi is közöltünk ilyet lapunk 1984/6. számában, a Primo-ismeretőben.) E módszer hátránya, hogy a legtöbb gépen futtatható programokat igényel. Ilyenek készítésénél viszont a gépek előnyös, egyedi tulajdonságait nem vehetik figyelembe.

Egy másik lehetőség az összehasonlításra, ha nem a program, hanem a feladat adott. Így, ha a feladatok a leggyakrabban előforduló felhasználásokat felelik, egy olyan összehasonlítást kapunk, ami realisabb képet adhatna. A feltételes mód oka az, hogy itt az is feltétel, hogy a gépek tulajdonságait teljesen kihasználó programokat kellene összehasonlítani. Ehhez viszont „mindentudó” programozók kellene. Ilyenek természetesen nincsenek.

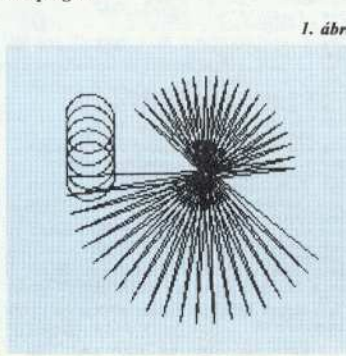
Egyes nagy példányszámú nyugati lapok a legelterjedtebb géptípusok lehetőségeit viszonylag jól ismerő állandó szakírogárdával igyekeznek hasonló színvonalú programokat készíteni ugyanarra a feladatra. Ezek támpontot jelentenek a gépkiválasztásnál.

A COMPUTE! c. szaklap 1983. októberi számában közölt egy olyan programot, ami cikket tíz gépe közül ötre, valamint két másikra (Apple II és VIC-20) ad változatot. A közölt változatok a két Atari, a C64 és a két Dragon gépen futtathatók. (A TRS 80 Color és a Dragon gépek BASIC-szinten azonosak.)

Maga a program a Spirograph nevű eszköz rajzolásmodját általánosítja. Egy olyan vonalhálózatot állít elő, amely két sugár mozgásából jön létre. Az egyik sugár — a beadott értéktől függően — egyenletesen körbe fordul vagy egyenes mentén mozog; a másik sugár középpontja az első, szabad végpontja. Beadandó a két sugár hosszának aránya, a vonalhálózatok forgásának mértéke, az egyik forgás közben zsurgordjon vagy sem, végül letörölhető az eddigi rajzok, vagy egymásra rajzolhatók

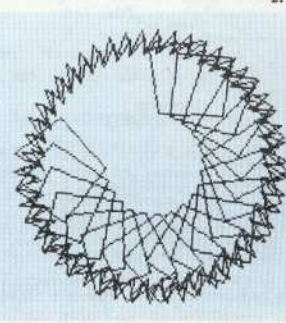
10	I=49152:CK=8	49712	DATA255,207,13,193,207,144,38
20	READR CK=CK+R:PRINT J,CK:IF	49720	DATA56,173,226,207,237,254,207
	R#256:HEN40	49728	DATA141,226,207,173,227,207,237
30	POKE I,R:I=I+1:GOTO 20	49736	DATA235,207,141,227,207,24,173
40	IFCK=167356 THENPRINT	49744	DATA250,207,103,246,207,141,250
	"ERROR IN DATA STATEMENTS"	49752	DATA207,173,251,207,109,247,207
49152	DATA 173,234,207,141,236,207,173	49760	DATA141,251,207,32,46,194,206
49160	DATA 232,207,141,238,207,173,235	49768	DATA252,207,173,252,207,201,255
49168	DATA 207,141,237,207,173,233,207	49776	DATA249,3,76,104,193,206,253
49176	DATA 141,239,207,173,236,207,56	49784	DATA207,76,104,193,96,173,251
49184	DATA 237,230,207,141,240,207,173	49792	DATA207,240,8,173,250,207,201
49192	DATA 237,207,237,231,207,141,241	49800	DATA55,144,1,56,173,248,207
49200	DATA 207,176,24,173,240,207,73	49808	DATA201,129,14,96,173,249
49208	DATA 255,141,240,207,173,241,207	49816	DATA207,240,1,96,173,250,207
49216	DATA 73,255,141,241,207,236,240	49824	DATA141,212,207,173,251,207,141
49224	DATA 207,206,3,238,241,207,173	49832	DATA213,207,173,248,207,141,214
49232	DATA 248,207,56,237,228,207,141	49840	DATA207,173,249,207,141,215,207
49240	DATA 244,207,173,239,207,237,229	49848	DATA173,215,207,74,141,217,207
49248	DATA 207,141,245,207,176,24,173	49856	DATA173,214,207,106,141,216,207
49256	DATA 244,207,73,255,141,244,207	49864	DATA173,217,207,74,141,217,207
49264	DATA 173,245,207,73,255,141,245	49872	DATA173,216,207,106,141,216,207
49272	DATA 207,235,244,207,208,3,238	49880	DATA173,217,207,74,141,217,207
49280	DATA 245,207,56,173,236,207,237	49888	DATA173,216,207,106,141,216,207
49288	DATA 230,207,141,192,207,173,237	49896	DATA173,213,207,74,141,219,207
49296	DATA 207,237,231,207,13,192,207	49904	DATA173,212,207,106,141,218,207
49304	DATA 240,15,144,24,169,1,141	49912	DATA173,219,207,74,141,219,207
49312	DATA 222,207,169,6,141,223,207	49920	DATA173,218,207,106,141,218,207
49320	DATA 76,169,132,169,6,141,222	49928	DATA173,219,207,74,141,219,207
49328	DATA 207,141,223,207,76,169,132	49936	DATA173,219,207,106,141,219,207
49336	DATA 169,255,141,222,207,141,223	49944	DATA173,214,207,41,7,141,220
49344	DATA 207,56,173,238,207,237,228	49952	DATA207,173,216,207,10,46,217
49352	DATA 207,141,193,207,173,239,207	49960	DATA207,10,46,217,207,10,141
49360	DATA 237,229,207,13,193,207,240	49968	DATA210,207,46,217,207,173,217
49368	DATA 15,144,24,169,1,141,246	49976	DATA207,141,211,207,173,210,207
49376	DATA 207,169,6,141,247,207,56	49984	DATA110,46,217,207,10,46,217
49384	DATA224,192,169,6,141,246,207	49992	DATA173,211,207,106,141,216,207
49392	DATA141,247,207,76,224,192,169	49996	DATA217,207,173,216,207,10,46
49400	DATA255,141,246,207,141,247,207	50000	DATA173,217,207,10,46,217,207,10
49408	DATA173,230,207,141,248,207,173	50004	DATA141,218,207,24,173,216,207
49416	DATA228,207,141,250,207,173,231	50008	DATA173,217,207,106,141,216,207
49424	DATA207,141,249,207,173,229,207	50012	DATA173,217,207,10,46,217,207,10
49432	DATA141,251,207,141,211,207,173	50016	DATA141,218,207,24,173,216,207
49440	DATA237,240,207,141,211,207,173	50020	DATA173,217,207,106,141,216,207
49448	DATA245,207,237,241,207,13,211	50024	DATA173,217,207,10,46,217,207,10
49456	DATA207,176,15,173,240,207,141	50028	DATA173,217,207,106,141,216,207
49464	DATA252,207,173,241,207,141,253	50032	DATA173,217,207,10,46,217,207,10
49472	DATA207,76,40,193,173,244,207	50036	DATA173,217,207,106,141,216,207
49480	DATA141,252,207,173,245,207,141	50040	DATA173,217,207,10,46,217,207,10
49488	DATA253,207,173,252,207,141,254	50044	DATA173,217,207,106,141,216,207
49496	DATA207,173,253,207,141,255,207	50048	DATA173,217,207,10,46,217,207,10
49504	DATA56,173,252,207,237,244,207	50052	DATA173,217,207,106,141,216,207
49512	DATA141,211,207,173,253,207,237	50056	DATA173,217,207,10,46,217,207,10
49520	DATA245,207,13,211,207,208,17	50060	DATA173,217,207,106,141,216,207
49528	DATA173,255,207,74,141,161,207	50064	DATA173,217,207,10,46,217,207,10
49536	DATA173,254,207,106,141,160,207	50068	DATA173,217,207,106,141,216,207
49544	DATA76,184,193,173,255,207,74	50072	DATA173,217,207,10,46,217,207,10
49552	DATA141,227,207,173,254,207,106	50076	DATA173,217,207,106,141,216,207
49560	DATA141,226,207,141,225,207,13	50080	DATA173,217,207,10,46,217,207,10
49568	DATA253,207,240,207,173,253,207	50084	DATA173,217,207,106,141,216,207
49576	DATA8,22,24,173,168,207,109	50088	DATA173,217,207,10,46,217,207,10
49584	DATA240,207,141,160,207,173,161	50092	DATA173,217,207,106,141,216,207
49592	DATA207,109,241,207,141,161,207	50096	DATA173,217,207,10,46,217,207,10
49600	DATA76,142,193,76,45,194,56	50100	DATA173,217,207,106,141,216,207
49608	DATA173,160,207,237,254,207,141	50104	DATA173,217,207,10,46,217,207,10
49616	DATA211,207,173,161,207,237,255	50108	DATA173,217,207,106,141,216,207
49624	DATA207,13,211,207,144,30,56	50112	DATA173,217,207,10,46,217,207,10
49632	DATA173,160,207,237,254,207,141	50116	DATA173,217,207,106,141,216,207
49640	DATA160,207,173,161,207,237,255	50120	DATA173,217,207,10,46,217,207,10
49648	DATA207,141,161,207,24,173,248	50124	DATA173,217,207,106,141,216,207
49656	DATA207,109,222,207,141,248,207	50128	DATA173,217,207,10,46,217,207,10
49664	DATA173,249,207,109,223,207,141	50132	DATA173,217,207,106,141,216,207
49672	DATA249,207,24,173,226,207,199	50136	DATA173,217,207,10,46,217,207,10
49680	DATA249,207,141,226,207,173,227	50140	DATA173,217,207,106,141,216,207
49688	DATA207,109,245,207,141,227,207	50144	DATA173,217,207,10,46,217,207,10
49696	DATA56,173,226,207,237,254,207	50148	DATA173,217,207,106,141,216,207
49704	DATA141,193,207,173,227,207,237	50152	DATA173,217,207,10,46,217,207,10

1/a. program



1. ábra

2. ábra



```

10 POKE53281,1:PRINT"XXXXXXXXXXXXXXXXXXXX";
20 PRINT"J"
30 PRINT"
40 PRINT"      B          B"
50 PRINT"      | SPIRALIZER |"
60 PRINT"      |              |"
70 PRINT"      |              |"
80 PRINT"XXXXXXXXXXXXXXXXXXXXPRESS SPACE"; POKE198,0:FORT=1T0300:NEXT
100 WAIT197.64:64:POKE53272,21:POKE53265,27:PRINT"J";
110 Z=5:PRINT"SPEED<-50 TO 50? "; "      ";GOSUB500:K=Z
115 IFZ<-50OR Z>50THEN110
125 K=K-1
127 Z=35:PRINT"RADIUS (1 TO 60)? "; "      ";GOSUB500:R=Z
130 R=R+13:S=1
135 Z=1:PRINT"SPIN(1 TO 18)? "; "      ";GOSUB500
137 IFZ<1ORZ>18THEN135
140 R=1/2:IFZ>1ANDZ<9THEN200
143 SM=1:M=2:PRINT"MOVEMENT OR DECREMENT (M/D)? "; "      ";
145 GETX$:IFX$="" THEN145
148 IFX$=CHR$(13) THEN M=0:SM=0:PRINT:PRINT"NONE";:GOTO200
150 IFX$="M" THEN SM=0:GOTO190
160 IFX$="D" THEN M=0:GOTO190
170 GOTO145
190 PRINTX$
200 PRINT:PRINT"CLEAR (Y/N)?
201 GETT$:IFT$="" THEN201
203 IFT$<"N" THENPRINT"YES":PRINT"J";:SYS50012
205 IFT$="N" THENPRINT"NO":SYS50120
210 W=1:Z=139:IFM=2THEN Z=80:IFR=1THEN W=5:M=1:Z=122
215 IFSM=R=0THEN W=5
220 IFR<1THEN K=K+R
230 C=0.001:IFR<1/9 THEN M=M/2:C=C/2
240 J=R:I=79-R
250 X0=Z+20-Y0=30:REM:SYS49152
260 REM MAIN LOOP
270 FORT=0T06.2831/A#W STEP 0.06283
280 IFSM THEN J=R#S:I=79#S-J:S=S-C
290 X1=Z+T#M-SIN(T)*J+SIN(T#K)*I:X1=X1+20
300 Y1=79-COS(T)*J-COS(T#K)*I:Y1=Y1+30
308 GOSUB1000:X0=X1:Y0=Y1
310 NEXT
500 L0=1:L1=1:B$=""
510 PRINTX$;
520 GETX$:IF R$="" THEN520
530 IF R$="" AND L0=0THEN PRINTX$;B$=R$:L0=1:L1=2:GOTO510
540 IFR$=CHR$(13) AND L0=0THENZ=VAL(B$):PRINT:RETURN
550 IF R$=CHR$(13) THEN PRINT:RETURN
560 IF R$=R$ AND L0>0THEN PRINTX$;B$=LEFT$(B$,LEN(B$)-1):L0=L0-1:GOTO510
570 IF R$=R$ AND L0=0THEN PRINTX$;B$="":L0=0:GOTO510
580 IFL0<1THEN510
590 IFR$<"0"ORR$>"9" THEN510
600 PRINTX$;B$=B$+R$:L0=L0+1:GOTO510
900 POKE53272,21:POKE53265,27:PRINT"J":END
1000 POKE53272,PEEK(53272):OR8:POKE53265,PEEK(53265):OR32
1005 X0=INT(X0):IFX0<255THEN POKE53220,X0:POKE53221,0
1010 IF X0>255THEN POKE53220,(X0-255):POKE53221,1
1030 POKE53222,Y0:POKE53223,0
1040 X1=INT(X1):IFX1<255THEN POKE53224,X1:POKE53225,0
1050 IF X1>255THEN POKE53224,(X1-255):POKE53225,1
1060 POKE53226,Y1:POKE53227,0
1200 SYS49152:RETURN

```

1/b. program

```

10 CLS:0
20 PRINT"SPIRALIZER";
25 FORI=1 TO 1000:NEXT
30 CLS:3
35 K=5
40 INPUT "SPEED (-50 TO 50)";:K=K
41 K=VAL(K)
42 IF K=0 THEN K=5
45 IF K<-50OR K>50THEN48
47 K=K-1
48 R=35
50 INPUT "RADIUS (1-60)";:R=
51 R=VAL(R)
52 IF R=0 THENR=35
55 IF R<1 OR R>60 THEN50
57 R=1/3:SM=1
60 INPUT "SPIN 1-18";:Z=
61 Z=VAL(Z)
63 IF Z=0 THENZ=1
65 IF Z<1OR Z>18THEN 60
70 M=1/2:IF Z>1AND Z<9THEN200
80 SM=1:M=2
85 INPUT "MOVEMENT OR DECREMENT";:X$
90 IF LEFT$(X$,1)<"M" ANDLEFT$(X$,1)<"D" THENM=0:SM=0:PRINT:PRINT"NONE";:GOTO200
95 IF LEFT$(X$,1)="M" THENSM=0:GOTO 190
100 IFLLEFT$(X$,1)="D" THENM=0:GOTO190

```

```

190 PRINT X0
200 PRINT:INPUT"CLEAR (Y/N)";:T$
205 IF LEFT$(T$,1)<"Y" THENPRINT"YES";:E=
210:REMPUT CLEAR OF HIRES
210 IF LEFT$(T$,1)="Y" THENPRINT"NO"
220 W=1:Z=139:IF M=2 THEN Z=80:IF A=1 TH
EM=M/2:IF Z=122
230 IF SM=0 THENM=5
235 IF A<1 THENM=K+A
238 D=0.001:IFA(1/9) THENM=M/2:C=C/2
240 J=R:I=79-R
250 X0=Z+Y0=20
260 REM MAIN LOOP
270 FOR T=0 TO 6.2831/A#W STEP 0.06283
280 IF SM THENJ=R#S:I=79#S-J:S=S-C
290 X1=Z+T#M-SIN(T)*J+SIN(T#K)*I
300 Y1=79-COS(T)*J-COS(T#K)*I:Y1=Y1+20
310 GOSUB 1000:X0=X1:Y0=Y1
320 NEXT
330 A=INTKEY$:IFA$="" THEN330
340 GOTO300
1000 SCREEN 1,1
1010 IF LEFT$(T$,1)="Y" ANDM=0 THENICLS:
E=1
1020 LINE (X0,Y0)-(X1,Y1):PSET
1030 RETURN

```

2. program

AND	C8	ON	88
AUDIO	A3	OPEN	9B
CIRCLE	B1	OR	C9
CLEAR	96	PAINT	B2
CLOAD	99	PCLEAR	AF
CLOSE	9C	PCLS	AB
CLS	A0	PCOPY	B6
COLOR	B0	PLAY	B8
CONT	9A	PMODE	B7
CSAVE	9A	POKE	93
DATA	86	PRESET	AD
DEF	98	PRINT	87
DEL	A6	PSET	AC
DIM	8C	PUT	B4
DLOAD	B9	READ	8D
DRAW	B5	REM	82
EDIT	A7	RENUM	BA
ELSE	84	RESET	9F
END	8A	RESTORE	90
EXEC	A4	RETURN	91
FN	BE	RUN	8F
FOR	80	SCREEN	AE
GET	B3	SET	9E
GO	81	SKIPF	A5
IF	85	SOUND	A2
INPUT	89	STEP	C1
LET	8E	STOP	92
LINE	AA	SUB	BD
LIST	95	TABC	BB
LLIST	9D	THEN	BF
MOTOR	A1	TO	BC
NEW	97	TROFF	A9
NEXT	8B	TRON	A8
NOT	C0	USING	CD
OFF	C2		

1. táblázat. Parancsok, utasítások BASIC kódjai (69)

+	C3	↑	<7
-	C4	=	CB
.	83 (REM)	>	CA
*	<5	<	CC
/	<6	!	<E

2. táblázat. Jelek (10)

az egyes hálózatok. A változók értékének változtatásával rendkívül érdekes ábrák adódnak.

A VIC-20 változat csak Super Expander használatával valósítható meg! A C64 változat – a grafikus utasítások hiányában – egy grafikus utasításokat előállító programot is tartalmaz. Szívesen közlünk C16, Spectrum, Primo, HT változatot is! Ilyeneket várunk!

Számszerűen összehasonlítva a változatokat, az Apple II változat 115, a VIC-20 változat 142, az Atari-változat 122, a Dragon-változat 78, a C64 változat 143 és a grafikus program 154 utasításból áll. Látható egyrészt az, hogy a Dragon-változat nagyon rövid, másrészt az, hogy a C64 még a grafikus programot figyelmen kívül hagyva is igen hosszú!

Az összehasonlítás és a géphasznalet megkönnyítésére közlöm a C64 (1/a. és 1/b. program)és a Dragon 64 (2. program) változat listáit és kulcsszó-kódtáblázatát. A Dragon rajztudásának szemléltetésére pedig bemutatok néhány, a rajzolóprogram készítette ábrát (1-2. ábra).

Az első bájtt mindig FF

ABS	82	MID\$	98
ASC	90	PEEK	8C
ATN	8B	POINT	99
CHR\$	91	PPOINT	9F
COS	89	POS	83
EOF	92	RIGHT\$	97
EXP	87	RND	84
FIX	94	SGN	80
HEX\$	95	SIN	88
INKEY\$	9A	SQR	85
INSTR	9D	STR\$	8E
INT	81	STRING\$	A0
JOYSTK	93	TAN	8A
LEFT\$	96	TIMER	9E
LEN	8D	USR	A1
LOG	86	VAL	8F
MEM	9B	VARPTR	9C

3. táblázat. Dragon BASIC függvénykódok (34)

AND	AF	NOT	A8
CLOSE	A0	ON	91
CLR	9C	OPEN	9F
CMD	9D	OR	B0
CONT	9A	POKE	97
DATA	83	PRINT	99
DEF	96	PRINT #	98
DIM	86	READ	87
END	80	REM	8F
FN	A5	RESTORE	8C
FOR	81	RETURN	8E
GET	A1	RUN	8A
GOTO	89	SAVE	94
GOSUB	8D	SPC/	A6
IF	8B	STEP	A9
INPUT	85	STOP	90
INPUT #	84	SYS	9E
LET	88	TAB/	A3
LIST	93	THEN	A7
LOAD	9B	TO	A4
NEW	A2	VERIFY	95
NEXT	82	WAIT	92

4. táblázat. C64 parancsok, utasítások BASIC kódjai (44)

+	AA	↑	AE
-	AB	=	B2
*	AC	>	B1
/	AD	<	B3

5. táblázat. C64 jelek (8)

Nincs két bájtt			
ABS	B6	PEEK	C2
ASC	C6	POS	B9
ATN	C1	RIGHT\$	C9
CHR\$	C7	RND	BB
COS	BE	SGN	B4
EXP	BD	SIN	BF
FRE	B8	SQR	BA
INT	B5	STR\$	C4
LEFT\$	C8	TAN	C0
LEN	C3	USR	B7
LOG	BC	VAL	C5
MID\$	CA		

6. táblázat. C64 függvénykódok (23)

PTA-4000

LH 5801

utasításkészlet

Az utasítások megadásának módja azonos azzal, amit más cégek alkalmaznak, tehát

- utasításnév rövidítéssel (3 betű),
- a művelet tárgya rövidítve,
- a művelet.

A használt rövidítések

- A – akkumulátor
- T, TM, X, Y, U – regiszterek
- L, H – alsó, illetve felső bájtt
- B – mindkét bájtt
- P – programszámláló
- S – veremmutató
- Rreg – regiszter általában
- (Rreg) – regisztertartalom által megadott memóriacím
- a, b – a memória alsó, illetve felső bájttartalom
- * – a második lap (MEI)
- j, i – közelebbbról meg nem határozott alsó, illetve felső bájtt
- C – carry bit
- IE – megszakítást engedélyező bit
- Z – zéró bit
- V – túlszordulás bit
- H – fél carry bit
- ⇌ – adatmozgatási irány
- ^ – ÉS
- ∨ – VAGY
- (+) – kizáró VAGY

1. Matematikai műveletek

ADC: Egy regiszter vagy memóriahely tartalmát és a C értékét hozzáadja A tartalomhoz. Az eredmény az A-ba kerül.

- Fajttái: ADC RL
ADC RB
ADC (Rreg)
ADC *(Rreg)
ADC (ab)
ADC *(ab)

ADI: A művelethez tartalomhoz egy számot ad (ha ez az A, akkor C értékét is). Az eredmény ugyanott lesz.

- Fajttái: ADI A,i
ADI (Rreg),i
ADI *(Rreg),i
ADI (ab),i
ADI *(ab),i

DCA: Az ADC-hez hasonló csak decimális.

- Fajttái: DCA (Rreg)
DCA *(Rreg)

ADR: A regiszterek és A tartalmát adja össze. Az alsó bájtt az A-ba, a C-vel növelt felső a regiszter felső bájttjába kerül.

- Fajttái: ADR Rreg
SBC: Az ADC-hez hasonló, csak kivó-

nás. A kisebbítendő az A, és C helyett annak inverze (C) szerepel.

- SBI: Az ADI-nek megfelelő kivonás.
DCS: A DCA-nak megfelelő kivonás.
INC: Eggyel növeli A,R tartalmát. 16 bites regiszternél a C nem változik.

- Fajttái: INC A
INC RL
INC RH
INC Rreg
DEC: Az INC-hez hasonló, csak csökkent.

2. Logikai műveletek

AND: Az A és R tartalmát ÉS kapcsolatba hozza bitenként. Az eredmény az A-ba kerül.

- Fajttái: AND (Rreg)
AND *(Rreg)
AND (ab)
AND *(ab)
ANI: Az AND-hez hasonló, az ADC és ADI közti eltéréssel.
ORA: Az AND-hez hasonló, csak VAGY.
ORI: Az ANI-hez hasonló, csak VAGY.
EOR: Az ORA-hoz hasonló, csak kizáró VAGY.
EAI: Az ORI-hez hasonló, csak kizáró VAGY.

3. Összehasonlítás és bitvizsgálat

CPA: Az A és egy másik hely tartalmát hasonlítja össze. Az eredmény a C,V,H,Z értéket változtatja.

- Fajttái: CPA RL
CPA RH
CPA (Rreg)
CPA *(Rreg)
CPA (ab)
CPA *(ab)

CPI: Hasonló a CPA-hoz.
Fajttái: CPI RL,i
CPI RH,i
CPI A,i

BIT: Az A és egy másik hely tartalmát ÉS kapcsolatba hozza. Az eredmény a Z bitet változtatja.

- Fajttái: BIT (Rreg)
BIT *(Rreg)
BIT (ab)
BIT *(ab)

BII: Hasonló a BIT-hez, azzal az eltéréssel, ami az ADC és ADI közt van.

4. Adatmozgatás

LDA: Az adatot az A-ba viszi.

- Fajttái: LDA RL
LDA RH
LDA (Rreg)
LDA *(Rreg)
LDA (ab)
LDA *(ab)

LDE: Hasonló az LDA-hoz, csak a Rreg tartalmát utólag eggyel csökkenti.

Fajtái: LDE Rreg

LIN: Hasonló az LDE-hez, csak növel.

LDI: Hasonló az LDA-hoz, az ADC és ADI között eltéréssel, de itt S is szerepelhet.

LDX: Rreg, S, P tartalmát X-be tölti.

STA: Hasonló LDA-hoz, csak a mozgás iránya fordított.

SDE: Az LDE párja.

SIN: Az LIN párja.

STX: Az LDX párja.

PSH: A, Rreg tartalmát S-be tölti, majd bájtónként eggyel csökkenti S címét.

POP: A PSH ellentettje.

ATT: A tartalmát T-be tölti.

TTA: ATT ellentettje.

5. Blokkmozgatás

TIN: X tartalmát Y-ba tölti, majd a regiszterek mutatóit eggyel növeli.

CIN: Az A,X tartalmát hasonlítja össze, majd X címét eggyel növeli.

6. Forgatás, léptetés

ROL: A bitjei feljebb, legfelső bitje C-be kerül, míg C eredeti értéke A legelső bitje lesz.

ROR: A forgatás iránya ellentett.

SHL: Hasonló ROL-hoz, csak A legelső bitje 0 lesz.

SHR: A forgatás iránya ellentett.

DRL: X, ill. *X és A közötti átvitel, a regiszter alsó fél bájta a felsőbe, a teljes bájtt átkerül A-ba, A eredeti felső fél bájta a regiszter alsó felébe kerül.

DRR: Az előző ellentettje.

AEX: Az A két fél bájta cserél.

7. Vezérlés

SEC: C=1 értéket állít be.

REC: C=0 értéket állít be.

CDV: A belső órát nullázza.

ATP: Az A tartalmát az adatbuszra viszi.

SPU: PU=1 értéket állít be.

RPV: PV=0 értéket állít be.

SPV: PV=1 értéket állít be.

RPV: PV=0 értéket állít be.

SDP: DISP=1 értéket állít be.

RDP: DISP=0 értéket állít be.

ITA: Az adatbusz tartalmát az A-ba tölti.

SIE: IE=1 értéket állít be.

RIE: IE=0 értéket állít be.

AM0: A tartalmát TM-be viszi és TM8=0 lesz.

AM1: Hasonló az előzőhöz, csak TM8=1

NOP: Nem történik semmi.

HLT: Leállítja a mikroprocesszort és megszakítást vár.

OFF: A billenőkört visszaállítja.

8. Ugrások

JPM: Az ij bájtpár által megadott címre állítja P-t.

BCH: i lépéssel változtatja P-t.

Fajtái: BCH +i

BCH -i

BCS: Hasonló BCH-hoz, de csak C=1 esetén ugrik

BCR: Hasonló BCH-hoz, de csak C=0 esetén ugrik

BHS: Hasonló BCH-hoz, de csak H=1 esetén ugrik

BHR: Hasonló BCH-hoz, de csak H=0 esetén ugrik

BZS: Hasonló BCH-hoz, de csak Z=1 esetén ugrik

BZR: Hasonló BCH-hoz, de csak Z=0 esetén ugrik

BVS: Hasonló BCH-hoz, de csak V=1 esetén ugrik

BVR: Hasonló BCH-hoz, de csak V=0 esetén ugrik

LOP: UL tartalmát eggyel csökkenti, és ha $UL \geq 0$, akkor P értékét i-vel csökkenti, különben nem.

SJP: Hasonló JMP-hoz, csak P értékét S-be viszi.

VEJ: Hasonló SJP-hez, csak az ugrás az FF00H+i cím és a következő által megadott címre történik. Visszatérés a VEJ utáni címre.

VMJ: Hasonló VEJ-hez, csak a visszatérés a VMJ utáni címre.

VCS: Hasonló VMJ-hez, de csak C=1 esetén ugrik.

VCR: Hasonló VCS-hez, de C=0

VHS: Hasonló VCS-hez, de H=1

VHR: Hasonló VCS-hez, de H=0

VZS: Hasonló VCS-hez, de Z=1

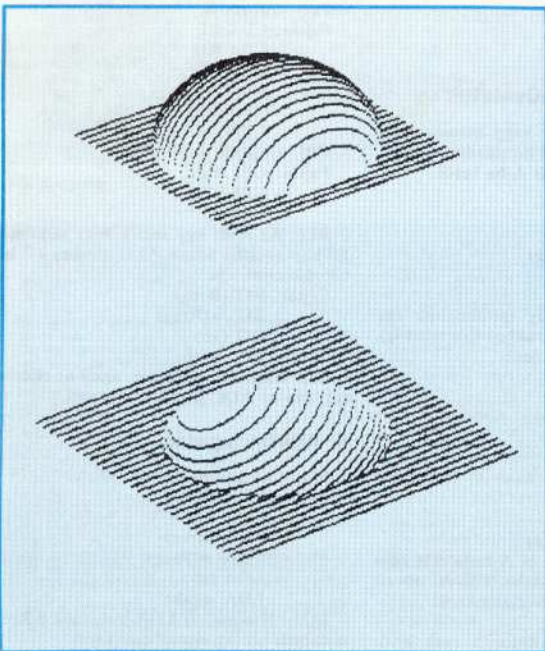
VZR: Hasonló VCS-hez, de Z=0

VVS: Hasonló VCS-hez, de V=1

RTN: Visszatérés a szubrutinból.

RTI: Visszatérés a megszakítás után.

DR. SIMONYI ENDRE



HELYREIGAZÍTÁS

Lapunk 1985/6. számában az UNIN rutin programlistájában sajnálatos módon hibásan jelentek meg egyes sorok. Ezek helyesen a következők:

```
105 T1 = PEEK(51):T2 = PEEK(52)
:T6 = T1 + 256*T2 - 41:AS = " " +
AS:T4 = INT(T6/256):T3 = T6 - T4*256
140 B = LEN(AS):IFB > 15:AND B < 6:AND B >
0:THEN FORB = B + 1:TO 6:
JS = " " + AS:GOSUB 540:NEXT:GOTO 160
210 Z = (55296 = L*40 + C):FORI = Z:TO Z + P
:POKEI,CO:NEXT:Y = 1024 +
L*40 + C:X = Y + P:IFB <> 7:ORB = 0:THEN 230
265 POKESI,S3:S3 = 0:S4 = 80:
IFB8 = 7:AND TS = " " AND A = 1:THEN GOSUB 535:
JS = CHR$(14):GOSUB 540:A = 0:GOTO 195
340 A = A + 1:GOSUB 535:
JS = LEFT$(AS,A-2) + TS + RIGHT$(AS,
(AV - A + ABS(AV - A))/2):
GOSUB 540:IFA > AV:THEN AV = A
495 IFB0 <> 7:AND (B1 = 0:AND LEFT$(AS,1) = " ")
OR (B1 <> 0:AND RIGHT$(
AS,1) = " ") THEN A = 1:AV = B + 1:GOTO 240
500 IFB6 = 0:THEN FORI = X:TO YSTEP - 1:
POKEI,ASC(MID$(AS,1 - X + B,1))
AND 63:OR I28:NEXT:GOTO 195
540 LS = LS + " ":POKE820,PEEK(71):
POKE821,PEEK(72):
POKE(PEEK(820) + PEEK(821)*256,41) - LEN(JS)
```


Módosított hűtőlemez ZX81-hez

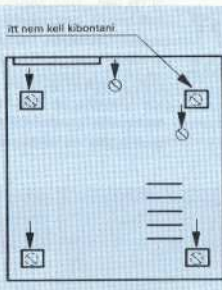
Több Sinclair mikrogép-tulajdonos tapasztalatait összegezve megállapítható, hogy ezeknél a gépeknél a leggyakrabban előforduló hirtelen halál nem az idült tápegység-elégtelenség. Ennek tünetei a következők. A készülék bekapcsoláskor kifogástalanul üzemel, majd 3-5 órai folyamatos működés után elkezd tévedni. Ez a legkülönbözőbb formákban nyilvánulhat meg: összetöredezik a képernyőtartalom, esetleg az interpreter „száll el”, vagy spontán átvált a gép FAST üzemmódba. Szerencsétlen esetben meghibásodik a μ A 7805-ös stabilizátor IC,

és a tápláló 9 V-os egyenfeszültség „mészárlást végez”.

A legradikálisabb megoldást egy külső 5 V-os tápegység jelentené, amely gyökerében megszüntetne minden ilyen problémát. Ehhez azonban nagyon meg kellene bolygatni a gépet, így a legtöbb tulajdonos várhatóan elzárkózna az ilyen „műtét” elől.

Ezért egy másik, már jól bevált megoldást javaslok: cseréljük ki nagyobbra a beépített stabilizátor hűtőlemezét. Ezáltal egészen elfogadható tüneti kezelést kapunk.

A ZX81-be eredetileg beépített hűtőlemez 2 mm vastag



2. ábra

alumíniumlemezről készült, minden felületkikészítés nélkül. Felülete kb. 18 cm². Ennek a zárt dobozba csukott kisméretű fényes lemeznek kellene kb. 2 W teljesítményt elvezetnie, még akkor is, ha a RAM kiegészítéseket nem számoljuk. Már a számadatokból is látható, hogy egy ilyen hűtőlemez valószínűleg nem tudja teljes egészében ellátni a feladatát.

A javasolt hűtőlemez felülete kb. kétszerese az eredetinek, és célszerű felületkezeléssel (eloxálás vagy matt fekete festés) is el van látva. Használatával az említett hibajelenségek nem fordultak elő. Az eredeti és a módosított hűtőlemez rajza az 1. ábrán látható.

A módosítás menete a következő.

1. Elkészítjük a módosított hűtőlemezt. Felületét durva csiszolópapírral érdesítjük, de a stabilizátorra felfekvő felületét épen hagyjuk. Az érdesített felületet eloxáljuk fekete-re, vagy egyszerűen matt fekete-re festjük. A festékréteg minél vékonyabb legyen!

2. A ZX81-et felfordítjuk, és a 2. ábrán jelölt 3 gumitalpat eltávolítjuk. Ezek a talpak kétoldalas, ún. emblémaragasztó szalaggal vannak rögzítve. Vi-

gázunk, ne gyűrődjön a ragasztószalag, mert ezzel fogjuk visszarágaztatni a talpakat.

3. A 2. ábrán jelölt 5 csavart csavarjuk ki. Az alplemez eltávolítása után láthatóvá válik a ZX81 lelke és a NYÁK-ról lelógó hűtőlemez. Aki kíváncsi, kapcsolja be így a gépet, és néhány perces üzem után, megtapintva a hűtőlemezt, meggyőződhet az elégtelen hűtésről.

4. A NYÁK-lemezt rögzítő két csavart hajtjuk ki, és emeljük ki a NYÁK-lemezt. Vigyázzunk, a billentyűzethez vezető hajlékony fóliát ne sértsük meg!

5. Csavarjuk ki a stabilizátor és a hűtőlemez összeszorító M 3,5-ös csavart, és emeljük ki a hűtőlemezt.

6. Az új hűtőlemez stabilizátor IC-hez kerülő részét vékonyan kenjük be szilikonzsírral.

7. Illesszük a helyére az új hűtőlemezt, és ellenőrizzük, nem kerül-e valamelyik része veszélyes közelségbe valamelyik NYÁK-ponttal.

8. Helyezzük be a stabilizátort és a hűtőlemezt összefogó M 3,5-ös csavart, és minél jobban húzzuk meg.

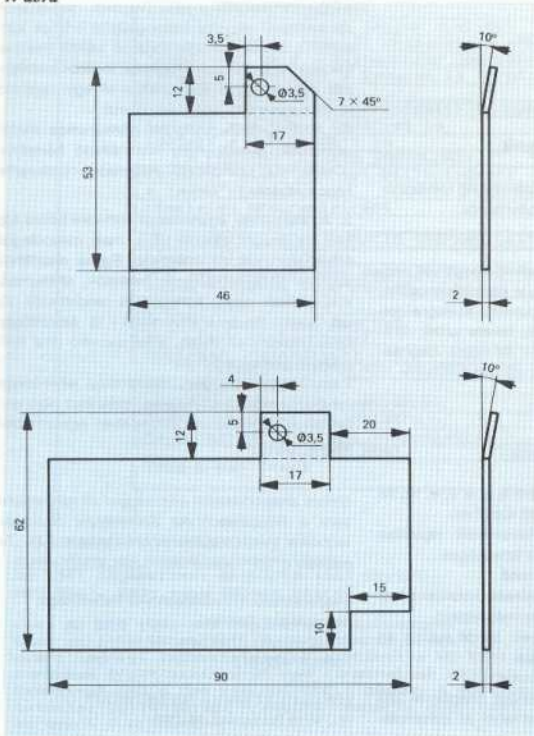
9. Ellenőrizzük, nem csinálnak-e a hűtőlemezrel zárlatot, illetve nem szakítottuk-e le a billentyűzet csatlakozását.

10. Ha mindent rendben végeztünk, szereljük össze a gépet. Vigyázat! A csavarokat nem szabad nagyon meghúzni, mert a műanyag házból kiszakad a menet.

Már csak a bekapcsolás van hátra. Ha csak egy kicsit is odafigyeltünk, akkor gépünk ezután akár állandó üzemben is megbízhatóan fog működni. Az átalakításához mindenkinek sok sikert kívánok.

NAGY GÁBOR

1. ábra



Személyszámítógép-javítás, karbantartás közületeknek, magánszemélyeknek. Egyedi megrendelés alapján kiegészítő berendezések gyártása. Pl. Sinclair fényceruza, Joystick interface, oktatási intézménynek kabinet kialakítása.

Pásztor Ferenc személyszámítógép-javító és -karbantartó kisiparos. Szolnok, Mátyás király u. 2. V/3.

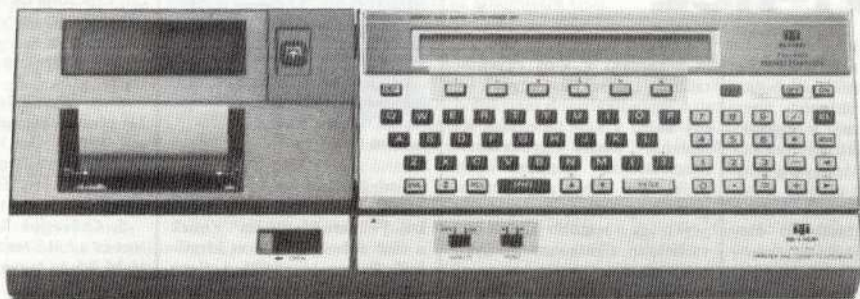
Zsebkalkulátor? Személyi számítógép? Mindkettő!

Könyv, kisméretű, zsebben hordozható, képes, magas szintű nyelven programozható személyi számítógép beépített megjelenítővel! Mivel hálózattól függetlenül és képernyő nem kell hozzá, használata nincs szobához kötve: terepen, üzemben, bányában, szántóföldön, bárhol azonnal, a helyszínen elvégezheti számításait. Miután programozható, felhasználási lehetőségei korlátlanok: az ipar, a mezőgazdaság, a tudomány, az ügyvitel bármely területén könnyebbé, gyorsabbá és hatékonyabbá teszi munkáját.

A gép kezelése nem igényel semmiféle előképzettséget. A géphez mellékelt kezelési és programozási kézikönyv — mely önállóan is megvásárolható — alapján rövid idő alatt, önállóan elsajátíthatja a gép kezeléséhez szükséges tudnivalókat.

A Híradástechnika Szövetkezet a japán SHARP cég engedélye alapján gyártja.

Kategóriájában a világ jelenlegi élvonalába tartozik.



PTA 4000 + 16 HORDOZHATÓ SZÁMÍTÓGÉP

Főbb jellemzői:

- A készülék memóriája kikapcsoláskor nem törlődik, a beadott adatokat és programokat megőrzi!! Ezáltal igen sok adat-, programbeolvasási műveletet és időt takaríthatunk meg, mert a teljes memória tartalma bekapcsoláskor rendelkezésre áll.
- Külön előny, hogy a gyakorlatlan felhasználónak nem kell az ismételt program-, illetve adatbetöltésekkel foglalkoznia.
- Memóriakapacitás: (19 968 bájtt) 20 kb-át RAM 16 kb-át ROM
- 7 x 156 pontból álló, grafikus megjelenítésre is alkalmas folyadékkristályos kijelzője tetszőleges grafikát és 96 féle kis- és nagybetűt, számot és írásjelet tud megjeleníteni.
- Egyszerre 26 karakter látható. Egy sorba maximum 80 karakter írható.
- 10 számjegyes számítás pontosság. Számábrázolás: 10 számjegy mantissza, 2 jegy kitevő.
- Jól áttekinthető, könnyen kezelhető billentyűzete 65 többfunkciós gombot tartalmaz.
- Beépített, programozható hanggenerátorával 230 Hz és 7 kHz közötti hangok szólaltathatók meg.
- Kikapcsolt állapotban is működő, naptári időt is nyilvántartó óra.
- Háttértárolóként kazettás magnetofon alkalmazható, melyből egyidejűleg kettő csatlakoztatható a számítógéphez.
- A gép teljes buszrendszere egy 60 pólusú csatlakozón a felhasználó rendelkezésére áll, így bármilyen saját készítésű hardvercsatlakozás létrehozható.
- Mérete: 195 x 86 x 25,5 mm
- Tömege: 410 gramm (elemekkel)
- Tápellátása: 6 V egyenfeszültség (4 db ceruzaelem).
- Teljesítményfelvétele: 0,13 W (6 V DC).



Kiegészítő egységek,

melyekkel a PTA 4000 + 16 számítógép felhasználási lehetőségei jelentősen bővíthetők:

1. KA 160 grafikus rajzoló

A PTA 4000 + 16 számítógéphez illeszthető, kisméretű nyomtató. Nem igényel különleges papírt, famentes, 57 mm széles összeadó szalagra dolgozik 4 db, piros, zöld, kék, fekete színű, újratölthető golyóstollpatronnal ír és rajzol. Segítségével papírra rögzíthetők: programlisták, adatok, számítási eredmények, különböző grafikák, diagramok, függvénygörbék.

A karakterek száma soronként 4, 5, 6, 7, 9, 12, 18 vagy 36, a választott mérettől függően. A kilencféle betűméretű karakterek rajzolása négyféleképpen elforgatva is lehetséges.

4 színű grafikus ábra készíthető. Nyomatási sebesség: a legkisebb karakterméret esetén maximum 11 karakter/másodperc. Nyomatási irány 4-féle lehet: balról jobbra és vissza, felülről lefelé és vissza. Beépített akkumulátorral üzemel. A tartozékként szállított hálózati adapterrel 220 V-ról működtethető, és ezzel lehet az akkumulátorokat is feltölteni.

2. Az IFSP típusú illesztőegység segítségével lehetséges a nemzetközileg szabványosított RS232C felületre való csatlakozás. Ezáltal a PTA 4000 + 16 számítógéppel végzett számítások eredményei, vagy az általa gyűjtött adatok közvetlenül, közbenso adathordozó nélkül beadhatók egy nagyobb számítógépbe további feldolgozás céljára. Így a PTA 4000 + 16 egy nagyobb rendszer termináljaként kezelhető.

Az IFSP illesztőegységnek párhuzamos kiviteli csatornája is van, mely használható bármilyen „Centronics” rendszerű nyomtató rendszerbe csatlakozására.

3. Az SB típusú, úgynevezett software board egy 14 x 10 mezőre osztott tábla, mely másodlagos billentyűzetként alkalmazható. Egyes mezőkhöz szoftver úton tetszőleges funkciók, alfanumerikus értékek, string konstansok rendelhetők. Ez azt jelenti, hogy a PTA 1400 + 16 számítógép billentyűzete 140 darab, a felhasználó által definiálható gombbal bővíti!

Ezáltal egyrészt a gép felhasználási lehetőségei rendkívül módon kibővülnek, másrészt bizonyos feladatok végrehajtása lényegében egyszerűbbé válik.

Szíves érdeklődésére készséggel áll rendelkezésére a Híradástechnika Szövetkezet Számítástechnikai Vevőszolgálat: 1116 Budapest XI., Temesvári u. 20. Telefon: 869-522/130

Beszerezhető: Híradástechnika Szövetkezet Számológép Service Budapest VII., Thököly u. 32. Telefon: 422-972

HÍRADÁSTECHNIKA SZÖVETKEZET
H-1519 Budapest, Pf. 268.
Telex: 22-6151 htsz h

A MINŐSÉGÜGY KÖZÜGY

A minősítő formula elemzése

Most a gépmínősítő számot adó

$$M = \frac{T + \dot{O} + S + O}{\dot{A}}$$

formula elemzése lesz a feladatunk.

Emlékeztetünk a betűk jelentésére. M a minősítő szám, T a közvetlen elérési tár befogadóképessége, \dot{O} , S , O pedig rendre az 1 másodperc alatt elvégezhető lebegőpontos összeadási, szorzási és osztási műveletek száma, \dot{A} a gép ára.

Formulánk viszonylag egyszerű, könnyen használható. Ránézésre megállapítható, hogy több lényeges dolgot helyesen ragad meg. Mégis hiányérzetünk van vele kapcsolatban: vannak ugyanis fontos tényezők, amelyekre egyszerűen *érzéketlen*. Néhány ezek közül, amiket egy igazán jó minősítő formulának — ha ilyen egyáltalán van — figyelembe kell vennie:

— Nem választja szét a fixpontos és a lebegőpontos műveleteket. A fixpontos műveletekből általában egy másodperc alatt lényegesen több végezhető el, mint lebegőpontos megfelelőikből.

— Nem szerepelteti a többszörös pontosságú műveleteket. Sőt még azt sem veszi számításba a minősítésnél, hogy ilyenek egyáltalán elvégezhető-e a géppel.

— Minden műveletet — melyet figyelembe vesz — egyformán vesz számításba. Ritka az olyan feladatkör, ahol a szereplő műveletek nagyjából egyenlő számban végezzük el.

— Számításon kívül nagy más gyakori (például összehasonlítás eredményét képező) műveleteket. Vannak olyan, például ügyviteli adatfeldolgozási feladatok, amelyekben nagyon sok az összehasonlításra épülő tevékenység, például a lebegőpontos műveletekhez képest.

— Nem veszi számításba a nem közvetlen elérési táraikat. Sőt azt sem, hogy ilyenek egyáltalán csatlakoztathatók-e

a géphez. A nem közvetlen elérési táraikból sokféle létezik. Ezek azért is fontosak, mert megfelelő tárolásszervezéssel (ezt megvalósító programok révén) elérhető, hogy a különböző táraiban levő információkat a felhasználó úgy kezelhesse, mintha azok egy nagy, közvetlen elérési tárban lennének.

Az árakkal kapcsolatban sem fogunk szűkölködni nehézségeket okozó kérdésekben. Mint mondtuk, az árat forintban számoljuk. A formula azonban túl egyszerűen intézi el a pénzügyeket.

— Nem veszi tekintetbe, hogy a gépért forintban, szocialista vagy pedig tőkés valutában kell-e fizetni. Ez pedig nem mindegy. Hiába van meg valakinek a szükséges mennyiségű forintja, ha a gép árat dollárban kéri, nem biztos, hogy azonnal sikerülni fog forintjának dollár váltása.

Az ár egyfajta törléstést, egyösszegű kifizetést sugall. Ez azonban csak egy lehetőség, egy ellenszolgáltatási mód a sok közül.

— A formula nem veszi figyelembe a különböző ellenszolgáltatási lehetőségeket. Lehetséges például részletfizetési „kedvezmény”. Ilyenkor a részletek összege több szokott lenni annál az árnál, amit az egyösszegű vételkor kér az eladó. A bérletvétel (leasing, ejtsd lízing) is gyakori megoldás. Az eladó ilyenkor egy ideig tulajdonképpen bérbeadó, az üzleti kapcsolat tulajdonképpen egy ideig bérlet. A bérlet végén válik a bérbeadó eladóvá. A bérlet és a bérletvétel között az a különbség, hogy a bérletnél bérleti díjat fizet a bérlő, és a szerződés lejártával a bérlet objektumot visszaadja a bérbeadónak, a bérletvételnél viszont az eladó a bevétől bérletvételi díjat kap, és a bérletvételi szerződés lejártával a bérlet objektum a bérbevevő tulajdonává válik.

Ha a bérleti vagy a bérletvé-

teli időszak közben megszünik a kapcsolat, a gép visszakérül a tulajdonoshoz, a másik fél pedig nem fizet tovább bérleti, illetve bérletvételi díjat. Ha viszont szabályosan lejár a bérletvételi szerződés, akkor annak lejártával a bérlet gép a bérbevevő tulajdonosa lesz. Ez abban különbözik a részletfizetéstől, hogy a részletre vásárolóknak nincs jogi lehetősége arra, hogy a részletfizetési időszak alatt meggondolja magát és visszalépjen a vételtől. A bérletvétel teljes összege ezért még a részletfizetési összegnél is lényegesen magasabb szokott lenni.

Szembeötlő érzékletlenségein kívül vannak formulánknak más, „finomabb” hiányai is, amelyek azonban nem kevésbé jelentősek, mint azok, amiket eddig észrevettünk. Foglalkozunk ezekkel is egy kicsit!

Feltételezve, hogy az összes eddigi tökéletlenséget hiánytalanul kiküszöbölő minősítőszámot előállító formulát sikerült kifejlesztünk, vajon e formulával nagyobb biztonsággal indulhatnánk vásárolni, mint az eredetivel? Sajnos nem.

Vegyük két teljesen azonos gépet! Formulánk nyilvánvalóan azonosan fogja minősíteni ezeket. Akkor is, ha e pillanatban csúsztatják elénk mindkettőt a gyártó sor végén levő sikeres minőség-ellenőrzés után, és akkor is, ha az egyik évke óta raktárban volt, a másikat pedig használták. Bíráló megjegyzésünk lehet a következő.

— Nem veszi figyelembe a gép állagát és ún. „erkölcsi” korát. A gép saját életkora sem közömbös adat, mert akkor sem, ha új. Joggal lehetnek fenntartásai egy 3 éves, de állagát tekintve használatlan, tehát „új” géppel kapcsolatban. (Az „erkölcsi” kornak semmi köze az erkölcshez, azt értik rajta, hogy mennyi idő s szóban forgó típus. Egy termék erkölcsi életkora nem mindig kapcsolatos használati értékével. Lehetnek egészen új

típusok, amelyek sokkal korszerűbbek az előzőeknél, azonban gyorsan avulnak. Úgy, hogy az avulásban megelőzik a korábbi termékeket. Így például egy kisipari, faszegekkel szegelt vásári tucatszirma erkölcsi kora a mai csizmákéhoz képest matuzsálemi, viszont a maiakkal ellentétben sohasem ázott be.)

Nem foglalkozunk a bevalótlan kishibás, a felújított, a javított, a használt és a selejtes gépek kereskedelmi forgalmával és minősítésével, garanciális és egyéb kérdéseivel. Ezek inkább az ószerasztakmához, mintsem a számítástechnikához tartozó — egyébként fontos — kérdések.

Tovább lehet az életkor kérdését taglálni, ha azt is figyelembe vesszük, hogy a gép mely egységének mi a kora. És ez a kérdés továbbvezet egy minden eddiginél nehezebb feladathoz. Külön egysége-e a szoftver a gépnek, vagy nem, nem kívánjuk vizsgálni. Az azonban biztos, hogy nélkülözhetetlenül hozzá tartozik. Az árba mindenesetre beszámítottuk.

— Formulánk nem minősíti a gép programkészletét. Sem az operációs rendszert, sem az alapprogramokat, sem a felhasználói programokat. Sem azokat, amelyeket megvásároltunk (vagy meg kívánunk vásárolni), sem azokat, amik az eladónál rendelkezésre állnak.

Mi köze lenne a minősítésnek az eladónál rendelkezésre álló programokhoz? Azokat ügysem használjuk. Most nem, de később lehet, hogy szükségünk lesz egyike-másikra. És ezzel beleütköztünk a bővíthetőség kérdésébe.

— Formulánk érzéketlen a szoftver bővíthetőségével szemben. Csak a szoftverével szemben? Nem.

— Formulánk a hardver bővíthetőségét, alakíthatóságát sem veszi figyelembe. Nem mindegy ugyanis, hogy egy géphez akár ennél, akár annál

az eladónál milyen programkészlet áll rendelkezésünkre. Nem mindegy például, hogy egy géphez egyetlenegy, vagy ezenkívül még 20-féle nyomtató illeszthető. Nem mindegy, hogy a gép valamilyen szempontból felkellette-e az érdeklődését akár a szoftverfejlesztőknek, akár a különböző perifériális egységeket vagy bővítményeket gyártó vállalatoknak. A számítástechnikai berendezésekhez ugyanúgy kell pótkatrész, mint a gépkocsiszárnálathoz. Ez itt is, ott is a karbantartók dolga. Vannak azonban jellegzetesen számítástechnikai „pótkatrész”-fajták is, a fejlesztést szolgáló hardverbővítmők, csatlók, illesztők, az újabb és újabb perifériális egységek és a szolgáltatási kört bővítő és a kényelmet fokozó programok. Ezeknek a „pótl-

atrészeknek” egy része a vásárlás pillanatában már rendelkezésre áll. Másik részükről már bejelentették, hogy rövidesen megjelenik. A továbbiakat pedig csak becsülni lehet, mint az új gép várható „erkölcsi” elavulási idejét és

— tényleges élettartamát, amelyet formulánk szintén nem vesz figyelembe. Az élettartamkérdésekben azonban meglehetősen nehéz a megbízható becslésadás.

A gép több egységből összetett szerkezet. Más az élettartama a különböző hajlékonylemez-egységeknek, a különböző hőnyomatóknak, a tintacsépes nyomtatóknak, a lézernyomtatóknak, a fixlemez egységeknek stb. Sőt például az egyes nyomtatóknál bizonyos — gyorsabban elhasználódó — alkatrészeknek is más-

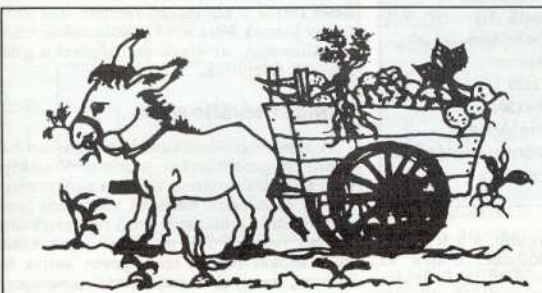
más az élettartama. Mit értünk hát a gép élettartamán? A kérdést csupán felvetni tudjuk, megválaszolni nem.

Most azonban nem is az ilyen nehéz részfeladatok megoldása volt a célunk, hanem az, hogy érzékeltessük a gépmínősítés feladatának összetett és nehéz voltát. Reméljük, emellett még azt is sikerült el-értünk, hogy az olvasó kellő kritikával fogadja majd a különböző gépek minősítésével kapcsolatos ítéleteket, és mint vásárló, nemcsak annak hisz majd, amit a gyártók és viszonteladók mondanak — ez minden udvarias számítástechnikus kötelessége —, hanem saját józan eszének is.

Annak érdekében, hogy megbízhatóan el tudjuk kövölni a felelőtlen és megalapozatlan gépválasztásokat a

körültekintően és gondosan elvégzetté, mindenekeelőtt szükségünk van a legfontosabb bírálati szempontok összegyűjtésére. A minél nagyobb mértékű teljességre törekvés minden lelkismeretes gépválasztónak kötelessége és elemi érdeke is. Mivel itt mi nem gépválasztással, hanem gépválasztás-módszertannal foglalkozunk, a szempontokra vonatkozó bőségnek elebe kell helyeznünk a szempontok tipikusságát.

A tipikus szempontok alapján úgyis könnyebb a konkrét választási feladat sajátos szempontrendszerét kialakítani, mint egy nem rendezett — de sokféleképp rendezhető — szempontmögöből kiérezni a minősítés szempontjából lényeges elemeket.



Zöldségeskert

Neumann János — John Von Neumann. Színes, magyar dokumentumfilm. Rendezte: DÉNES GÁBOR A számítógép „atyjáról”, a magyar származású Neumann Jánosról emlékeznek világhírű egyetemek, világhírű tudósai, közeli munkatársak és rokonok. Többek között WIEGNER JENŐ, TELLER EDE fizikus; PÓLYA GYÖRGY, GEORG DANTZIG matematikus; ISAAC ASIMOV író, SZENT-ÁGOTHA JÁNOS agykutató. ... Egy időben volt közöttük egy mondan, hogy kétféle emberi lény van a Földön: emberek és magyarok. Mert ők annyira kiválóak voltak, hogy alig lehetett elhinni, hogy egyszerűen csak magyarok.” (Howá menjünk moziba? Budapes-

ti filmtájékoztató. 1986. jan. 30-tól febr. 5-ig)

Neumann nem „a számítógép atyja”, csak egy azok közül. Úgy véljük, hogy Neumann tudósok van olyan nagy, hogy nincs szüksége arra, hogy mások kárára — akik közül nem egy őt megelőzően ért el fontos eredményeket — egyedüli „atyasággal” tüntessék ki. Atanasoff, Berry, Eckert, Kozma, Maunchly, Zuse semmiképp sem hagyható ki a modern gép atyáságából. Sőt Turing és Shannon sem. Nyilvánvaló az atyai szerepe Babbage-nak, Hollerith-nak, Jacquard-nak és még több nagy tudósak is, a régebbi „atyákról” most nem beszélve. Sőt van a gépnek nőne-mű „atyja” is, Lovelace.

Az, a faszizmus leggőzösebb

hőbortját is túllicítáló gondolat pedig, hogy „kétféle emberi lény van a Földön: emberek és magyarok”, sohasem volt és sohasem lehetett sem tisztességes magyar embernek, sem komoly tudósaknak a gondolata, ezért az ilyenféléktől most is nyomatekosen elhatároljuk magunkat, és a magunk részéről egyértelműen károsnak tartjuk.

„Alkalmazói software” a könyv címe. Első mondata pedig: „A könyv megírásának célja a COMMODORE 64 gépekre megvő gyári szoftveranyagok kezelésének könnyebbé tétele a magyar felhasználó számára”. Aztán pedig egy mondat a címodal után: „Szoftver és dokumentáció másolása tilos az 1969. évi III. törvény 13. paragrafusa és a 18. paragrafus 1. bekezdése alapján”. (Alkalmazói softwarek I. Novotrade RT)

Az előbbi megállapítás persze nem igaz. De ha igaz volna is, akkor sincs baj, mert csak a szoftver másolása tiltatik, a szoftver anyagok és softwarek nem. A kedves szülőök figyelmét azonban felhívjuk arra, hogy kiskorú gyermeiket semmiképp se ijesztessék, hogy „elvisz a szások ember” vagy „odaadlak a Drakulának”. A softwarek zót pedig ki se ejtsék a szájukon, mert a gyerekek azt hiszik erről, hogy valamiféle szörnyeteg, az ijedség életré szőlo nyomot hagyhat a lelkükben, fel-

nőtt korukra pedig különböző pszichés zavarokba nyilvánulhat meg a gyermekkori trauma: sohasem tanulnak meg például értelmesen beszélni vagy helyesen írni, és emiatt legfeljebb csak számítástechnikai pályára lesznek alkalmasak.

„CHIP, a számítógép memóriájának apró, de legfontosabb eleme: lehetővé teszi a műveletvégzés sebességének milliómód másodpercekre csökkentését” — ez van a címlapon. A borító belső oldalán pedig: „A chip többek között a számítógép apró, de legfontosabb eleme”. (Siker 86/1.)

A chip elsősorban nem elem, nem alkatrész, hanem megvalósítási forma, egy áramkör megvalósítási formája. Ma már nagyon sok áramkörnek van chip formájú, azaz speciális lapkán elhelyezett mikrominiaturizált megvalósítása. Helytelen „chipről” mint tárelemről beszélni. A helyes: a tár is állhat chipekből — pontosabban IC-kből — és sok másféle számítógép-alkatrész is. Van már olyan chip, amely tokozva, azaz IC formában egy egész mikroszámítógépet tartalmaz — természetesen perifériális egységek nélkül.

Ez a hasznos lap az Újtitok Lapjának utóda. Kár, hogy egyébként mindönesen érdekes első száma címlapján számítástechnikai hiba éktelenkedik. Hibamentes folytatást és sok sikert kívánunk a Sikernek.

Assemblerek, cross-assemblerek

3. A (cross)-assembler program mint rendszer

A számítástechnikai szaknyelvben gyakran használatos a számítógépes programokkal kapcsolatban a „rendszer” fogalma. Rendszernek neveznek egyes összetettebb programtermékeket vagy programok összefüggő együttesét (operációs rendszer, adatbázis-kezelő rendszer stb.). Nem kívánunk itt a rendszer fogalmának pontos értelmezésével foglalkozni, de a szóhasználat mögött nyilvánvalóan az áll, hogy ezekben az esetekben olyan, bonyolult belső összefüggésekkel rendelkező objektumokról van szó, amelyek előírt környezeti feltételek között, adott gerjesztésre (a bemeneti változóknak adott értékeire) adott módon választanak, illetve ha a program nem az előírt módon viselkedik, akkor hibás működésről beszélünk (de ez már a programmegbízhatóság kérdéséhez tartozik). „Egyszerűbb” programok, mint például egy fordítóprogram esetében már ritkábban használják a rendszer megnevezést, noha nem nehéz belátni, hogy elvi azonosságuk alapján ez éppoly jogosan megilleti őket, mint a fenti programkomplexumokat.

Természetesen a rendszer fogalmának a számítógépes programokra való alkalmazása önmagában nem sok hasznot jelent. Értelmet akkor nyer, ha a fogalommal együtt igénybe vesszük azokat az eszközöket is, amelyek a rendszerek általános leírásához, vizsgálatához, kezeléséhez a rendelkezésünkre állnak. Ha tehát rendszernek tekintjük a programokat, akkor segítségül kell hívni a rendszermodellezés által kidolgozott módszereket, s meg kell vizsgálni, hogyan alkalmazhatók ezek a programok viselkedésének leírásához, illetve hogy milyen haszon származhat alkalmazásukból. Ez az, amit általában a programrendszertervek, -ismertetők, -leírások, -kezelési útmutatók elmulasztanak.

Ezért mutatjuk most be ennek az eljárásnak az alkalmazását a (cross)-assembler programok példáján.

Rendszerállapot és rendszerváltó

Minden rendszer leírásához használt fontos attribútumok a rendszerállapotok és rendszerváltók. Meg kell tehát először is keresnünk ezek megfelelőjét (cross)-assembler programunkban, illetve ennek modelljében.

A rendszerállapotok és rendszerváltók kijelölése azonban nem egyértelmű, mindkettőt különböző szinten is lehet definiálni. A rendszerállapotok meghatározásakor két szélsőséges eset képzelhető el:

— csak az aktív-inaktív állapotokat különböztetjük meg egymástól (azaz, hogy a rendszer működik-e vagy kikapcsolt állapotban van-e);

— minden, a rendszerműködés folyamában eleminek tekinthető művelet végrehajtásához külön rendszerállapotot rendelünk. Így a rendszer működése állapotátmenetek időbeni sorozatával írható le.

Nyolcreszes sorozatunk a mikroszámítógépek assemblereiről, cross-assemblereiről szól. Célja, hogy a cross-assemblerek példáján keresztül megismertesse az olvasót az assembler programok működésével. A bemutatáshoz a rendszermodellezési eszközöket használjuk fel, s így készíthet el az assemblerek működésének egy szabványosított algoritmusát, modelljét.

1. (Cross)-assemblerek és a rendszermodellezés

2. Az Intel 8080 assembly nyelv

3. A (cross)-assembler program mint rendszer

4. A rendszer működése I. — A fordítás két menete

5. A rendszer működése II. — Táblák és adatterületek

6. Az operátorkészlet I.

7. Az operátorkészlet II. — A rendszer kapcsolatábrája

8. Példák a rendszer működésére

Ez utóbbi az általánosabb, s a leggyakrabban alkalmazott módszer. Ebben az esetben az egyes állapotok jellemzésére szolgálnak az ún. rendszerváltók. A rendszerek viselkedését elvileg végtelen sok elsődleges és származtatott, közvetlenül vagy közvetve mérhető változó időfüggvénye írhatja le. A gyakorlatban természetesen csak véges számú rendszerváltó kezelésére vagyunk képesek, tehát a valamilyen szempontból kevésbé lényegesnek ítélteteket figyelmen kívül kell hagynunk.

Esetünkben mind a rendszerállapot, mind a rendszerváltó definíciók három némiig eltérnek a fenti elvektől. Nem lenne ugyanis célszerű, ha minden műveletnél önálló rendszerállapotot tételeznék fel — például megőkövívalasztó, címkevező-feldolgozósi, háttértárra való írási állapotok stb. szerepelnének —, mert akkor a modell állapotábrája tartalmazna a kidolgozandó kapcsolatábrák működési modellel egyezne meg. Ezzel szemben programunk állapotterméddeljével inkább azt szeretnénk elérni, hogy vele a rendszer globális folyamatát tudjuk jellemezni.

Rendszerállapotok

Ezért csak néhány, elemi állapotokat összefogó rendszerállapotot jelölünk ki, amelyeknek mindegyikében előírt műveletek, előírt sorrendben belől végrehajtási folyamata játszódik le. Ezzel tudatosn vállaljuk azt az ellentmondást, hogy a tulajdonképen

pen a rendszer működéséről vett pillanatképeket tekinthető rendszerállapotokat egy-egy rövidebb-hosszabb ideig tartó rész-folyamattal azonosítjuk. A (cross)-assembler program modelljében megkülönböztetendő állapotok (állapottipusok) az alábbiak lesznek:

— alapállapot, amelyben sem makródefinió, sem makrókifejtés nem történik, hanem csak elsődleges forrásprogramsorok fordítása, feldolgozása;

— makródefiniációs állapot;

— makrókifejtési állapot, mégpedig az egymásra skatulyázás szintje szerint különböztetve őket;

— az első és a harmadik típusú állapoton belül meg kell különböztetni a fordítási és a fordításkimaradási állapotokat. Az utóbbiba akkor kerül a rendszer, ha a feltételek fordítás feltétele nem teljesül (IF direktíva 0 értékű operandussal jelent meg);

— végül mindegyik előző típuson belül külön kell választani a fordítás ún. két menetéit (erről a következő részben lesz szó), és így jönnek létre azok a különböző rendszerállapotok, amelyek segítségével a gráfmodell felépíthető.

Rendszerváltók

A rendszerváltókkal kapcsolatban hasonló megszorításokat teszünk. Voltaképpen a fordítóprogram minden programváltozója és minden használt adatmező (pufferterületek, táblázatok stb.) rendszerváltóként fogható fel, mi mégis ezek közül csak azokat fogjuk ténylegesen annak tekinteni, amelyek a rendszerállapotok azonosítására, illetve az egyes állapotok közötti átmenetek vezérlésére szolgálnak. Ilyen lesz például egy MIND nevű rendszerváltó, amelynek értéke azt mutatja meg, hogy az adott időpontban makrókifejtés történik-e vagy sem, és ha igen, akkor hányadik belső szinten levő hívás kifejtésére kerül sor.

Egyéb programváltozók

A többi, rendszerváltóznak nem tekintett programváltozó igen sokféle feladatot láthat el. Léteznek közöttük számlálók, puffterületek szabad területre mutató változók (ponterek), lemezterület hozzáférési címeket tartalmazó változók, valamilyen adatmező-tartalom típusának azonosítására szolgáló változók (flagek), továbbá egyéb, a program működése során keletkező információk — például hibajelzések — átmeneti tárolását végző indikátorok.

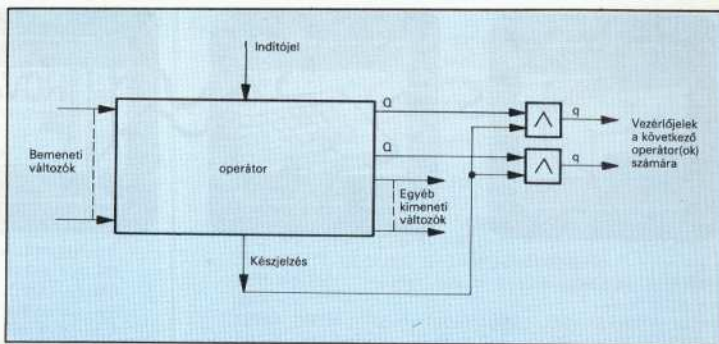
Operátorok

A rendszerként felfogott (cross)-assembler program működését, mint azt az első részben már említettük, „szabványosított” működésű operátorok segítségével írjuk le. Az operátorok olyan funkcionális modulok, amelyek a feldolgozási (forrásprogram-fordítási) folyamat egy adott szinten

elemek tekintett műveletet najtják végre. Az operátorok a feladatuk elvégzéséhez szükséges rendszerváltozókhöz, egyéb programváltozókhöz, adatmezőkhöz hozzá tudnak férni, s azokat módosítani is képesek. Az egy operátor által használt és esetleg módosított programváltozók és adatmezők alkotják az operátor bemeneti és kimeneti változóit, amelyeket az operátor definiálásakor kell kijelölni.

A (cross-)assembler program modellezéséhez kidolgozott operátorkészlet nem „egyenszilárdságú”, azaz közöttük egyszerűbb és összetettebb feladatokat ellátó operátorok is vannak: az egyszerűek akár valamilyen elemi logikai műveletet (logikai összeadás vagy szorzás), akár például egyetlen programváltozó adott szempontú vizsgálatát végzik el, az összetettek pedig igen bonyolult adatkezelési műveleteket hajthatnak végre. Ilyen összetett operátor például az operandusmezőt kiértékelő operátor. Ezek az operátorok szükség esetén tovább bonthatók egyszerűbb működésű operátorokra. Ezt a felbontást példánkban nem végeztük el.

Az operátoroknak magával a fordító-



3. ábra

programmal való kapcsolata egyszerű, mert általában is kimondható, hogy egy operátor egy szubrutinnal valószínűsíthető meg, amelynek paraméterei az operátor bemeneti és kimeneti változói. Az egyszerűbb esetekben azonban nincs szükség külön szubrutinra, mivel ilyenkor a főprogramrészen belül elhelyezhető a néhány utasítás segítségével „kódolt” operátorok.

Az operátorok működési módja

Külön kell foglalkoznunk az operátorok működési módjával, azzal, hogy miként tudjuk modellezni, hogy az egyes operátorok szekvenciálisan aktivizálódnak, egymásnak adva át a vezérlést. Ha az operátorok valóságos megjelenési formáját, a szubrutinokat tekintjük, az egymás utáni gerjesztés szinkronizálása automatikusan megvalósul, hiszen mindaddig, amíg a felhívott szubrutin nem végzi el feladatát, addig a vezérlés rajta van, ha pedig működését befejezte, akkor visszaadja a vezérlést a felhívó programnak, és következhet a megfelelő következő szubrutin kiválasztása és indítása.

Más a helyzet a modellben. Itt az operátorokat össze kell kapcsolni ahhoz, hogy a rendszer működését kapcsolatábrában tudjuk leírni, s az összekapcsolt operátorok nem működhetnek aszinkron módon, mert minden operátor működése feltételezi a megelőző operátor(ok) befejezett működését. Az operátorok működését tehát szinkronizálni kell egymáshoz. Erre szolgál az operátorok két új „változójának”, az indítójelnek és a készültségnek a bevezetése. Vegyük észre, hogy természetesen bonyolítottuk a valósághoz képest a modellezési eszközt, mert különben a modell nem tudt volna felépíteni.

A két változóval történt kiegészítés után az operátoroknak a modellben használt általános jelölési módja az 1. ábrán látható.

Az operátor az indítójel-vezetéken bekövetkező 0→1 átmenet hatására kerül aktív állapotba. Ha működését befejezte, és az összes kimeneti változója elnyerte a szükséges (új) értéket, készültséget ad, ami a soron következő operátor indítójeleket szolgálhat.

Ezeknek a jeleknek két feltételt kell teljesíteniük. A készültségnek elegetöően hosszú ideig a kimeneten kell maradnia, mert

léteznek olyan esetek, ahol több, egymástól függetlenül működő operátor készültségének együttes megjelenése indíthatja csak el a következő operátort. A készültség ugyanakkor nem állandósulhat a kimeneten, mert léteznek hurkok is az operátorok gerjesztési grájfjában, s minden ilyen operátor újbóli aktivizálása előtt annak indítójelét és készültségét alapállapotba kell hozni.

Ezért feltételezzük, hogy minden operátor indítása után alapállapotba hozza valamilyen olyan operátor készültségét, amely az adott operátor indításában közreműködött. Az indítójel és a készültség időbeni változását szemlélteti a 2. ábra egy operátor működési időtartama alatt.

Léteznek olyan operátorok is, amelyeknél a kijelölt művelet eredménye valamilyen logikai változó, vagy van a kimeneti változók között egy logikai változó, ami csak igaz vagy hamis értéket vehet fel, és ez a kimeneti változó vezérli a következő operátor(ok) működését, tehát feltételese elágazás történik a modellben. Ebben az esetben az operátoroknak a 3. ábrán bemutatott, de a kapcsolatábrában sehol sem jelölt működési módja érvényes.

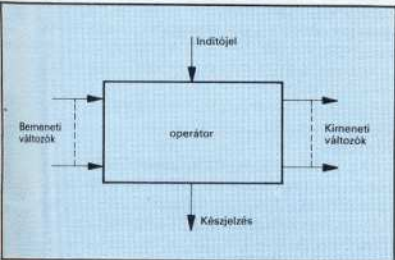
Természetesen az elmondottak nem vonatkoznak az egyszerű logikai műveletet (logikai összeadás és szorzás) végző operátorokra. Ezeknél nincs indítójel és készültség.

Ezzel áttekintettük, illetve definiáltuk a (cross-)assembler programok rendszermodellezési eszközökkel történő tárgyalásához szükséges legfontosabb fogalmakat és alapelveket. A következő részben áttérhetünk a rendszer működésének részletesebb vizsgálatára. A felülről lefelé való haladás elvét alkalmazva előbb a globális rendszerműködést írjuk le, segítségül hiva az állapotábrák grafos szemléltetését is. Ezután további korlátozásokat és feltételezéseket vezetünk be a vizsgált (cross-)assembler programmal kapcsolatban, majd megadjuk a szükséges programtáblák definícióit, és újra, már részletesebben végigkövetjük a program működését.

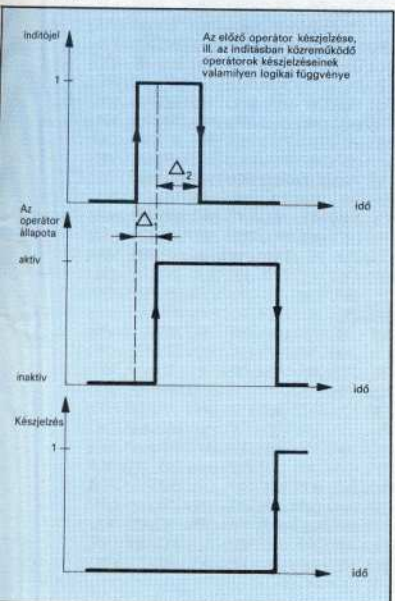
Mindezzel az a célunk, hogy pontosan meghatározzuk, milyen feladatokat kell a programnak ellátnia, s ehhez milyen műveletek, milyen módon való végrehajtása szükséges. Ezek ismeretében térhetünk rá az operátorok definiálására és az operátorok segítségével a program működésének kapcsolatábrás modellezésére.

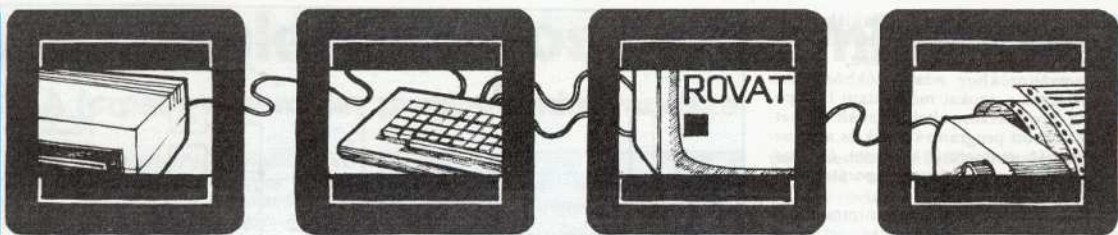
VÁRGEDŐ TAMÁS

1. ábra



2. ábra





KÉTSZERES NAGYSÁGÚ, TECHNIKAI TÍPUSÚ KARAKTEREK

A feladatom az volt, hogy készítsék egy 16 x 16 rácspontból álló, technikai betűtípusú karakterkészletet úgy, hogy könnyen felhasználható, ugyanakkor egy képernyőn belül az eredeti karakterkészlettel keverhető legyen. Mivel a karakterkészlet átkapcsolásakor a képernyő összes felirata az új készlet szerinti lesz, ezért a feladat csak úgy oldható meg, ha az eredeti és a kétszeres nagyságú karakterek is egy készleten belül érhetők el.

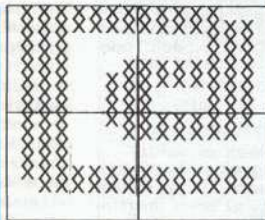
A 0-63 képernyőkódú nagyított karaktereket készítettem el. Az eredetiek közül ugyanezen kódú karakterek és inverzeik maradtak meg a saját helyükön, az új, nagyított karakterek a 64-127 kódúak és inverzeik helyére kerültek. A módosításhoz a nagybetűgrafika-készletet használtam fel, így a nagybetűk és inverzeik közvetlenül elérhetők, a módosított karakterek részei pedig a grafikus jelek és inverzeik helyére kerültek.

Tekintettel arra, hogy egy nagyított karakter általában négy eredeti helyét foglalja el, a 64 darab a teljes készlet helyét igényelné. A rendelkezésre álló 128 helyre csak úgy férnek el, ha gondos tervezéssel egy-egy részlet több karakterhez is felhasználható. A tervezésnél a technikai betűtípus jellegzetességei mellett figyelembe vettem azt is, hogy a tévéképernyőn két vízszintes pontsor kb. ugyanannyi vastag, mint három függőleges. Az az jel képtől például az *ábrán* láthatjuk.

Két egymás fölötti karakterpozíciót foglalnak el négy helyett az alábbi jelek:

() + - = . , : ! !

A megszerkesztett karakterek adatainak DATA sorokba való írásához a 16-os szám-



rendszert választottam, mert így sokkal könnyebb a rajzokról a szükséges adatokat leolvasni. Például az az jel bal felső karakterének első bájta a 3F számmal megadható.

Szigorúan különválasztandó két tevékenység: a karakterkészlet elkészítése és felhasználása. Az elkészítést két program végzi. Az első a PROGRAMKÉSZÍTŐ, átmásolja a karakter ROM-ot, betölti a hexadecimális kódokkal megadott saját karaktereket, majd az így keletkezett új készletet a NEW CHAR. nevű programfájba menti lemezre. Ügyel a mentés helyes voltára is. A második a SEQ KÉSZÍTŐ, egy 16x16 CHR. nevű szekvenciális fájlhoz létre a lemezen, az általában négy részből és a szükséges vezérlőkaraktérből álló nagyított karakterek gyors elérésére. Mindkét programba beépítettem egy ellenőrző összeget az adatok elírásából származó esetleges hibák kiküszöbölésére. A karakterkészlet csak akkor használható, ha az előző két fájl hibátlanul, lemezen tárolva már a rendelkezésünkre áll. Magnón történő tároláshoz a fájlnyitási és -zárási, valamint a mentési töltőutasítások értekezésszerű átírása szükséges. Ekkor elmarad a lemezegység esetleges hibáját ellenőrző rutin is.

Az így elkészített karakterek

```

10 REM ***** PROGRAM KESZITO *****
11 DIMA(511):OPEN15,8,15
12 OPEN8,8,8,"@:NEW CHAR.,PRG,WRITE"
13 PRINT#8,CHR$(0);CHR$(8);
14 B=53248:GOSUB20:GOSUB30
15 B=54272:GOSUB20:GOSUB30
16 CLOSE8,15:IFW=81981THENEND
17 PRINT"HIRA VAN A DATA SORBAN!":STOP
18
19 REM ***** CHAR. ROM MASOLAS *****
20 POKE56334,0:POKE1,51
21 FORC=0TO511:A(C)=PEEK(B+C):NEXTC
22 POKE1,55:POKE56334,1
23 FORC=0TO511:PRINT#8,CHR$(A(C));
24 GOSUB26:NEXTC:RETURN
25 REM ***** DISK ERROR SBR. *****
26 INPUT#15,A1,A2#A3,A4
27 IF#1=0THENRETURN
28 PRINT"DISK ERROR:";A1;A2#;A3;A4:STOP
29 REM ***** SAJAT CHAR. BETOLTES *****
30 FORL=1TO512:READ#
31 FORC=1TO2:A(C)=ASC(MID$(A#,C,1))-48
32 IF#C>9THEN#A(C)=A(C)-7
33 NEXTC:D=A(1)*16+A(2):W=W+D
34 PRINT#8,CHR$(D):NEXTL:RETURN
35
100 DATA 3F,7F,70,70,71,73,73,73,FC,FE
101 DATA 0E,0E,FE,FE,0E,0E,73,71,70,70
102 DATA 7F,3F,00,00,FE,FC,00,00,FE,FE
103 DATA 00,00,0F,1F,1C,1C,3F,7F,7C
104 DATA F0,F8,38,38,38,FC,FE,0E,7C,7C
105 DATA 7C,7C,7C,38,00,00,0E,0E,0E,0E
106 DATA 0E,04,00,00,3F,7F,70,70,70,7F
107 DATA 7F,7C,7C,7C,7C,7C,7F,3F,00,00
108 DATA 0E,0E,0E,0E,FE,FC,00,00,3F,7F
109 DATA 70,70,70,78,7C,7C,FC,FE,0E,0E
110 DATA 00,00,00,00,00,00,04,0E,0E,FC
111 DATA 00,00,FC,FE,0E,0E,0E,0E,0E,0E
112 DATA FC,FE,0E,0E,00,7C,7E,0E,20,70
113 DATA 70,70,70,7F,7F,7C,04,0E,0E,0E
114 DATA 0E,FE,FE,0E,10,38,38,38,3C,3C
115 DATA 3E,3E,00,20,70,70,7F,3F,00,00
116 DATA 3E,3E,3E,3E,FE,FC,00,00,10,38
117 DATA 38,38,38,FC,FE,0E,3F,7F,70,70
118 DATA 70,70,70,70,7C,FE,3E,3E,3E,1E
119 DATA 0E,0E,70,70,70,70,7F,3F,00,00
120 DATA 70,70,71,73,7F,3F,00,00,0E,0E
121 DATA FE,FE,FE,FC,00,00,3F,7F,70,70
122 DATA 70,7F,3F,00,FC,FE,0E,0E,00,FC
123 DATA FE,3E,7F,7F,03,03,03,03,03,03
124 DATA FE,FE,80,80,80,00,00,0E,03,03
125 DATA 03,03,03,01,00,00,0E,0E,0E,0E
126 DATA 00,00,00,00,20,70,70,70,70,70
127 DATA 7C,7C,04,0E,0E,0E,0E,0E,0E,0E
128 DATA 38,7C,7C,7C,7C,7C,7C,04,0E,0E
129 DATA 0E,0E,0E,0E,0E,1C,3C,1C,1C,1C

```



```

130 DATA 1F,0F,00,00,38,38,38,38,F8,F0
131 DATA 00,00,20,70,70,70,70,3F,1F,3E
132 DATA 04,0E,0E,0E,0E,FC,F8,1C,20,70
133 DATA 70,70,70,7F,3F,03,04,0E,0E,0E
134 DATA 0E,FC,F8,E0,3F,7F,70,00,07
135 DATA 3F,7E,FC,FE,0E,0E,7E,FC,E0,00
136 DATA 1F,3F,38,38,3F,3F,3E,FC,FC
137 DATA 00,00,00,FC,FC,00,3E,3E,3E,3E
138 DATA 3F,1F,00,00,00,00,00,00,FC,FC
139 DATA 00,00,3E,3E,3E,3E,3E,1C,00,00
140 DATA F8,FC,1C,1C,1C,FC,F8,00,7F,FF
141 DATA E1,C1,E1,F1,F9,F9,FE,FF,C7,C7
142 DATA C7,C7,C7,C7,F9,F9,F9,F9,70
143 DATA 00,00,C7,C7,C7,C7,82,00,00
144 DATA 40,E1,E1,E1,E1,F1,F9,F9,82,C7
145 DATA C7,C7,C7,C7,C7,C7,F9,F9,F9
146 DATA FF,7F,00,00,C7,C7,C7,FF,FE
147 DATA 00,00,3F,7F,70,70,00,00,3F
148 DATA FC,FE,0E,0E,0E,0E,7E,7F,7C
149 DATA 7C,7C,7F,3F,00,00,FC,00,00
150 DATA FE,FE,00,00,3F,7F,70,70,0F
151 DATA 0F,00
152 DATA F0,F8,38,38,38,FC,FE,3E,3F,7F
153 DATA 70,70,70,70,70,7F,F0,F8,38,38
154 DATA 38,38,3E,3E,3F,00,00,00,00
155 DATA 00,00,FE,F8,F8,FC,F8,70,00,00
156 DATA FE,FE,00,00,FC,FE,3E,3F,7F
157 DATA 70,70,70,7F,70,0F,1F,1C,1C
158 DATA 1C,3F,7F,70,FE,3E,3E,3E,1C
159 DATA 00,00,FC,FE,0E,0E,0E,1E,3E,3E
160 DATA 3F,3F,00,00,00,00,00,F8,FC
161 DATA 1C,1C,1C,3C,7C,7C,08,1C,1C
162 DATA 1C,3C,7C,7C,1E,3E,38,38,38
163 DATA 38,38,38,38,38,3E,1E,00,00
164 DATA 0F,1F,1C,1C,1C,7F,7F,F8,FC
165 DATA 1C,1C,00,00,E0,E0,1C,1C,1C,1C
166 DATA 7F,7F,00,00,00,00,00,FE,FE
167 DATA 00,00,78,7C,1C,1C,1C,1C,1C
168 DATA 1C,1C,1C,7C,78,00,00,03,07
169 DATA 0F,1F,3B,73,23,03,00,C0,E0,F0
170 DATA B8,9C,88,80,03,03,03,03,01
171 DATA 00,00,80,80,80,80,00,00,00
172 DATA 00,00,02,07,0E,1C,38,7F,00,00
173 DATA 00,00,00,00,00,FE,7F,7F,38,1C
174 DATA 0E,07,02,00,FE,FE,00,00,00
175 DATA 00,00,3E,1C,00,1C,3E,1C,00,00
176 DATA 3C,7E,7E,3E,0E,0E,00,00,00
177 DATA 0C,0E,07,03,1F,1F,00,00,18,38
178 DATA 70,E0,FC,FC,03,07,0E,00,00,00
179 DATA 00,00,E0,70,38,18,00,00,00
180 DATA 00,04,0E,0E,3F,3F,0E,00,00,20
181 DATA 70,70,FC,FC,70,70,3F,3F,0E,0E
182 DATA 04,00,00,00,FC,FC,70,70,20,00
183 DATA 00,00,03,03,1F,3F,38,38,3F,1F
184 DATA 80,80,FC,FC,00,00,F8,FC,00,00
185 DATA 3F,3F,03,03,00,00,7C,7C,FC,F8
186 DATA 80,80,00,00,78,FC,CC,CC,78
187 DATA 03,3F,02,07,07,07,07,FC,C0
188 DATA 1E,3F,33,33,3F,1E,00,00,0F,1F
189 DATA 1C,1C,1C,1C,3F,7F,F8,F8,00,10
190 DATA 38,38,FE,FE,3F,3F,00,00,00,1F
191 DATA 3F,3E,00,00,3C,7E,7E,3C,00,00
192 DATA 00,00,00,00,08,1C,1C,7F,7F,1C
193 DATA 1C,08,00,00,00,00,00,00,00,00
194 DATA 00,00,7F,7F,00,00,00,00,00,00
195 DATA 03,0F,04,0E,0E,0E,1E,7C,F0,C0
196 DATA 3E,7C,7C,7C,7C,38,00,00,00,05
197 DATA 0E,1C,38,70,70,38,1C,0E,06,00
198 DATA 00,00,00,00,60,70,38,1C,0E
199 DATA 0E,1C,38,70,60,00,00,00,00,00
200 DATA 00,00,00,00,00,7E,7E,00,00,7E
201 DATA 7E,00,00,00,00,00,00,00,00,00
202 DATA 00,00,00,00,00,00,00,00,00,00
203 DATA 00,00

```

```

480 REM ***** SEQ KESZTO *****
500 OPENS,0,8,"@:16X16 CHR.,SEQ.WRITE"
510 FORL=32T095:FORC=1T04:A$(C)="■"
520 READA:IFSON(A)=-1THENA$(C)="■"
525 W=W+A:A=ABS(A)
530 A$(C)=A$(C)+CHR$(A):NEXT
540 ONA+1GOTO560
550 PRINT#8,A$(1)+A$(2)+"■■■"+A$(3);
555 PRINT#8,A$(4)+"□":CHR$(13):GOTO570
560 PRINT#8,A$(1)+"■■■"+A$(3)+"□";
565 PRINT#8,CHR$(13);
570 NEXT:CLOSE8:IFW=6544THENEND
580 PRINT "HIBA VAN A DATA SORBAN!":STOP
590 :
1000 DATA 32, 32, 32, 32, 114, 0
1010 DATA-125, 0,-126,-126, 32, 32
1020 DATA-163,-164,-165,-166,-167,-168
1030 DATA-169,-170,-171,-172,-180,-173
1040 DATA-174,-175, 105, 166,-126, 0
1050 DATA 32, 0,-109, 0,-110, 0
1060 DATA-115, 0,-116, 0,-127,-160
1070 DATA-161,-162,-178, 0,-179, 0
1080 DATA 32, 0,-126, 0,-180, 0
1090 DATA 32, 0, 32, 0,-177, 0
1100 DATA-181,-182,-183, 32, 118,-105
1110 DATA 120, 116,-108, 0, 102, 0
1120 DATA 187, 188, 189, 190, 191, -96
1130 DATA 115, 116, -97, -98, -99,-100
1140 DATA 123,-101, 115, 116,-102, 124
1150 DATA 120, 116,-106,-107, 32, 102
1160 DATA-103, -96, 120, 116, -97, 188
1170 DATA -99,-104,-177, 0,-177, 0
1180 DATA-177, 0,-126, 0,-184, 0
1190 DATA-185, 0,-188, 0,-189, 0
1200 DATA-186, 0,-187, 0,-176, 178
1210 DATA-125, 32, 96, 97, 98, 99
1220 DATA 100, 101, 102, 103, 104, 101
1230 DATA 105, 106, 107, 108, 105, 109
1240 DATA 107, 110, 105, 106, 173, 174
1250 DATA 175, 176, 173, 174, 177, 32
1260 DATA 107, 111, 105, 106, 112, 113
1270 DATA 102, 103, 114, 0, 177, 0
1280 DATA 32, 114, 115, 116, 112, 117
1290 DATA 102, 103, 114, 32, 175, 176
1300 DATA 179, 180, 181, 182, 107, 110
1310 DATA 102, 103, 118, 119, 120, 106
1320 DATA 173, 178, 177, 32, 118, 110
1330 DATA 121, 122, 104, 101, 102, 103
1340 DATA 123, 124, 115, 116, 125, 126
1350 DATA 127, 160, 161, 162, 105, 106
1360 DATA 163, 164, 165, 166, 183, 184
1370 DATA 185, 186, 167, 168, 182, -103
1380 DATA 169, 170, 127, 169, 171, 172
1390 DATA 105, 109,-109, 0,-110, 0
1400 DATA-111,-112,-113,-114,-115, 0
1410 DATA-116, 0,-117,-118,-119,-120
1420 DATA-121,-122,-123,-124

```

```

10 REM ***** LOADER *****
20 PRINT"POKE44,16:POKE4096,0:NEW"
30 PRINT"LOAD"CHR$(34)"NEW CHR.:";
35 PRINTCHR$(34)",8,1"
40 PRINT"LOADPOKE53272,19:LOAD"CHR$(34);
45 PRINT"FO PRG,"CHR$(34)",8"
50 PRINT"RUN"
60 FORC=0T04:READA:POKE631+C,A:NEXTC
70 POKE198,C:END
80 DATA 19,13,13,13,13

```

```

10 REM ***** FŐ PRG. *****
20 GOSUB505
30 POKE53280,11:POKE53281,11
40 PRINT"*****KESZITETTE FOLDI ENDRE";
45 PRINT"1986.01."
50 A$(0)="000":A$(1)="000":A$(2)="000"
60 SR$="EDE SOFTWARE"86"
70 FORW=0TO2:PRINT$(W):X=4+W
75 GOSUB605:PRINT"NEXTW
80 PRINT"*****CREATED BY FOLDI ENDRE";
85 PRINT"1986.01."
90 READSR$:IFSR$<>"*"THEN95
92 FORC=0TO20000:NEXT:END
95 X=0:GOSUB605
100 PRINT:PRINT:GOTO90
205 DATA "*****HASZNALT VALTOZOK"
210 DATA "SR$:X;"
220 DATA "SR$(0):C:A:XX;"
240 DATA "*****RUTIN HIVASA"
250 DATA "SR$=SA KIIRANDO SZOVEG"
260 DATA "X=KKEZDO OSZLOPPOZICIO"
265 DATA "GOSUB 605":*
500 REM *** STRING BETOLTES ***
505 DIMSR$(255)
510 OPEN8,8,8,"16X16 CHR.,SEQ,READ"
515 FORC=32TO95:INPUT#,SR$(C)
520 NEXTC:CLOSE8:RETURN
600 REM *** SUBROUTINE ***
605 PRINTTAB(X);
610 FORC=1TOLEN(SR$(X))
615 A=ASC(MID$(SR$(X),C,1))
620 IFA<32ORA>95THENGOSUB505:GOTO640
625 XX=2+(LEN(SR$(X))=7):X=X+XX
630 IFX>40THENGOSUB645
635 PRINTSR$(X);
640 NEXTC:RETURN
645 PRINT:PRINT:X=XX:RETURN
650 PRINTCHR$(X):RETURN

```

```

10 REM ***** ALAP *****
20 GOSUB100
30 POKE53280,11:POKE53281,11
40 REM ***** SAJAT PROGRAM *****
50 SR$="*****KARAKTEREK!"
60 X=4:GOSUB150:PRINT
70 PRINT"*****MEHET?":POKE198,0:WAIT198,1
80 END
90 REM ***** STRING BETOLTES *****
100 DIMSR$(255)
110 OPEN8,8,8,"16X16 CHR.,SEQ,READ"
120 FORC=32TO95:INPUT#,SR$(C)
130 NEXTC:CLOSE8:RETURN
140 REM ***** SUBROUTINE *****
150 PRINTTAB(X);
160 FORC=1TOLEN(SR$(X))
170 A=ASC(MID$(SR$(X),C,1))
180 IFA<32ORA>95THENGOSUB240:GOTO220
190 XX=2+(LEN(SR$(X))=7):X=X+XX
200 IFX>40THENGOSUB230
210 PRINTSR$(X);
220 NEXTC:RETURN
230 PRINT:PRINT:X=XX:RETURN
240 PRINTCHR$(X):RETURN

```

felhasználásakor a következőket tartuk szem előtt. Az új készlet az 1-es számú memória 2k-4k közötti területét. Ezért betöltése előtt a BASIC terület kezdetét állítsuk át 4k-ra (ezt végzi el a LOADER program 20-as sora), majd az új karaktereket az eredeti helyükre töl-

sük vissza a LOAD "NEW CHAR",8,1 utasítással, és kapcsoljunk át ezek használatára. Ezután foglaljunk helyet az SR\$ tömbnek a DIM SR\$(255) utasítással és olvassuk be a 16*16 CHR. nevű szekvenciális fájlt a tömb 32-96 elemeibe. A felhasználáshoz ezután már csak a FŐ

PRG. nevű program 600-650-es sorai szükségesegek. A LOADER és a FŐ PRG. nevű programok a létrehozott karakterkészlet egy demo jellegű bemutatását adják. A LOADER példa lehet más betöltő programok készítésére is. Amennyiben 20-as sorában a tinta színét a papír színére változtatjuk, úgy működésének logikája a képernyő nem követhető. Az ALAP nevű, teljesen önálló felhasználást bemutató program a LOADER átírásával tölthető be.

Az új karakterkészlet használatának előnye azonkívül, hogy betűi messzebből (például egy osztályterem belől)

is jól olvashatók, az, hogy a billentyűk felső lapján lévő karakterek egy képernyőn belül állnak a rendelkezésünkre akár a normál, akár a nagyított technika típusát használjuk is. A normál karakterek egyszerű PRINT utasítással, az újak az SR\$-be való beírással és a kiíró szubrutin meghívásával írathatók ki. A kiírt rutin X változója az egy soron belül történő pozicionálásra szolgál. Az SR\$ értékadásakor az időzítőlen belül szabadon, a megszokott módon használható és működőnek a kurzorvezérlő és színbeállító karakterek.

FÖLDI ENDRE

A SZÁMÍTÓGÉP ANATÓMIÁJA

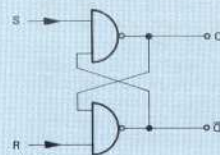
Előző cikkünkben ismertettük egy számítógép általános felépítését, valamint az adat-és címbusz működését. Most a vezérlőbuszra térünk rá.

Egy rendszer működésében alapvető az időzítés kérdése. Ez azt jelenti, hogy biztosítani kell, hogy a buszon akkor jelenjen meg az adat, illetve a cím, amikor azt a fogadóegység képes feldolgozni. Vigyázni kell ezenkívül arra is, hogy mindig a megfelelő eszköz álljon készen az információ fogadására. Tehát például ha a CPU a memóriába akar adatot írni, jeleznie kell a rendszer számára, hogy 1. valamilyen

külső egységgel akar foglalkozni; 2. konkrétan a memóriával, nem a perifériával és 3. abba írni akar (nem olvasni belőle).

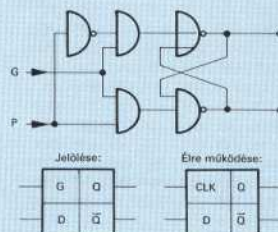
Ha memória vagy I/O rendszer írása történik, először el kell helyezni az adatot és a címet a megfelelő buszokon, és csak ezután lehet jelezni az írási szándékot. Ellenkező esetben ugyanis írás történik a címbusz tartalmának megfelelő — még nem helyes — címre.

Az, hogy a fenti három információit milyen vezérlőjelkombinációk segítségével közöljük a rendszerrel, az adott számítógéptől, leginkább az



BEMENETEK		KIMENETEK	
S	R	Q	Q̄
L	H	H	L
H	L	L	H
H	H	Q _{0,000}	Q̄ _{0,000}
L	L	H	H

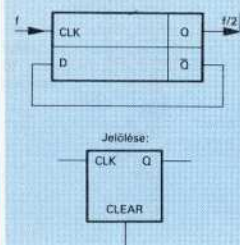
1. ábra



BEMENETEK		KIMENETEK	
G	P	Q	Q̄
H	L	L	H
H	H	H	L
L	X	Q _{0,000}	Q̄ _{0,000}

X: azt jelenti, hogy lényegtelen, milyen szint kerül a bemenetre

2. ábra



3. ábra

adott CPU-tól függ. A leggyakrabban használt kombinációk a következők.

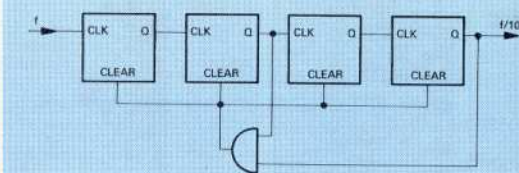
a) A jelek pontosan megegyeznek a fent leírttal. Tehát van egy információengedélyező jel (elnevezése V vagy DEN), egy IO/M jel, aminek egyik állapota azt jelenti, hogy memória-hozzáfordulás történik, a másik pedig periféria-hozzáfordulást jelöl, és egy R/W, ami az írást és az olvasást választja szét. Ilyenek például a 6809 és 8086 alapú rendszerek.

b) Négy jel van: I/O, M, R, W. Az első kettő választja ki a hozzáfordulás célját, a másik kettő az irányát. Ha a CPU külső eszközzel akar kommunikálni, a négy jel közül kettő aktív szinten van. Ilyenek általában a Z80 alapú rendszerek.

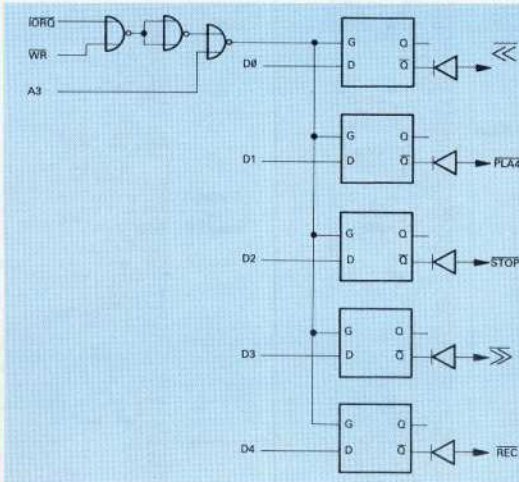
c) Itt is négy jel van: IOR, IOW, MR, MW. Az első a periféria olvasását jelenti (I/O Read = I/O olvasás), a második perifériaírást (I/O Write = I/O írás), a harmadik és negyedik memóriaolvasást, illetve -írást jelent. Az érvényességi állapotot bármelyik jel aktív szintje jelzi. Ilyenek például az I8080-as számítógépek.

A legfontosabb vezérlőjelek közé tartozik még az órajel (Clk, \emptyset). Ez határozza meg a rendszer működési sebességét, és ezt használja a CPU és néhány kiegészítő egység saját belső folyamatainak időzítéséhez. Ebből következik, hogy minden vezérlőjel változása e jel változásával együtt történik, tehát a Clk határozza meg az „időegységet” a rendszerben.

Előző írásunkban megismertünk a logikai kapukkal és a digitális áramkörök tervezésének módszereivel. Most e módszer segítségével megtervezünk a számítógép leggyakrabban használt alapáramköreit. Ezeknek az áramköröknek a megépítésére a gyakorlatban ritkán van szükség, mert egy IC-ben készen kaphatók,



4. ábra



5. ábra

de működésüket csak így érthetjük meg igazán jól.

Kezdjük a tárolókkal. Vizsgáljuk meg az 1. ábra szerinti kapcsolás működését! Érdekes dolgot fedezhetünk fel: ha mindkét bemenet logikai HI szinten van, a kimenet a bemenetek előző állapotától függ! Tehát létrehozunk egy olyan áramkört, ami „emlékezik” előző állapotára. Ezt a kapcsolást hívják RS-tárolónak, más néven RS-flip-flopnak. A név a két bemenetre, az 1-be állító Setre és a 0-ba állító Resetre utal (lásd az igazságtáblázatot).

Ehhez hasonló elemekből épül fel a RAM áramkörök egyik fajtája, a statikus RAM. Ha az RS-flip-flopot nem NAND, hanem NOR kapukból építjük fel, a bemenetek aktív szintje HI lesz.

Nézzük meg mi történik, ha mindkét bemenet aktív? Ebben az esetben mind a Q, mind a Q-bar kimenetek értéke (NAND kapuk esetén) HI lesz. Ha pedig ezután mind a két bemenetet inaktívvá tesszük, a flip-flop véletlenszerű állapotba kerül. Ezért ezt az állapotot tiltott állapotoknak nevezzük.

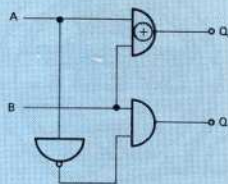
Az RS-flip-flop egyik hátrá-

nya, hogy külön bemenete van az 1 és külön a 0 számára. Ez kiküszöbölhető a D-flip-flop alkalmazásával. Ennek is két bemenete van: az „adat” és az „adat érvényes” vonalak. Kapcsolási rajza és igazságtáblája a 2. ábrán látható.

A gyakrabban használt D-flip-flopok nem a vezérlőjel aktív szintje esetén, hanem annak le- vagy felfutó éle hatására tárolják el az adatot. Ennek a változatnak a kapcsolási rajza már bonyolultabb. A legtöbb ilyen típus tartalmaz egy R és S bemenetet is, amivel mint RS-flip-flop használható. Ezeket a vonalakat gyakran hívják clearnek, illetve preset-nek.

A flip-flopok segítségével sok hasznos áramkört hozhatunk létre. Ilyen például a frekvenciaosztó, melynek feladata a bemenetre kapcsolt rezgés formájú jel frekvenciájának elosztása egy egész számmal.

Minden frekvenciaosztás megvalósítható 2-vel osztó egységekből. Ennek rajza a 3. ábrán látható. Házi feladatként az eddigiek alapján bizonyítsuk be a kapcsolás működését! Ha például 4-gyel akarunk osztani, két egymás utáni



A	B	Q ₁	Q ₂
0	0	0	0
0	1	1	1
1	0	0	1
1	1	0	0

$Q_1 = A_{XOR} B$ $Q_2 = A_{AND} B$

Az előző rész házi feladatának megoldása

2-es osztóra van szükség. Általában 2^n -nel való osztás n db kettes oszttal oldható meg.

De mit tegyünk, ha 10-zel akarunk osztani?

Készítsünk egy 16-os oszttól olyan D-flip-flopokból, amelyek rendelkeznek clear bemenettel, és egy logikai áramkörrel figyeltessük a flip-flopok kimeneteit. Amikor a kimenetekből összeállítható bináris szám eléri a 1010-et, az áramkör adjon minden clear bemenetre egy impulzust! Ez a berendezés minden tizedik bemenő impulzus hatására ad egy kimenő impulzust (4. ábra). Most ne csak az utolsó flip-flop kimenetét vezessük ki, hanem az összesét! Ezzel létrehozhatjuk a SZÁMLÁLÓT. Ez megszámlálja a bemenő impulzusokat, és a kimenetein bináris formában jelzi a pillanatnyi értéket. Ha elér egy bizonyos maximumot, 0-tól kezdő újra.

A D-flip-flop alkalmazására mutatjuk be példaként az 5. ábra szerinti magnóvezérlő kapcsolást. Ez elektronikus vezérlésű magnókat tud bármilyen üzemmódba kapcsolni. Ilyen például az AKAI cég minden CSF-fel kezdődő típusszámú magnója. Mivel a CSF-ekben CMOS logika van (lásd előző rész), ami 8 V-os tápfeszültséggel működik, illesztőelemeként diódát kell alkalmaznunk. Erről majd később lesz szó. Az áramkör ilyen formában Z80-as számítógéphez illeszthető. Működése: a NOR kapuk akkor juttatnak HI szintet a D tárolók G bemenetére, ha iorq és WR aktív és A3 LO szintű. Ekkor a flip-flop az adatbuszon lévő, neki szóló adatot tárolja és a kimenetére juttatja.

MATEMATIKA

Ez a program főleg házi feladatok megoldására szolgál. Található benne kalkulátor-, függvényábrázoló és szövegszerkesztő program.

```

10 GO TO 200
15 BORDER 6 PAPER 6 INK 0 B
16 LIGHT 0 OVER 0 INVERSE 0 FLAS
17 CLR
18 CLEAR 55300
19 PLOT 1 200 100
20 TITLE 1 200 100
21 DRN 20 120 80
22 PRINT AT 7.13."Naguito"
23 PRINT AT 15.9."By Papp Zolt"
24
25 PRINT AT 17.13."0 1985"
26 PRINT AT 0.0."7-8 Bal"
27
28 .Lobb Enter-Naguit
29 Break-Basic R-Print
30 P-Print-Restart"
31
32 PRINT AT 15.10,
33 LOAD
34 RANDOINIZE USR 55693
35 PAUSE 120
36 RANDOINIZE USR 55693
37 LOAD
38 RANDOINIZE USR 55322
39
40 NOP
41 CALL #C05D
42 #E43000D LD BC,#0000
43 #E43000D LD BC,#0001,BC
44 CALL #C05D
45 #E43000D LD BC,#0001,BC
46 CALL #C05D
47 LD HL,#C00C
48 PUSH HL
49 #E43000D LD BC,#0001,BC
50 #E43000D LD BC,#0001,BC
51 #E43000D LD BC,#0001,BC
52 #E43000D LD BC,#0001,BC
53 #E43000D LD BC,#0001,BC
54 #E43000D LD BC,#0001,BC
55 #E43000D LD BC,#0001,BC
56 #E43000D LD BC,#0001,BC
57 #E43000D LD BC,#0001,BC
58 #E43000D LD BC,#0001,BC
59 #E43000D LD BC,#0001,BC
60 #E43000D LD BC,#0001,BC
61 #E43000D LD BC,#0001,BC
62 #E43000D LD BC,#0001,BC
63 #E43000D LD BC,#0001,BC
64 #E43000D LD BC,#0001,BC
65 #E43000D LD BC,#0001,BC
66 #E43000D LD BC,#0001,BC
67 #E43000D LD BC,#0001,BC
68 #E43000D LD BC,#0001,BC
69 #E43000D LD BC,#0001,BC
70 #E43000D LD BC,#0001,BC
71 #E43000D LD BC,#0001,BC
72 #E43000D LD BC,#0001,BC
73 #E43000D LD BC,#0001,BC
74 #E43000D LD BC,#0001,BC
75 #E43000D LD BC,#0001,BC
76 #E43000D LD BC,#0001,BC
77 #E43000D LD BC,#0001,BC
78 #E43000D LD BC,#0001,BC
79 #E43000D LD BC,#0001,BC
80 #E43000D LD BC,#0001,BC
81 #E43000D LD BC,#0001,BC
82 #E43000D LD BC,#0001,BC
83 #E43000D LD BC,#0001,BC
84 #E43000D LD BC,#0001,BC
85 #E43000D LD BC,#0001,BC
86 #E43000D LD BC,#0001,BC
87 #E43000D LD BC,#0001,BC
88 #E43000D LD BC,#0001,BC
89 #E43000D LD BC,#0001,BC
90 #E43000D LD BC,#0001,BC
91 #E43000D LD BC,#0001,BC
92 #E43000D LD BC,#0001,BC
93 #E43000D LD BC,#0001,BC
94 #E43000D LD BC,#0001,BC
95 #E43000D LD BC,#0001,BC
96 #E43000D LD BC,#0001,BC
97 #E43000D LD BC,#0001,BC
98 #E43000D LD BC,#0001,BC
99 #E43000D LD BC,#0001,BC
100 #E43000D LD BC,#0001,BC
101 #E43000D LD BC,#0001,BC
102 #E43000D LD BC,#0001,BC
103 #E43000D LD BC,#0001,BC
104 #E43000D LD BC,#0001,BC
105 #E43000D LD BC,#0001,BC
106 #E43000D LD BC,#0001,BC
107 #E43000D LD BC,#0001,BC
108 #E43000D LD BC,#0001,BC
109 #E43000D LD BC,#0001,BC
110 #E43000D LD BC,#0001,BC
111 #E43000D LD BC,#0001,BC
112 #E43000D LD BC,#0001,BC
113 #E43000D LD BC,#0001,BC
114 #E43000D LD BC,#0001,BC
115 #E43000D LD BC,#0001,BC
116 #E43000D LD BC,#0001,BC
117 #E43000D LD BC,#0001,BC
118 #E43000D LD BC,#0001,BC
119 #E43000D LD BC,#0001,BC
120 #E43000D LD BC,#0001,BC
121 #E43000D LD BC,#0001,BC
122 #E43000D LD BC,#0001,BC
123 #E43000D LD BC,#0001,BC
124 #E43000D LD BC,#0001,BC
125 #E43000D LD BC,#0001,BC
126 #E43000D LD BC,#0001,BC
127 #E43000D LD BC,#0001,BC
128 #E43000D LD BC,#0001,BC
129 #E43000D LD BC,#0001,BC
130 #E43000D LD BC,#0001,BC
131 #E43000D LD BC,#0001,BC
132 #E43000D LD BC,#0001,BC
133 #E43000D LD BC,#0001,BC
134 #E43000D LD BC,#0001,BC
135 #E43000D LD BC,#0001,BC
136 #E43000D LD BC,#0001,BC
137 #E43000D LD BC,#0001,BC
138 #E43000D LD BC,#0001,BC
139 #E43000D LD BC,#0001,BC
140 #E43000D LD BC,#0001,BC
141 #E43000D LD BC,#0001,BC
142 #E43000D LD BC,#0001,BC
143 #E43000D LD BC,#0001,BC
144 #E43000D LD BC,#0001,BC
145 #E43000D LD BC,#0001,BC
146 #E43000D LD BC,#0001,BC
147 #E43000D LD BC,#0001,BC
148 #E43000D LD BC,#0001,BC
149 #E43000D LD BC,#0001,BC
150 #E43000D LD BC,#0001,BC
151 #E43000D LD BC,#0001,BC
152 #E43000D LD BC,#0001,BC
153 #E43000D LD BC,#0001,BC
154 #E43000D LD BC,#0001,BC
155 #E43000D LD BC,#0001,BC
156 #E43000D LD BC,#0001,BC
157 #E43000D LD BC,#0001,BC
158 #E43000D LD BC,#0001,BC
159 #E43000D LD BC,#0001,BC
160 #E43000D LD BC,#0001,BC
161 #E43000D LD BC,#0001,BC
162 #E43000D LD BC,#0001,BC
163 #E43000D LD BC,#0001,BC
164 #E43000D LD BC,#0001,BC
165 #E43000D LD BC,#0001,BC
166 #E43000D LD BC,#0001,BC
167 #E43000D LD BC,#0001,BC
168 #E43000D LD BC,#0001,BC
169 #E43000D LD BC,#0001,BC
170 #E43000D LD BC,#0001,BC
171 #E43000D LD BC,#0001,BC
172 #E43000D LD BC,#0001,BC
173 #E43000D LD BC,#0001,BC
174 #E43000D LD BC,#0001,BC
175 #E43000D LD BC,#0001,BC
176 #E43000D LD BC,#0001,BC
177 #E43000D LD BC,#0001,BC
178 #E43000D LD BC,#0001,BC
179 #E43000D LD BC,#0001,BC
180 #E43000D LD BC,#0001,BC
181 #E43000D LD BC,#0001,BC
182 #E43000D LD BC,#0001,BC
183 #E43000D LD BC,#0001,BC
184 #E43000D LD BC,#0001,BC
185 #E43000D LD BC,#0001,BC
186 #E43000D LD BC,#0001,BC
187 #E43000D LD BC,#0001,BC
188 #E43000D LD BC,#0001,BC
189 #E43000D LD BC,#0001,BC
190 #E43000D LD BC,#0001,BC
191 #E43000D LD BC,#0001,BC
192 #E43000D LD BC,#0001,BC
193 #E43000D LD BC,#0001,BC
194 #E43000D LD BC,#0001,BC
195 #E43000D LD BC,#0001,BC
196 #E43000D LD BC,#0001,BC
197 #E43000D LD BC,#0001,BC
198 #E43000D LD BC,#0001,BC
199 #E43000D LD BC,#0001,BC
200 #E43000D LD BC,#0001,BC

```

```

0000 LD BC,#00
0001 DEC C
0002 LD BC,#00
0003 DEC C
0004 LD BC,#00
0005 DEC C
0006 LD BC,#00
0007 DEC C
0008 LD BC,#00
0009 DEC C
000A LD BC,#00
000B DEC C
000C LD BC,#00
000D DEC C
000E LD BC,#00
000F DEC C
0010 LD BC,#00
0011 DEC C
0012 LD BC,#00
0013 DEC C
0014 LD BC,#00
0015 DEC C
0016 LD BC,#00
0017 DEC C
0018 LD BC,#00
0019 DEC C
001A LD BC,#00
001B DEC C
001C LD BC,#00
001D DEC C
001E LD BC,#00
001F DEC C
0020 LD BC,#00
0021 DEC C
0022 LD BC,#00
0023 DEC C
0024 LD BC,#00
0025 DEC C
0026 LD BC,#00
0027 DEC C
0028 LD BC,#00
0029 DEC C
002A LD BC,#00
002B DEC C
002C LD BC,#00
002D DEC C
002E LD BC,#00
002F DEC C
0030 LD BC,#00
0031 DEC C
0032 LD BC,#00
0033 DEC C
0034 LD BC,#00
0035 DEC C
0036 LD BC,#00
0037 DEC C
0038 LD BC,#00
0039 DEC C
003A LD BC,#00
003B DEC C
003C LD BC,#00
003D DEC C
003E LD BC,#00
003F DEC C
0040 LD BC,#00
0041 DEC C
0042 LD BC,#00
0043 DEC C
0044 LD BC,#00
0045 DEC C
0046 LD BC,#00
0047 DEC C
0048 LD BC,#00
0049 DEC C
004A LD BC,#00
004B DEC C
004C LD BC,#00
004D DEC C
004E LD BC,#00
004F DEC C
0050 LD BC,#00
0051 DEC C
0052 LD BC,#00
0053 DEC C
0054 LD BC,#00
0055 DEC C
0056 LD BC,#00
0057 DEC C
0058 LD BC,#00
0059 DEC C
005A LD BC,#00
005B DEC C
005C LD BC,#00
005D DEC C
005E LD BC,#00
005F DEC C
0060 LD BC,#00
0061 DEC C
0062 LD BC,#00
0063 DEC C
0064 LD BC,#00
0065 DEC C
0066 LD BC,#00
0067 DEC C
0068 LD BC,#00
0069 DEC C
006A LD BC,#00
006B DEC C
006C LD BC,#00
006D DEC C
006E LD BC,#00
006F DEC C
0070 LD BC,#00
0071 DEC C
0072 LD BC,#00
0073 DEC C
0074 LD BC,#00
0075 DEC C
0076 LD BC,#00
0077 DEC C
0078 LD BC,#00
0079 DEC C
007A LD BC,#00
007B DEC C
007C LD BC,#00
007D DEC C
007E LD BC,#00
007F DEC C
0080 LD BC,#00
0081 DEC C
0082 LD BC,#00
0083 DEC C
0084 LD BC,#00
0085 DEC C
0086 LD BC,#00
0087 DEC C
0088 LD BC,#00
0089 DEC C
008A LD BC,#00
008B DEC C
008C LD BC,#00
008D DEC C
008E LD BC,#00
008F DEC C
0090 LD BC,#00
0091 DEC C
0092 LD BC,#00
0093 DEC C
0094 LD BC,#00
0095 DEC C
0096 LD BC,#00
0097 DEC C
0098 LD BC,#00
0099 DEC C
009A LD BC,#00
009B DEC C
009C LD BC,#00
009D DEC C
009E LD BC,#00
009F DEC C
00A0 LD BC,#00
00A1 DEC C
00A2 LD BC,#00
00A3 DEC C
00A4 LD BC,#00
00A5 DEC C
00A6 LD BC,#00
00A7 DEC C
00A8 LD BC,#00
00A9 DEC C
00AA LD BC,#00
00AB DEC C
00AC LD BC,#00
00AD DEC C
00AE LD BC,#00
00AF DEC C
00B0 LD BC,#00
00B1 DEC C
00B2 LD BC,#00
00B3 DEC C
00B4 LD BC,#00
00B5 DEC C
00B6 LD BC,#00
00B7 DEC C
00B8 LD BC,#00
00B9 DEC C
00BA LD BC,#00
00BB DEC C
00BC LD BC,#00
00BD DEC C
00BE LD BC,#00
00BF DEC C
00C0 LD BC,#00
00C1 DEC C
00C2 LD BC,#00
00C3 DEC C
00C4 LD BC,#00
00C5 DEC C
00C6 LD BC,#00
00C7 DEC C
00C8 LD BC,#00
00C9 DEC C
00CA LD BC,#00
00CB DEC C
00CC LD BC,#00
00CD DEC C
00CE LD BC,#00
00CF DEC C
00D0 LD BC,#00
00D1 DEC C
00D2 LD BC,#00
00D3 DEC C
00D4 LD BC,#00
00D5 DEC C
00D6 LD BC,#00
00D7 DEC C
00D8 LD BC,#00
00D9 DEC C
00DA LD BC,#00
00DB DEC C
00DC LD BC,#00
00DD DEC C
00DE LD BC,#00
00DF DEC C
00E0 LD BC,#00
00E1 DEC C
00E2 LD BC,#00
00E3 DEC C
00E4 LD BC,#00
00E5 DEC C
00E6 LD BC,#00
00E7 DEC C
00E8 LD BC,#00
00E9 DEC C
00EA LD BC,#00
00EB DEC C
00EC LD BC,#00
00ED DEC C
00EE LD BC,#00
00EF DEC C
00F0 LD BC,#00
00F1 DEC C
00F2 LD BC,#00
00F3 DEC C
00F4 LD BC,#00
00F5 DEC C
00F6 LD BC,#00
00F7 DEC C
00F8 LD BC,#00
00F9 DEC C
00FA LD BC,#00
00FB DEC C
00FC LD BC,#00
00FD DEC C
00FE LD BC,#00
00FF DEC C

```

```

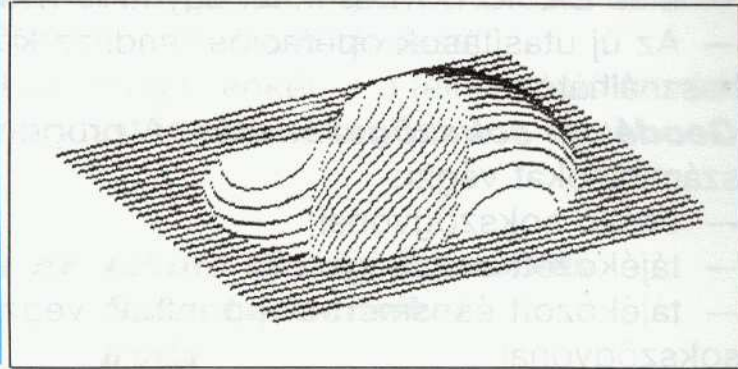
0000 LD BC,#00
0001 DEC C
0002 LD BC,#00
0003 DEC C
0004 LD BC,#00
0005 DEC C
0006 LD BC,#00
0007 DEC C
0008 LD BC,#00
0009 DEC C
000A LD BC,#00
000B DEC C
000C LD BC,#00
000D DEC C
000E LD BC,#00
000F DEC C
0010 LD BC,#00
0011 DEC C
0012 LD BC,#00
0013 DEC C
0014 LD BC,#00
0015 DEC C
0016 LD BC,#00
0017 DEC C
0018 LD BC,#00
0019 DEC C
001A LD BC,#00
001B DEC C
001C LD BC,#00
001D DEC C
001E LD BC,#00
001F DEC C
0020 LD BC,#00
0021 DEC C
0022 LD BC,#00
0023 DEC C
0024 LD BC,#00
0025 DEC C
0026 LD BC,#00
0027 DEC C
0028 LD BC,#00
0029 DEC C
002A LD BC,#00
002B DEC C
002C LD BC,#00
002D DEC C
002E LD BC,#00
002F DEC C
0030 LD BC,#00
0031 DEC C
0032 LD BC,#00
0033 DEC C
0034 LD BC,#00
0035 DEC C
0036 LD BC,#00
0037 DEC C
0038 LD BC,#00
0039 DEC C
003A LD BC,#00
003B DEC C
003C LD BC,#00
003D DEC C
003E LD BC,#00
003F DEC C
0040 LD BC,#00
0041 DEC C
0042 LD BC,#00
0043 DEC C
0044 LD BC,#00
0045 DEC C
0046 LD BC,#00
0047 DEC C
0048 LD BC,#00
0049 DEC C
004A LD BC,#00
004B DEC C
004C LD BC,#00
004D DEC C
004E LD BC,#00
004F DEC C
0050 LD BC,#00
0051 DEC C
0052 LD BC,#00
0053 DEC C
0054 LD BC,#00
0055 DEC C
0056 LD BC,#00
0057 DEC C
0058 LD BC,#00
0059 DEC C
005A LD BC,#00
005B DEC C
005C LD BC,#00
005D DEC C
005E LD BC,#00
005F DEC C
0060 LD BC,#00
0061 DEC C
0062 LD BC,#00
0063 DEC C
0064 LD BC,#00
0065 DEC C
0066 LD BC,#00
0067 DEC C
0068 LD BC,#00
0069 DEC C
006A LD BC,#00
006B DEC C
006C LD BC,#00
006D DEC C
006E LD BC,#00
006F DEC C
0070 LD BC,#00
0071 DEC C
0072 LD BC,#00
0073 DEC C
0074 LD BC,#00
0075 DEC C
0076 LD BC,#00
0077 DEC C
0078 LD BC,#00
0079 DEC C
007A LD BC,#00
007B DEC C
007C LD BC,#00
007D DEC C
007E LD BC,#00
007F DEC C
0080 LD BC,#00
0081 DEC C
0082 LD BC,#00
0083 DEC C
0084 LD BC,#00
0085 DEC C
0086 LD BC,#00
0087 DEC C
0088 LD BC,#00
0089 DEC C
008A LD BC,#00
008B DEC C
008C LD BC,#00
008D DEC C
008E LD BC,#00
008F DEC C
0090 LD BC,#00
0091 DEC C
0092 LD BC,#00
0093 DEC C
0094 LD BC,#00
0095 DEC C
0096 LD BC,#00
0097 DEC C
0098 LD BC,#00
0099 DEC C
009A LD BC,#00
009B DEC C
009C LD BC,#00
009D DEC C
009E LD BC,#00
009F DEC C
00A0 LD BC,#00
00A1 DEC C
00A2 LD BC,#00
00A3 DEC C
00A4 LD BC,#00
00A5 DEC C
00A6 LD BC,#00
00A7 DEC C
00A8 LD BC,#00
00A9 DEC C
00AA LD BC,#00
00AB DEC C
00AC LD BC,#00
00AD DEC C
00AE LD BC,#00
00AF DEC C
00B0 LD BC,#00
00B1 DEC C
00B2 LD BC,#00
00B3 DEC C
00B4 LD BC,#00
00B5 DEC C
00B6 LD BC,#00
00B7 DEC C
00B8 LD BC,#00
00B9 DEC C
00BA LD BC,#00
00BB DEC C
00BC LD BC,#00
00BD DEC C
00BE LD BC,#00
00BF DEC C
00C0 LD BC,#00
00C1 DEC C
00C2 LD BC,#00
00C3 DEC C
00C4 LD BC,#00
00C5 DEC C
00C6 LD BC,#00
00C7 DEC C
00C8 LD BC,#00
00C9 DEC C
00CA LD BC,#00
00CB DEC C
00CC LD BC,#00
00CD DEC C
00CE LD BC,#00
00CF DEC C
00D0 LD BC,#00
00D1 DEC C
00D2 LD BC,#00
00D3 DEC C
00D4 LD BC,#00
00D5 DEC C
00D6 LD BC,#00
00D7 DEC C
00D8 LD BC,#00
00D9 DEC C
00DA LD BC,#00
00DB DEC C
00DC LD BC,#00
00DD DEC C
00DE LD BC,#00
00DF DEC C
00E0 LD BC,#00
00E1 DEC C
00E2 LD BC,#00
00E3 DEC C
00E4 LD BC,#00
00E5 DEC C
00E6 LD BC,#00
00E7 DEC C
00E8 LD BC,#00
00E9 DEC C
00EA LD BC,#00
00EB DEC C
00EC LD BC,#00
00ED DEC C
00EE LD BC,#00
00EF DEC C
00F0 LD BC,#00
00F1 DEC C
00F2 LD BC,#00
00F3 DEC C
00F4 LD BC,#00
00F5 DEC C
00F6 LD BC,#00
00F7 DEC C
00F8 LD BC,#00
00F9 DEC C
00FA LD BC,#00
00FB DEC C
00FC LD BC,#00
00FD DEC C
00FE LD BC,#00
00FF DEC C

```

A kalkulátor a BASIC függvényeinek és hat saját függvénynek a használatával testzőleges aritmetikai kifejezés kiszámítását teszi lehetővé. Függvényábrázolókor magunk választotta koordináta-rendszerben ábrázolhatunk akár több függvényt is egyszerre. A program printerre is dolgozik. A szövegszerkesztő lehetővé teszi az ékezetes betűk használatát. Ehhez a táblázat adatait kell begépelni.

A programot a 9900-as sorral lehet megőrizni.

PAPP ZOLTÁN



Commodore 64-üzemeltetőknek ajánljuk az alábbi programtermékeinket

DBASIC VI.O A program megkönnyíti a programozási munkát. A definiált új utasítások a standard BASIC-ben megoldhatatlan vagy több utasítással megoldható funkciókat végeznek.

— Lerövidíti a futási időt.

A DBASIC utasítások sebessége kb. 100-szoros a megfelelő standard BASIC utasításcsoportéhoz képest.

— Megnöveli a diskette kapacitását.

A numerikus mezők pakoltan tárolhatók (2 szám és egy bájtban).

— Nem csökkenti a BASIC területet.

Más bővítésektől eltérően nem a standard BASIC területen működik.

— Megnöveli a tár kapacitását.

Lehetővé teszi a standard BASIC által semmire sem használt 16 kb-át memória használatát.

— Más BASIC bővítésekkel együtt is használható.

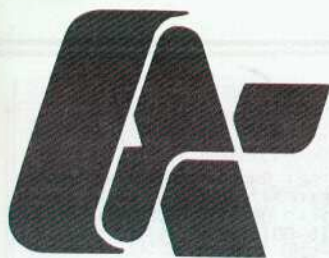
— Az új utasítások operációs rendszerként is használhatók.

Geodéziai sokszögszámítás. A program a következő számításokat végzi:

— önálló sokszögvonala

— tájékozott sokszögvonala

— tájékozott és ismert alappontban végződő sokszögvonala



**Alkotó Ifjúság Egyesülés
Számítástechnikai Iroda
Budapest V., Garibaldi u. 2.
Levél cím: 1519 Budapest, Pf. 330
Telex: 22-7272
Telefon: 112-666, 113-608**

- kettősen tájékozott sokszög vonal
- beillesztett sokszög vonal
- zárt sokszög vonal

Síkbeli keretszámítás. A program egyenes tengelyű prizmatikus rudak igénybevételének számítását végzi, a rúdvégi elmozdulások és a rúdon működő terhek figyelembevételével. A számítás a „kis elmozdulások” módszerét alkalmazza, bármilyen anyagú és geometriájú keretszerkezet méretezhető a segítségével.

Szén-, olaj-, gázkészülékek hatásfokának számítása

Légcsatornák áramlástechnikai méretezése

Porleválasztók kiválasztása

kibocsátásszámítás alapján

Fűtőtest-kiválasztás, hőveszteség-számítás

„CROSS” statikai keretszámító program

SOFTKEEPER. Lehetővé teszi a C—64-en futó programok, illetve az általuk használt adatok tetszőleges kulcs szerinti átkódolását és titkosított formában történő tárolását.

A kódoló utasításon kívül még tizenkét — a felhasználó munkáját megkönnyítő — utasítást ismer.

**Forgalomba hozza az Alkotó Ifjúság Egyesülés
Számítástechnikai
Iroda**

C16-ra

MEMORY

A „RÜN” begépelése után 4 részre osztott, számozott mező látható a képernyőn. Egy billentyű leütésére indul a játék. A gép véletlenszerűen kitalál egy számot, az annak megfelelő mezőt betárazza (színes tévén eltérő színnel), és ad hozzá egy hangot. Ilyenkor az „EN” felirat látható a képernyő felső részén, majd ez a felirat átváltozik „TE”-re. Most a játékosnak vissza kell játszania az előbb látottakat, mégpedig a megfelelő 1, 2, 3, 4 gomb használatával. Ha nem jól választott, a gép ezt jelzi. Ha sikerült, akkor a gép pittyen egyet, és ismét az „EN” felirat jelenik meg. A gép visszajátssza az előző mezőt és hangot, majd kitalál egy újabb kombinációt. Most ismét a „TE” felirat következik, és a játékosnak már két mezőt és két hangot kell megjegyeznie és helyesen visszajátszania. A játék 10 mezőig, hangig folytatódik a fentiek szerint, ha közben nem hibázik a játékos.

Ezután következik a második pálya, szintén gombnyomásra. A játék lefolyása megegyezik az előzővel, csak annyi az eltérés, hogy a gép 6 mező közül választ véletlenszerűen. A felhasználandó gombok értelemszerűen 1, 2, 3, 4, 5, 6. A játék ismét 10 mezőig és hangig megy hibátlan játék esetén, majd következik a harmadik pálya.

A játék teljesen azonos az előzővel, csak nem 10 mezőig és hangig, hanem addig folytatható, amíg a játékos nem hibázik. Hiba esetén a gép közli a játékosal, hogy hányadik lépésben tévedett.

Sajnos a printer nem ismert minden grafikus jelet, ezért a programhoz melléklet is készült.

PARRAGH ZSOLT—KECSKÉS KORNÉL

```

1 PRINT"J"
2 PRINT
3 DIMB(1000)
10 A$=""
20 B$=""
30 C$="A"
40 D$=""
60 PRINT A$:PRINTB$:PRINTC$:
PRINTB$:PRINTB$:PRINTD$
70 CHAR1,3,5,"1":CHAR1,9,5,"2":
CHAR1,3,11,"3":CHAR1,9,11,"4"
80 CHAR1,6,18,"ISMETELD MEG AMIT EN
JATSZOM!"
81 CHAR1,6,20,"MINDIG EGYEL TOBB HANGOT
KAPSZ."
82 CHAR1,3,22,"10 HANGIG MEGY./HA ELOLVY
STAD NYOMJ MEG EGY BILLENTYUT!/"
85 GETX$:IF X$=""THEN 85
86 PRINT"J"
88 PRINT
90 PRINTA$:PRINTB$:PRINTB$:PRINTC$:
PRINTB$:PRINTB$:PRINTD$
95 PRINT"■SEN"
110 VOL8
120 FOR A=1 TO10
130 B(A)=INT(RND(1)*4)+1
140 NEXTA
150 FOR K=1 TO 10
160 FOR D=1TO K
170 COLOR1,B(D)*4
174 B=(B(D)=2ORB(D)=4)*-6+1
175 C=(B(D)=3ORB(D)=4)*-6+3
180 SOUND1,B(D)*100,30
185 GOSUB1100
190 FORT=0 TO 200:NEXTT

```

```

200 NEXTD
210 FORF=1 TO K
215 PRINT"■STE"
217 FORQW=0 TO 5:GET QW$(QW):NEXT QW
220 GETE$(F):IF E$(F)=""THEN220
221 IF ASC(E$(F))<49 OR ASC(E$(F))>52
THEN SOUND3,700,15:GOTO220
225 M=ASC(E$(F))-48
226 B=(M=2 OR M=4)*-6+1
227 C=(M=3 OR M=4)*-6+3:COLOR1,M*4
228 SOUND1,(ASC(E$(F))-48)*100,30
229 GOSUB 1100
230 IF M=B(F)THEN NEXT F:FOR T=0 TO 400:
NEXT T:SOUND1,999,5:PRINT"■SEN":NEXT K:
GOTO4000
240 CHAR1,14,17,"■ELRONTOTTAD!"
245 SOUND3,650,260:FORGH=1 TO 2000:NEXT
GH
251 PRINT"J":PRINT:GOTO10
1100 FORW=B TO B+4
1200 FOR Q=C TO C+4
1300 CHAR1,W,Q,"■ "
1400 NEXT Q
1500 NEXT W
1600 FOR T=0 TO 300:NEXT T
1700 FOR W=B TO B+4
1800 FOR Q=C TO C+4
1900 CHAR1,W,Q,"■ "
2000 NEXT Q
2100 NEXT W
3000 RETURN
4000 PRINT"J":FOR S=1 TO 16
4010 FOR Z=1 TO 7
4020 COLOR4,S,Z
4030 COLOR0,S,8-Z
4040 NEXT Z
4050 NEXT S
4060 CHAR1,12,12,"■O.K. GRATULALOK!"
5000 COLOR0,2
5010 GET A$:IF A$=""THEN5010
5020 PRINT"J":PRINT
6000 PRINT"J":PRINT
6010 CLR
6020 A$=""
6030 B$=""
6040 C$="A"
6050 D$=""
6060 PRINT A$:PRINTB$:PRINTB$:PRINTC$:
PRINTB$:PRINTB$:PRINTD$
5070 CHAR1,3,5,"1":CHAR1,9,5,"2":CHAR1,1
5,5,"3":CHAR1,3,11,"4":CHAR1,9,11,"5"
6080 CHAR1,15,11,"6":CHAR1,13,16,"MOST 6
SZIN VAN!"
6090 GETX$:IF X$=""THEN 6090
6100 PRINT"J":PRINT
6110 PRINTA$:PRINTB$:PRINTB$:PRINTC$:
PRINTB$:PRINTB$:PRINTD$
6111 PRINT"■SEN"
6115 IF VE=1 THEN 8000
6120 DIM B(1000)
6130 VOL8
6140 FOR A=1 TO 10
6150 B(A)=INT(RND(1)*6)+1
6160 NEXTA
6170 FOR K=1 TO 10
6180 FOR D=1 TO K
6190 COLOR 1,B(D)*2,5
6200 B=(B(D)=2 OR B(D)=5 OR B(D)=3 OR
B(D)=6)*-6+(B(D)=3 OR B(D)=6)*-6+1

```


Melléklet a programlistához

```

6210 C=(B(D)=4 OR B(D)=5 OR B(D)=6)*-6+3
6220 SOUND1,B(D)*100,30
6230 GOSUB 1100
6240 FOR T=0 TO 200:NEXT T
6250 NEXT D
6260 FOR F=1 TO K
6265 FOR QW=0 TO 5:GETQW$(QW):NEXT QW
6267 PRINT"*****"
6270 GET E$(F):IF E$(F)="" THEN 6270
6280 IF ASC(E$(F))<49 OR ASC(E$(F))>54
    THEN SOUND 3,700,15:GOTO 6270
6290 M=ASC(E$(F))-48
6300 B=(M=2 OR M=3 OR M=5 OR M=6)*-6+
    (M=3 OR M=6)*-6+1
6310 C=(M=4 OR M=5 OR M=6)*-6+3
6315 COLOR1,M*2,5
6320 SOUND 1,M*100,30
6330 GOSUB 1100
6320 SOUND 1,M*100,30
6330 GOSUB 1100
6340 IF M=B(F) THEN NEXT F:FOR T=0 TO
    400:NEXT T:SOUND1,999,6:PRINT"*****":NEXT
    K:GOTO7000
6350 CHR1,23,14,"*****ELRONTOTTAD!"
6360 SOUND3,650,260:FOR GH=1 TO 2000:
    NEXT GH
6370 PRINT"*****":PRINT:GOTO6010
7000 PRINT"*****"
7010 FOR S=1 TO 16
7020 FOR Z=1 TO 7
7030 COLOR4,S,Z
7040 COLOR0,S,8-Z
7050 NEXT Z:NEXT S
7060 CHR1,12,12,"*****O.K. GRATULALOK!"
7070 COLOR0,2
7080 GETX$:IF X$="" THEN 7080
7090 VE=1:PRINT"*****":PRINT:GOTO 6020
8000 VOLS
8010 TY=TY+1:B(TY)=INT(RND(1)*6)+1
8020 FORK=1 TO TY
8030 COLOR1,B(K)*2,5
8040 B=(B(K)=2 OR B(K)=5 OR B(K)=3 OR
    B(K)=6)*-6+(B(K)=3 OR B(K)=6)*-6+1
8050 C=(B(K)=4 OR B(K)=5 OR B(K)=6)*-6+3
8060 SOUND1,B(K)*100,30
8070 GOSUB1100
8080 FOR T=0 TO 200:NEXT T
8090 NEXT K
8100 FOR F=1 TO TY
8105 FOR QW=0 TO 5:GET QW$(QW):NEXT QW
8107 PRINT"*****"
8110 GET E$(F):IF E$(F)="" THEN 8110
8120 IF ASC(E$(F))<49 OR ASC(E$(F))>54
    THEN SOUND3,700,15:GOTO8110
8130 M=ASC(E$(F))-48
8140 B=(M=2 OR M=3 OR M=5 OR M=6)*-6+
    (M=3 OR M=6)*-6+1
8150 C=(M=4 OR M=5 OR M=6)*-6+3
8160 COLOR1,M*2,5
8170 SOUND1,M*100,30
8180 GOSUB 1100
8190 IF M=B(F) THEN NEXT F:FOR T=0 TO
    400:NEXTT:SOUND1,999,5:PRINT"*****":
    GOTO 8010
8200 CHR1,5,17,"*****ELRONTOTTAD!"
8210 PRINT" A ";TY;". LEPESEN"
8220 FOR GH=1 TO 1000:NEXT GH
8230 PRINT"***** AKARSD MEG JATSZANI? (I/N)"
8240 INPUT JA$:IF JA$="I" THEN CLR:GOTO1
8250 IF JA$="N" THEN PRINT"*****":CHR1,15,
    12,"*****VIZONT LATASRAI!":END
8260 PRINT"NE MARHASKODJ!":FOR T=1 TO
    500:NEXT T:GOTO 8230

```

1 konka=1 karakter

```

1 A Z = " | "
2 B X = " | "
3 C Y = " | "
4 D Z = " | "

```

Ca /ez a COMMANDORE billentyű/ X=C,T,R,L+9

↑ = Ca + Q — = SCHIFT + Q
↓ = Ca + W | = SCHIFT + B
└ = Ca + R + = SCHIFT + plusz jel
┌ = Ca + S ▣ = SCHIFT + CLEAR/HOME
┐ = Ca + A ▣ = CTRL + I
└ = Ca + S ▣ = CLEAR/HOME
┌ = Ca + Z ▣ = SCHIFT + veassz /SCHIFT + FLASH ON/
┐ = Ca + X ▣ = CTRL + 8

A Z = idezjel,Ca+A, Stezsr SCHIFT+C, Ca+B,
Stazsr SCHIFT+ C, Ca+3, huzsonhét azókés,
SCHIFT + B, St azókés, SCHIFT + B, St azókés,
SCHIFT + B, idezjel
B X = idezjel, SCHIFT + B, St azókés, SCHIFT + B,
St azókés, SCHIFT+B, huzsonhét azókés, SCHIFT+B,
St azókés, SCHIFT+B, St azókés, SCHIFT + B, idezjel
C Y = idezjel, Ca,Q, Stezsr SCHIFT + S, SCHIFT + plusz jel,
Stazsr SCHIFT + C, Ca+R, idezjel
D Z = idezjel, SCHIFT + B, St azókés, SCHIFT + B,
St azókés, SCHIFT+B, huzsonhét azókés, Ca+Z,
Stazsr SCHIFT+C, Ca+B, Stezsr SCHIFT + C, Ca+X,
idezjel

```

5 0 2 0 A Z = "
6 0 3 0 B X = "
6 0 4 0 C Y = "
6 0 5 0 D Z = "

```

A Z = idezjel, Ca+A, Stezsr SCHIFT + C, Ca+R,
Stazsr SCHIFT+ C, Ca+3, Stezsr SCHIFT + C, Ca+R,
huzsonegy azókés, SCHIFT + B, St azókés, SCHIFT + B,
St azókés, SCHIFT + B, St azókés, SCHIFT + B, idezjel
B X = idezjel, SCHIFT + B, St azókés, SCHIFT + B, St azókés,
SCHIFT + B, St azókés, SCHIFT + B, huzsonegy azókés,
SCHIFT + B, St azókés, SCHIFT + B, St azókés,
SCHIFT + B, St azókés, SCHIFT + B, idezjel
C Y = idezjel, Ca+Q, Stezsr SCHIFT + C, SCHIFT + plusz jel,
Stazsr SCHIFT + C, SCHIFT + plusz jel, Stezsr SCHIFT + C,
Ca+W, idezjel
D Z = idezjel, SCHIFT + B, St azókés, SCHIFT + B, St azókés,
SCHIFT + B, St azókés, SCHIFT + B, huzsonegy azókés,
Ca+Z, Stezsr SCHIFT + C, Ca+R, Stezsr SCHIFT+C,
Ca+B, Stezsr SCHIFT + C, Ca+X, idezjel

idezjelteken belül

1	"SCHIFT + CLEAR/HOME"
0 6	"SCHIFT + CLEAR/HOME"
9 2	"CTRL + 1, CLEAR/HOME, BE"
2 1 5	"CTRL + 1, CLEAR/HOME, ZE"
2 4 0	"SCHIFT + FLASH ON, CTRL + 1, ELRONTOTTAD"
2 5 3	"SCHIFT + CLEAR/HOME"
1 3 0 0	"SCHIFT + 9, azókés"
1 9 0 0	"CTRL + 8, azókés"
4 0 0 0	"SCHIFT + CLEAR/HOME"
4 0 6 0	"CTRL + 1, C, pont K pont, azókés, GRATULALOK"
5 0 2 0	"SCHIFT + CLEAR/HOME"
6 0 0 0	"SCHIFT + CLEAR/HOME"
6 1 0 0	"SCHIFT + CLEAR/HOME"
6 1 1 1	"CLEAR/HOME, CTRL + 1, EN"
6 2 0 7	"CLEAR/HOME, CTRL + 1, ZE"
6 3 0 0	"SCHIFT + FLASH ON, CTRL + 1, ELRONTOTTAD"
6 3 7 0	"SCHIFT + CLEAR/HOME"
7 0 0 0	"SCHIFT + CLEAR/HOME"
7 0 6 0	"CTRL + 1, O pont, K pont, azókés, GRATULALOK!"
7 0 9 0	"SCHIFT + CLEAR/HOME"
8 1 0 7	"CLEAR/HOME, CTRL + 1, TE"
8 2 0 0	"SCHIFT + FLASH ON, CTRL + 1, ELRONTOTTAD"
6 3 4 0	"CLEAR/HOME, CTRL + 1, ER"
6 1 9 0	"CLEAR/HOME, CTRL + 1, EN"

A játékprogramozás technikája

HELIKOPTER

A cikksorozatban egy helikopter-játék-program részletes ismertetésén keresztül bemutatjuk a játékprogramokat felépítő elemeket, azok kapcsolódását, a játékprogramok általában használt megoldásokat. A program eredetileg DRAGON számítógépre készült, de a táblázatban szereplő változtatásokkal egyszerűen átirható IBM PC számítógépre vagy MSX típusú gépekre is. A gép sajátosságaiból eredő megoldásoknál utalunk arra is, hogy az adott játék milyen módon írható át más számítógépekre. Cikkeinkben a fő feladatokat, azok megoldásmódját és az általános tudnivalókat tárgyaljuk.

```
130 CLS:INPUT"DOES YOUR DRAGON W
ORK AT DOUBLE SPEED?";I#
140 ILEFT*(I#,1)="Y" THEN POKE
5495,0 ELSE POKE#5494,0
150 CLEAR2000
160 HS=0
170 CLS:PRINT"----- CHOPPER CO
MMAND -----"
180 PRINT
200 PRINT:PRINT"          FOR THE
DRAGON 32"
210 GOSUB1090
220 GOSUB1000
```

1. lista

2. lista

```
1890 PRINT:PRINT"          INSTRUCT
IONS?(Y/N)"
1900 I#=#INKEY#
1910 IFI#="Y" THEN1940
1920 IFI#="N" THEN RETURN
1930 GOTU1900
1940 CLS:I#=#CHR*(128)
1950 FORT=#T031:POKE1024+T,143:P
OKE1504+T,143:NEXTT
1960 FORT=#T0480:STEP32:POKE1024+
T,143:POKE1055+T,143:NEXTT
1970 PRINT"          CHOPPER COMMA
ND"
1980 PRINT#10+64,"INSTRUCTIONS";
1990 PRINT#3+128,"THE";0#;"OBJEC
T";0#;"OF";0#;"THE";0#;"GAME";0#
;"IS";
2000 PRINT#4+160,"TO";0#;"CATCH"
0#;"THE";0#;"FIVE";0#;"CREW";0#
;
2010 PRINT#3+192,"OF";0#;"A";0#;
"FALLING";0#;"PLANE";0#;"IN";0#;
"YOUR";
2020 PRINT#1+224,"HELICOPTER";
2030 PRINT#2+208,"RIGHT";0#;"JOY
STICK";0#;"MOVES";0#;"CHOPPER";
2040 PRINT#2+320,"BUTTON";0#;"AC
TIVATES";0#;"TURBO";0#;"BOOST";
2050 PRINT#6+352,"LAND";0#;"ON";
0#;"PAD";0#;"TO";0#;"REFUEL";
2060 PRINT#8+416,"ANY";0#;"KEY";
0#;"TO";0#;"START";
2080 IFINKEY#="" THEN2000
2090 RETURN
```

Előzetes

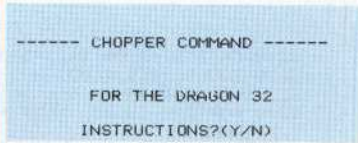
Egyes játékprogramoknál, mint a bemutatottnál is, előzetes információkat kér a program a játékostól. Megkérdezi például, hogy képes-e a gép kétszeres sebességgel is dolgozni.

Itt hívjuk fel a figyelmet arra, hogy minden játéknál igen fontos, hogy a képernyőn megjelenő üzenetek elégségesek legyenek a játék játszásához, és hogy ne tartalmaznak félreérthető kifejezéseket. Az 1. lista mutatja, hogy ezt az adott esetben hogyan valósítják meg. A teljes programlistát egyébként részletekben közöljük, mindig éppen azt a részletet, amelynek magyarázatával foglalkozunk. E részletek összerakásával adódik a teljes játék.

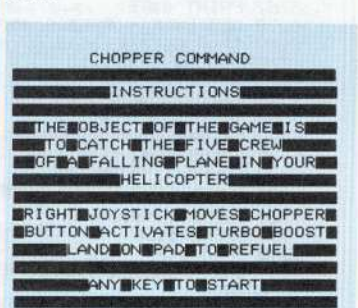
Válaszunktól függően a gép bekapcsolja az egyszeres vagy kétszeres sebességet. Ez egy-egy POKE utasítással történik. Az 1. listában szereplő CLS utasítás a képernyő törlésére szolgál.

3. lista

```
1000 PMODE4,1:PCLS
1010 DRAW#BM100,100;R12L6D6U4L4G
1D2L1R14U2G1D1L1L4U1L1R1U1H1D1L2
G1R1F1D2R2L6U2D2L2"
1020 DIMCL(23,15)
1030 GET(97,96)-(120,111),CL,G
1040 PCLS
1050 DRAW#BM101,100;L12R6D6U4R4F
1D2R1L14U2F1D1R1R4U1R1L1U1E1D1R2
F1L1G1D2L2R6U2D2R2"
1060 DIMCR(23,16)
1070 GET(82,96)-(105,112),CR,G
1080 PCLS:DRAW#BM100,100;F2E2G2D
1R2L4R2D2R1D2L2U2R2"
1090 DIMM(5,13)
1100 GET(99,95)-(104,108),M,G
1110 PCLS:DRAW#BM100,100;R13L7D2
R2F1D2G2D1R1L1U2L2D2L1R1U2L1U1R4
L2U1D1L3U2E1R3"
1120 DIMCM(23,17)
1130 GET(99,95)-(118,113),CM,G
1140 PCLS:DRAW#BM100,100;R2D2L2U
2D2R1D2R2L4R2D1F2HG2"
1150 DIMM1(5,13)
1160 GET(99,95)-(104,108),M1,G
1170 PCLS:DRAW#BM99,105;R6L3U1R1
D2L2U2"
1180 PRESET(102,105)
1190 DIMM2(6,13)
1200 GET(99,95)-(105,108),M2,G
1210 PCLS:DRAW#BM100,100;R2D2L2U
2D2R1D2R2U1D1L4D1U1R2D1F1R1L1H1D
"
1220 DIMM4(5,13)
1230 GET(99,95)-(104,108),M4,G
1240 PCLS:DRAW#BM100,100;D4R2E1U
2H1L2R2BR5;R1F1D3U2L3D2U3E1R1BR4
;D4U4F4U4BR3;R3L3D4R3U2L1BU2BR4;
R3L3D2R2L2D2R3BU4BR3;D4U4R3D2L3R
1F2"
1250 DIMDA(35,8);GET(99,99)-(137
,107),DA,G
1260 RETURN
```

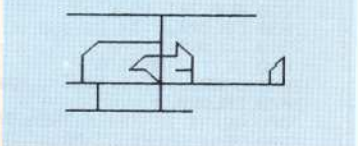


1. ábra



2. ábra

3. ábra



A címlap

Minden játék egy többé-kevésbé szíves címlappal kezdődik. Esetünkben a díszítés meglehetősen egyszerű: a cím mellett kétoldalt szaggatott vonal és néhány üres sor. Gyakran alkalmaznak különböző grafikai díszítéseket, ami azonban a lényegnek nem érinti. Az a fontos, hogy a címből kiderüljön: mi a játék neve, milyen gépre készült. Sokszor a gyártó és a kifejlesztő neve, továbbá a program védettsége is itt található (lásd az 1. és a 2. listát).

Az 1. ábra a képernyőt mutatja. A számítógép megkérdezi, szükségünk van-e utasításokra vagy sem. Ez a megoldás általában sorsra utal. A kezdő játékosok számára ugyanis a játék használatának módja kétféleképpen közölhető: vagy írásos anyag formájában, amire akkor van szükség, ha a játék használatához előzőleg sok információt kell megismernie, vagy a képernyőn megjelenő leírás formájában.

A mi játékunk nem túl bonyolult, ezért a szerző a második megoldást alkalmazta. Ha a játékos már gyakorlott, akkor nincs szüksége az utasítások állandó ismétlésére, ezért lehetőséget hagynak arra, hogy ezt elkerülhesse.

Programunk vizsgálatá csak akkor lesz teljes, ha végigkövetjük, ezért kérjük az utasítások közlését, amit az Y feliratú gomb le-

DRAGON-BASIC	IBM PC és kompatibilis GWBASIC
CLEAR n, m n — a karaktersorozatok részére igényelt memória nagysága m — a lefoglalt terület felső határa	CLEAR m, n n — a veremterület nagysága m — azonos
CLS n PCLS n PMODE 4,1 SCREEN 1,1 LINE utasításnál PSET, PRESET PUT utasításnál —(X2, Y2)	COLOR n COLOR n nincs helyette semmi SCREEN 1
PLAY utasításnál	helyette 1, ill. 0 hiányzik, így részlet nem vehető elő V nincs, így a hangerő nem szabályozható nincs G
GET utasításnál SOUND utasításnál relatív skálaérték PRINT @ n	helyette frekvenciaérték LOCATE INT (n/40), n—INT (n/40) : PRINT
Szövegelhelyezés a képernyőn	
Sor 16	25
Karakter 32	80
Grafika a képernyőn	
Vízszintes 256	640
Függőleges 192	200

nyomására megkapunk. Megjelenik a játékprogram neve, külön bekeretve az a szöveg, hogy utasítás, majd maga az utasítás (2. ábra). A képernyő alján a további teendőket láthatjuk. Ez lehet az, hogy valamilyen billentyű lenyomására lapozni fog, vagyis az utasítások folytatását jeleníti meg a képernyőn, vagy — mint ebben az esetben is — hogy kezdődik a játék.

A programban a folytatás engedélyezését a játékosra kell bízni: az INKEY\$ utasítással vagy „helyben járattja”, vagy továbbbengeti a programot.

Az 1940-es sorban a CLS0 a képernyőnek egy bizonyos színre történő „átmázolását” jelenti. A 128-as kód a karakterek színét váltja át inverz, illetve normál módra. Az 1950-es sorban a POKE utasítások karaktereknek a képernyőre való közvetlen kiírását jelentik. Az 1970-es sorsól pedig szövegkiírás történik a képernyő megfelelő helyére.

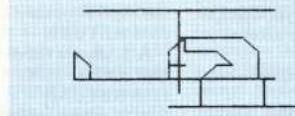
A rajzelemek előállítás

Minden játékprogram tartalmaz rajzelemeket, amelyeket a legtöbb esetben mozgatunk kell. A rajzelemeket a programon belül kell előállítanunk, ha nem akarjuk a számítógép viszonylag szegényes rajzelemkészletére korlátozni rajzudományunkat. Az adott programnál ez az 1000-es sorsól kezdődő, 3. listában történik. Az egyes rajzelemeket láthatatlanul állítjuk elő, vagyis a számítógép képernyőjén még nem jelenítjük meg.

Ennél a gépnél ez elsősorban a DRAW és a GET utasításokkal történik. Mivel ezek az utasítások nem közismertek, és csak kevés gépnél használatosak, részletesebben foglalkozunk velük.

A DRAW utasítás egy karaktersorozatban megadott alutasítás-sorozat megvalósítására szolgál. A benne szereplő karakter-sorozat tulajdonságaiban nem különbözik más karaktersorozatoktól, így a karaktersorozatokra érvényes utasításokkal kezelhető. Az alutasítások a következők lehetnek:

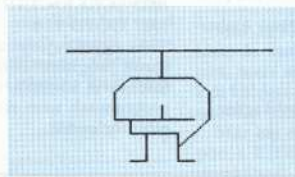
- M x, y — egy adott koordinátájú pontra viszi a fénypontot
- R, L, U, D, E, F, G, H n — a 8 fő irányba mozgatja a fénypontot n lépéssel, miközben azokat a pontokat, amelyeken keresztülhalad, rajzolja
- B — az előzők elé írva, a rajzolat letiltja
- N — a végponttól visszaugrik a kezdőpontra
- A n — n-szer 90 fokkal elforgatja az ábrának azt a részét, amit már megrajzoltunk
- S n — a rajznak azt a részletét, amit már megrajzoltunk, n/4-szeresére nagyítja (vagyis kicsinyíteni és nagyítani egyaránt tud), n értéke 1–62 lehet
- X — azt jelzi, hogy egy újabb karaktersorozatot is figyelembe kell venni a rajzoltató utasításnak; így a karaktersorozatokra



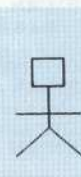
4. ábra



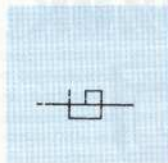
5. ábra



6. ábra



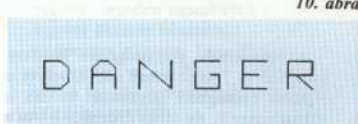
7. ábra



8. ábra



9. ábra



10. ábra

ennél a gépnél érvényes 255 karakteres maximum átléphető, mert újabb 255 karakter helyettesíthető be ezzel az alutasítással, illetve ha az újabb karakter-sorozatban is szerepel egy további X, akkor még további is

C n — a további rajzrészecskék színét változtatja meg

A GET utasítás a képernyőnek egy főátó végpontjaival megadott téglalap alakú területét a memóriába másolja. A téglalap területének nagyságát csak a memória mérete korlátozza. A téglalapok egy-egy numerikus változóként megnevezve kerülnek a tárbá. Darabszámuk nincs korlátozva. Használatuk előtt a numerikus változókat dimenzionálni kell.

A listánkban szereplő egyéb különleges utasítások a képernyő grafikus üzemmódra átkapcsolását (PMODE), felbontásának beállítását (SCREEN), grafikus üzemmódban a képernyő törlését (PCLS), illetve egy pont törlését (PRESET) valósítják meg. Az egyes rajzelemek az ábrákon láthatók. Mivel ezeket a program nem látható formában állítja elő, 8 kis programmal (il-

létve az eredeti program módosításával) tettük őket láthatóvá, és a nagyítótúratást használva olyan méretűvé, hogy a képek jól áttekinthetőek is legyenek.

A 3. ábrán helikopterünk látható, balra fordított állapotban.

1005 SCREEN 1,1:DRAW"S32"
1015 RETURN

A 4. ábra a jobbra fordított helikopter.
1007 GOTO 1050

1055 RETURN

Az 5. ábra a fejjel lefelé zuhanó emberke.

1007 GOTO 1080

1085 RETURN

A 6. ábra a szemből nézett helikopter.

1007 GOTO 1110

1115 RETURN

A 7. ábra az álló helyzetű emberke.

1007 GOTO 1140

1145 RETURN

A 8. ábra a fekvő emberke.

1007 GOTO 1170

1185 RETURN

A 9. ábra a jobbra haladó emberke.

1007 GOTO 1210

1215 RETURN

Végül a 10. ábra a „veszély” felirat.

1005 SCREEN 1,1:DRAW"S16"

1007 GOTO 1240

Az egyes rajzelemek kirajzoltatása a felsorolt programrészeket és a RUN 230 betoldásával történik. A betoldások a játékprogramnak nem részei, ezért a használat előtt tölendők.

GALINA FERENC

PROGRAMOZÓK

MESÉLIK



A Novotrade számítógépes játéktér-méséből a Traffic című az, amelyet a legtöbb országban játszanak. Itthon is sokan ismerik, de talán érdemes leírni a lényegét: egy londoni térképészleten közlekedési lámpákat irányít a játékos, és az a feladata, hogy elkerülje a forgalmi dugók kialakulását.

Tetszetős alapötlet, szépen megcsinált játék. Egy gond mégis volt vele: szinte senki sem tudta végigjátszani. Az első, viszonylag egyszerű térképészlet után ugyanis mind nehezebbek következnek, egyre több kereszteződéssel és autóval. Még maguk a programozók is egy könnyített változatot tesztelték a programot, hogy egyáltalán eljussanak az utolsó szintig. Egy-két, már-már hivatásos játékosnak nevezhető srácn kívül, akik hosszú heteket töltöttek vele, senki sem járt a végleges változat ötödik térfélpén.

A Commodore 64-re írt játék angol megrendelésre elkészült Amstrad gépre is, majd felfigyelték rá a japánok, és meg is rendelték az MSX-re való átirást.

Ilyenkor a vevőnek mindig akad néhány külön kívánsága, megjegyzése, hogy szerinte hogyan lehetne az ő vevői igényei szerint még jobbá tenni a játékot. A japánok esetében számítani lehetett rá, hogy változtatást kérnek például a grafikában, a címképen, ebben-abbban, de arra, ami történt, senki sem számított.

Telex érkezett a japán partnertől,

amelyben a szokásos udvariasság után („igazán nagyszerű a játék, kevés ehhez foghatóval találkoztunk stb.”) rá-tertek a lényegre: a játék túlságosan könnyű! Lehetne-e nehezíteni rajta? Például egy-egy karambolla, a hozzá tartozó mentő- és rendőrautóval, amelyek egy időre minden forgalom elől elzárnak egy kereszteződést?

A játék ebben a már-már játszhatatlannul bonyolult verzióban készült el az MSX-en, de a játékstúdió munkatársait nem hagyta nyugodni a kérdés: mitől könnyű a nehéz játék a japánoknak? Japánba utazva utána is néztek a dolognak.

A történet nélkülözi a csattanót. Kiderült, hogy amikor a Traffic eredeti Commodore-változata megérkezett, két munkatársat ráállítottak, és azok ketten vagy két hétig semmi egyébbe nem foglalkoztak, mint azzal, hogy a Trafficet játszották újra meg újra. A végén már minden motorkerékpárt személyesen ismertek, minden térképet úgy át tudtak tekinteni, mintha azon a környéken nőttek volna fel. Röviden: szakértőivé váltak a témának. Ezután mondtak véleményt, amit telexen elküldtek Magyarországra.

Az előbb azt írtuk: a történetnek nincs csattanója. A programozók és a játékstúdió számára azonban annál több a tanulsága.

R. T.

KEDVES OLVASÓINK!

Egyre több olyan levelet kapunk, amelynek írói beküldött cikkeik, programjaik megjelenését sürgetik. Egyetlen szerkesztőség sem vállalja, hogy minden beérkezőt irást közölni fog, de azt sem, hogy azokat, amelyek nem jelennek meg, visszaküldi. Nagyon sok cikket, programot elteszünk azonban, hogy esetleg később lapunkban vagy a *µ*könyvtár kötetekben felhasználjuk.

Holicsék Tibor, Tatabánya,

Sárberei ttp. 116. 2800

8. osztályba járok, és előttem a pályaválasztás. Most gimnáziumba jelentkeztem, de tanulmányaimat a számítógép közelében szeretném folytatni. Igaz, hogy ez egy kicsit még messze van, de azért megkérdelem a pesti és győri főiskola pontos címét és a bekerülési esélyeket.

Én is azt hiszem, hogy a főiskola utáni érdeklődés még egy kicsit korai, de discszerű. Véleményem szerint a főiskola vagy az egyetem kiválasztásával még egy-két évig érdemes várni, mert a középiskolában nem ritka, hogy az ember érdeklődése is megváltozik. Persze az sem tragédia, ha más pályát választ, hiszen ma számítástechnikával, számítógépekkel szinte valamennyi szakmában találkozni lehet.

Pataki Attila, Pécs,

Egri Gyula u. 42. 7632

Úgy tudom, hőérzékeny papírt nem gyártanak Magyarországon. Viszont több terbe csak ilyen speciális papír jó. Kérem, írják meg egy olyan nyugatnémet vagy osztrák gyár címet, ahol hőérzékeny papírt gyártanak.

Sajnos gyári nevet nem tudok mondani, de ha jól emlékszem, akkor ilyen papírt láttam a bécsi és a linzi Computer City boltokban (Wien, 1030, Marokkanergasse 3/1/21.).

Funika Vlagyimir, 277028, Kisinyov,

Groszula u. 15/2, Szorjetunio

Programozó vagyok, a moldavai tudományos akadémián dolgozom. Nagyon érdekel a számítástechnika és a szoftver fejlődése Magyarországon. Ezért meg szeretnék ismerkedni olyan programozókkal, akik SZM—4, SZM—1420, SZM—1600, PDP—11, TPA—1140, MERA—60 típusú számítógépekkel, Elektronika—60, ISZKRA—226, Wang típusú mikroszámítógépekkel, IBM PC személyi számítógéppel dolgoz-

nak. Könyveket és információkat szeretnék cserélni velük. Tudok angolul, németül, lengyelül, csehül, ezenkívül magyarul olvasok. Várom a leveleket.

*Ez az első levélünk a Szovjetunióból, aho-
va — úgy látszik — ugyancsak eljutott a
µM. Szívesen tesszük közre ismerkedési fel-
hívását. Remélem, hogy talál levelezőpart-
nert a hasonló érdeklődésű szakemberek kö-
zött.*

Szabóné Kis Illdi, Sopron,

Pázmány Péter u. 5. 9400

Tudom, hogy nem vagyok egyedül azzal a gondolattal, hogy rengeteg lehetőség rejlik a számítógép oktatás területén való felhasználásában. De túl kell lépniünk azon a gondolaton, hogy a számítógép elsősorban a matematika és a fizika tantárgyaknál, illetve a grafikus ábrázolásoknál, kísérletek szimulációjánál vagy demonstrációjánál kaphat jelentős szerepet. Az egyszerű szöveg, amely él, változik, mozog, képes lehetőségek cseréjére, variálására, ellenőrzésére, javítására és még ki tudja, hogy mire. Kítűnően szolgál a tudásanyag fokozottabb befogadására, elmélyítésére.

Igaz, még gyerekpőben járunk a programok készítésével, de valahol el kell kezdeni! Jó lenne számítógépes tananyagot összeállítani, mert a végső cél ez! Minél korábban nekikezdünk, annál több lehetőség van a programok fejlesztésére, az oktatás tökéletesebbé tételére. Már lelki szemeimben látom, hogy minden tanuló egy-egy számítógép előtt ül, és élvezettel tölti magába az ismereteket. A pedagógus összeállítja az órára a programokat, segít a tanulók esetleges problémájában, majd az óra utolsó részében tanítványaival beszélget a gyűjtött tudásanyagról, esetleg vitatkozik, eszmecsérelt folytat velük, majd értékeli munkájukat.

A gyerekek érdeklődését, figyelmét elsősorban a játékprogramok már a számítógép felé terelik, és nem vitatható, hogy szívesen tanulnak vele. Miért? Mert a gép türelmes, figyelmes, bármikor segít, és nem lehet ráfogni, hogy kivételez.

Sok-sok programot kellene írni, hogy olyan tananyag álljon össze belőlük, amelynek segítségével minél több elitiskola jöhhessen létre!

*Szívemből beszél, mindennel egyetértek,
amit leírt. Ezt hívják a számítógép és a számítástechnika igazi oktatási alkalmazásának.*

Tóth Péter, Budapest,

Menyecske u. 19. 1112

A szerkesztőség új szolgáltatása ragadtott tollat velem, hogy írjak. Egy viszonylag ritka gép, egy Laser 210 tulajdonosa vagyok, és szeretném, ha más Laser-tulajdonosok címét elküldenék nekem. Tudomásom szerint a Laser 210 kompatibilis a La-

ser 110-zel, 310-zel és talán a 200-zal is, így ha ilyen gép tulajdonosa jelentkezne, kérem, hogy az ő címét is küldjék el nekem.

Az én gépemhez nincs bővítés, 8 k RAM memóriája van. Saját magnómon kívül más perifériám nincs, és a DEMO-kazettán kívül gyári szoftverem sincs. Viszont van néhány saját készítésű játékprogramom, igaz, még mindegyik BASIC-ben. Most from első gépi kódot használó programomat.

*Levélét és felhívását szívesen közreadjuk,
hogy a Laser-tulajdonosok találkozását ezzel is elősegítsük. Ha elég sokan lesznek, ajánlom, keressék meg dr. Simonyi Endrét, és alakítsák meg a Laser-szekciót a HCC keretében.*

Molnár László, Pécs,

Korvin Ottó u. 17. 7629

Laser 210-es számítógépem perifériaki-
menetén végigcsúszott egy szál drót, és azóta összevissza mutat. Szíveskedjenek megírni, hogy hol tudnám megjavíttatni a gépet, mert nálunk nem foglalkoznak vele.

Sajnos csak levélnek közlésével tudunk segíteni, hátha egy Laser-tulajdonos tanácsot tud adni. Sokszor elmondtam már, hogy igen nagy kockázat olyan számítógépet vásárolni, amiből kevés van az országban, vagy amelynek javítószolgálatát nem szervezték meg.

Lovász Tibor, 94612 Zlatné na Ostrove

Tuzová 366, okr. Komárno, Csehszlovákia

A nálunk eléggé elterjedt SORD M-5 számítógép gépi kódú programozásához sajnos nem lehet irrodalmat szerezni. De a programozással még csak elboldogulnék, ám nincs ROM- és RAM-táblázatom (belső rutinok stb.). A másik problémám, hogy a videomemória a többi 64 kbájton kívül van, nem tudom aszembenben irányítani a Z80 utasításait. Kérem, küldjenek vagy ajánljanak erre valami orvosságot.

*Azt hiszem, megint nem tudunk segíteni,
így olvasóink tanácsát kérjük.*

Styt Sándor, Budakeszi,

Vörös Hadsereg út 88—90. 2092

I. Érdeklődni szeretnék, elkészült-e a MEV kitje? Ha igen, hol lehet kapni vagy

előjegyezni, mit tud, mennyibe kerül. Milyen perifériák vannak hozzá?

2. Hogyan lehet valaki a HCC tagja, mennyi a tagdíj, mit tud nyújtani a tagság?

1. Sajnos még nem. 2. Keresse meg dr. Simonyi Endrét az 556-245-05 telefonszámon. Tagsági díjat nem kell fizetni.

Kun László, Budapest,

Koppány u. 11. 1205

Már az Ötlet BIT-LET rovatának megírtam, de most megismétlem a közleményemet, mert azt olvastam, hogy erre a címre kell küldeni a mikroklubok működésére vonatkozó információkat.

Már két éve működünk, önfenntartó alapon, a KIOSZ segítségével, azaz társadalmi munkában, megszállottságból.

Klubtagságunk félénként megújul (cserelődik), de számos törzstagunk is van. Legnagyobb létszámunk eddig 100 körül volt.

Örömmel vennék, ha figyelemmel kísérnék munkánkat. Tavaly már felvettek a mikroklubok névsorába. Azóta csak az a változás, hogy van videókn is, és egyre több a C16-tulajdonos a tagok között. Örömmel fogadtuk az árrobbanást!

TI—99/4A SZEMÉLYI SZÁMÍTÓGÉP ÉS VIDEO KLUB

Nyitva: minden szombaton 9—13 óráig. Egyéni oktatás bármikor!

Csoportos oktatás: 11—13 óráig videóval segítve, a TV-BASIC felhasználásával. A tavalyi vizsgapédákat folyamatosan megbeszéljük.

A klub számítógépei vagy a saját számítógépek használhatók.

SZÁMÍTÓGÉPEK: TI—99/4A, Sinclair ZX81, ZX-Spectrum, C16, C20, C64 és amelyet még hoznak.

Helye: 1181 Budapest, Baththyány Lajos u. 78/a.

Telefon: 280-238.

Videókkal levelez a klubvezető: Kun László villamosmérnök-tanár, 1205 Budapest, Koppány u. 11.

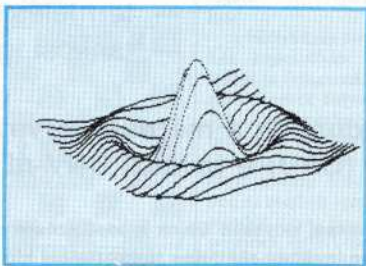
Lehetőség van a szekciókon belül a tanulási mellett a programok cseréjére is.

Féléves tagsági díj felnőtteknek: 800 Ft, KIOSZ-tagoknak és tanulóknak: 500 Ft.

Annak ellenére, hogy a szöveg egy kicsit reklámizú, a felhívást örömmel közöljük,

hiszen a társadalom informatizálása programban a klub mozgalmat rendkívül fontosnak tartjuk. Szeretnénk, ha sikerülne a klubokkal aktív kapcsolatot kialakítani, írniak tevékenységükről, egyáltalán a gyakorlatilag spontán létrejött klubokat valódi mozgalommá szervezhetnénk. Szeretnénk létrehozni azt az országos rendszert, amelynek segítségével országos méretű tapasztalatcsere, a klubok között szoros munkakapcsolat jöjjön létre. A mozgalom megszervezéséhez „megszállott és a számítástechnika iránt elkötelezett” klubvezetők és klubtagok jelentkezését várjuk.

KOVÁCS GYÖZŐ



A megrendelhető gépi konfigurációk a következők:

— egy munkahelyes kiépítés
I 8088 CPU 256 kbájt RAM
színes monitor
billentyűzet
360 kbájt floppy
27 Mbájt Winchester

— több munkahelyes kiépítés
I 8088 CPU 640 kbájt RAM
színes monitor
billentyűzet
360 kbájt floppy
10 Mbájt streamer
2 × 27 Mbájt Winchester
csoportvezérlő kártya
max. 4 terminál

- memóriabővítési lehetőség
- alapgépből folyamatos bővítéssel kialakítható hálózati rendszerek
- több munkahelyes csoportvezérlős rendszerek
- operációs rendszerek
- programnyelvek
- különféle segédprogramok
- matematikai CO-processor.

A gépekhez különféle felhasználói rendszereket is szállítunk igény szerint.

A fenti gépi konfigurációkat AT nagyságrendben is tudjuk szállítani:

I 80286 CPU 512 kbájt RAM, színes monitor, billentyűzet, 1,2 Mbájt floppy és 27 Mbájt Winchester kiépítésben.

Széles bővítési lehetőséggel, megfelelő operációs rendszerekkel és programnyelvekkel, illetve felhasználói programrendszerekkel állunk rendelkezésre.

Felhasználói rendszereink

- főkönyvi könyvvitel
- folyószámla-vezetés
- készletnyilvántartás
- állóeszköz-nyilvántartás

- bér- és munkaügy
- rendelés-nyilvántartás
- mérnöki tervezőrendszer-applikáció
- irodai szoftver
- termelésirányítás
- tanácsi igazgatási alrendszerek

Tekintse meg standunkon a mérnöki tervező munkahelyünket (CAD) a tavaszi BNV-n!

Előbbi cikkeinkben a táblaindexes ábrázolási módhoz tartozó lépésgenerálás fajtáival foglalkoztunk, a halmazmező módszerrel, a táblavezérlésű módszerrel és a lépéslista folyamatos felújításával. Ezáltal a bittérképes ábrázolási módhoz tartozó lépésgenerálást mutatjuk be.

A *figuratípusonkénti bittérkép* ábrázolásához 8×15 bájt szükséges. A saktábla 64 mezőből áll, és figuratípusonként minden mezőnek egy-egy bit felel meg, ezért egy-egy figuratípushoz 64 bit, azaz 8 bájt tartozik. Mivel 6-6 világos, illetve sötét figuratípus létezik (gyalog, futó, huszár, bástya, vezér, király), ez $2 \times 6 \times 8 = 12 \times 8$ bájtot foglal el, amihez színenkénti bittérkép csatlakozik 8-8 bájtban — ez az összes világos, illetve sötét figura elhelyezését regisztrálja —, majd 8 bájtban az a bittérkép, amely egybevetve a sötét, illetve világos figurákat, azt mondja meg, hogy hol van a táblán figura, nem téve különbséget a világos és sötét bábuk között. Ahol a figura áll, ott a bit értéke magas, ahol a tábla üres, ott alacsony.

Ábrázolhatjuk az állást *figuránkénti bittérkép* is, amelynek technikája hasonló az előbb említett figuratípusonkénti bittérképhez. De azaz a különbséggel, hogy itt az egyes típusokon belül figuránként lebontható tároljuk az állást. Ez azt jelenti, hogy például különbséget teszünk az alapállásban B1-en, illetve G1-en álló huszár között, és ezekhez külön-külön bittérkép tartozik. Ugyanígy a többi figurára: futókra, bástyákra, gyalogokra.

A bittérképes módszer előnyét főként az új 16 és 32 bites mikroprocesszorral működő professzionális személyi számítógépek nyújtotta lehetőségekkel használhatjuk ki, ugyanis ezek a 64 bitből álló rekordokat, azaz a táblapozíciót egy-két utasítással letapogatják.

Amint tudjuk, két bináris operandus logikai ÉS (AND) művelete azon a helyiértéken eredményez magas bitértéket, ahol a két eredeti operandus mindegyikében logikai 1 érték volt. A logikai VAGY (OR) művelet esetében azon a helyiértéken magas a bit értéke, amelyen a 2 operandus azonos

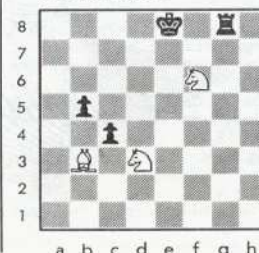
BITEK ÉS FIGURÁK LÉPÉSGENERÁLÁS III.

helyiértéken bármelyik bit értéke 1. Ezeket az egyszerű műveleteket használhatjuk fel a lépésgenerálásnál is.

A bittérképes ábrázolási módhoz kapcsolódó lépésgenerálás eredménye újabb bittérkép, szemben a táblaindexessel, amellyel konkrét lépésekhez jutunk. (A mi esetünkben nem konkrét lépésekről, hanem csak újabb állásokról van szó.) Így az értékelő függvény megkonstruálásánál csak az állások értékelésére hagyatkozhatunk.

Hogyan is generáljuk a lépéseket? Először is a programban letároljuk az összes mezőn minden figuratípus lehetséges lépéseinek bittérképeit. Ez azt jelenti, hogy a 64 mezőből álló saktábla minden egyes mezőjén külön-külön elkezdjük az egyes figuratípusokat. Először vesszük a gyalogot és a hozzá tartozó bittérképek sorozatát, vagyis azokat a bittérképeket, amelyek létrejöhetnek egy „félleg legális” gyaloglépés esetén. (A félleg legális kifejezés, amint már az előző cikkemben kifejtettük, azokat a lépéseket jelöli, amelyek legálisan megtehető lennének, ha nem vennénk figyelembe, hogy a lépés megtételével királyunk sakkban maradna vagy sakkba kerülne, illetve hogy a lépéssel saját figurát ütnénk ki, ami lehetetlen.) Ugyanezen a mezőn elkezdelhetjük a futót, a huszárt, a bástyát, a vezért és a

Világos huszárvilla — sötét gyalogvilla



királyt, majd mindezekre előállítva a gyalog mintája szerinti bittérképek sorozatát, megkapjuk egy mezőre az összes figura félleg legális lépéseiből adódó állások bittérképeit.

Ezt a műveletet minden mezőre elvégezve megkapjuk azt az adathalmazt, amelyet a lépésgenerálásnál felhasználunk.

A könnyebb megértés kedvéért a lépésgenerálás mechanizmusát példával világítjuk meg. Számítsuk ki a C3 mezőn álló világos huszár összes lehetséges lépéseinek a sorozatát. Ez tulajdonképpen bittérképek sorozatát adja.

A legális lépések eléréséhez vegyük elő a memóriából a C3 pozícióhoz tartozó félleg legális huszárlépések bittérképeit, majd vegyük elő a világos figurák jelenlegi állását tartalmazó térképet! Ezt negálva (vagyis a bitek értékeit felcserélve: a nulla helyébe egyet, az egy helyébe nullát írva) olyan térképhez jutunk, amely a logikai magas bitpozíciókkal jelzi, hogy melyik táblamezőben nincs világos figura. Ez a térkép és a lehetséges huszárlépések ÉS művelete adja az adott pozícióhoz tartozó megengedett huszárlépéseket. E művelet elvégzése után a huszár azokra a mezőkre léphet vagy üthet, amelyeknek bitpozíciója 1 értéket mutat. Ugyanígy járhatunk el a sakkadás és a lekötések vizsgálatánál is. Végül is így hozzájuthatunk a teljesen legális bittérképek sorozatához.

A bittérképes ábrázolásmódhoz kapcsolódó lépésgenerálásnál is felhasználhatjuk a legális lépések folyamatos megújításának módszerét, a táblaindexes ábrázolásmód lépéslistájának folyamatos megújításával analóg módon. Így ebben az esetben sem kell mindig minden egyes figura legális lépéseinek bittérképeit előállítani, hanem csak differenciáltan,

az előző cikkben kifejtett szabályban foglalt hat pont szerint.

A bittérképes ábrázolási mód a sakkprogram számos más fázisában jól kihasználható. Például könnyen meghatározhatjuk vele az egyes figurákat támadó vagy védő bábukat, vagy a királyt leginkább veszélyeztető bábukat. Jellegzetes további példa: meghatározhatjuk a módszerrel, hogy egy adott állásban valamelyik huszárral (esetleg gyaloggal) „villába” tudjuk-e fogni az ellenséges bástyát és királyt, vagy bármely más két bábút (ábra). Ehhez meg kell határozni, hogy léteznek-e olyan mező, ahonnan egy huszár az ellenséges király és bástya pozícióját egyszerre elérheti; huszárrunk egy lépésben el tudja-e érni ezt a „villa”-mezőt; védi-e az ellenfél ezt a villamezőt.

Ha egy sakkzó ránez az állásra, ezeket egy pillanat alatt meg tudja állapítani. Egy bittérképes lépésgeneráláson alapuló program alkalmazásával a számítógép ugyanezeket így regisztrálja.

Vegyük elő a memóriából az ellenséges király, bástya és a mi huszárrunk jelenlegi helyzetéhez tartozó félleg legális lépéseinek bittérképeit, ezután az összes világos figura bittérképeinek a negáltját. E négy bittérkép között az összes sorban elvégezve az ÉS logikai műveletet, olyan bittérképhez jutunk, amely megmutatja, léteznek-e ilyen villamező, és ha igen, hol. Ha létezik, akkor az ellenfél összes figurája által támadott mezőt hozzuk logikai ÉS kapcsolatba az előző eredménnyel. Ha az eredmény zérus, ez azt jelenti, hogy ezt a villa számára alkalmas mezőt az ellenséges figurák nem támadják.

E példa megmutatja, hogy bittérképes ábrázolás esetén milyen kevés művelettel tudunk egyes kérdésekre válaszolni. Ezzel növeljük a gyorsaságot, így programunk hatásfokát. Ámbár később kiderül, hogy a bittérképnek sok előnye mellett hátrányai is vannak. Magunk ezt a KEMPELEN I. sakkprogram Sinclair Spectrum számítógépre történő átírásakor tapasztaltuk.

KOVÁCS P. ATTILA

Dr. Brückner Huba: Videotex rendszerek és alkalmazásai

(Budapest, 1985.

SZÁMALK,
277 oldal. Ára: 136,— Ft)

Ez a könyv az eddigi gyakorlat-tól eltérően, átfogó jelleggel tárgyalja az interaktív és sugárzott videotex rendszereket. Műszaki és alkalmazástechnikai szempontokra koncentrált, ezzel széles körű olvasótáborra érdeklődési kört célozza meg.

Külön érdeme a szerzőnek, hogy bármilyen témakörrel legyen is szó, rendszerismerleti igénynél közelebbi meg a felmerülő kérdéseket, ami nagyban hozzájárul a könyv olvashatóságához. Így például az egyes áramkörök megoldások ismertetésében is sikerült elszakadnia a hagyományos tárgyalásmódtól. Ezáltal a laikus, az implementációs kérdéseket kevésbé ismerő olvasó számára is élvezhetővé válnak a hardvert ismertető fejezetek.

A témákhoz való hűség kritériumát vizsgálva a könyv kiállja a próbát. Az aktualitás gondja perze a nyomdai átfutási idő és a témákhoz igen gyorsan változó helyzete miatt joggal merül föl. Erre maga a szerző is utal. A videotex rendszerek, és főképp a párbeszéd-vezérlő változatok terén nap mint nap új rendszertechnikák és áramköri megoldásokról, szoftvertermékekről, eljárásokról olvashatunk. A szabványosításban is a kézirat elkészülte óta eltelt időben történtek a legfontosabb események. A fogadtatásra, elterjedésre és társadalmi hatásra vonatkozó nézetek is részben módosultak, számos korábbi prognózis nem állta ki az idő próbáját.

Egy könyvtől természetesen nem szabad megkövetelnünk a műszaki folyóiratok aktualitását. A szerző által felsorakoztatott tényanyag egy 1984. év végi realis helyzetképpel maradv minden képpen kifogástalannak tekinthető. E korlátozást elfogadva még a témában tájékozott olvasó is talál a könyvben számára új tényeket, megoldásokat, fejleményeket.

A könyv felöleli mindazt, amit a videotex témában le lehet, illetve le kell írni. Az adott terjedelmén belül a szerző jól gazdálkodott a hellyel az arányok kialakításában, az egyes fejezetek felépítésében.

Különösen értékesek a megjelenítési szabványokat és létező terminálmegoldásokat ismertető részek. A hazai mikroszámítógépes szakmai környezetben ez a témakör valóban kiemelt jelentőséget kaphat. A szerző talán ezen a területen nyújtja a legtöbb olyan információt, ami a mindennapi gyakor-

lati fejlesztéssel, implementálással foglalkozó szakembereknek is új lehet, segítheti őket a továbblépésben. Kiemelkedően jónak tartom ebből a szempontból a CEPT megjelenítési eljárás és VLSI áramkörei ismertetését. A teletext-Prestel rendszerek főbb jellemzőit tárgyaló F2 függelékre pedig külön fel szeretném hívni a figyelmet! Ez a rész önmagában is elegendő egy Prestel alapterminál mikrogepen történő implementálásához.

Ahol a szerző szűkszavú marad — például az adatbázisok szerkezetéről szóló alfejezetben —, ott ezt az adott területet kevésbé vitx-szempontjából vizsgálhatjuk vissza. A bő irodalomjegyzék viszont jó támpont azok számára, akik egy-egy témakörben jobban el szeretnének mélyülni.

Az adatátviteli hálózatokkal foglalkozó fejezet igen alaposan tárgyalja a hálózati működés architektúráis jellemzőit. A nyit hálózatok architektúrája hatodik-hetedik rétegeibe sorolható funkciókat a gyakorlatban megvalósító NSZK EHKP protokollrendszerrel szóló rész a hálózatok és a táv-adatfeldolgozás kérdéseivel foglalkozó szakemberek számára is újat mond. A virtuális terminál témakörrel rokon EHKP protokollrendszer egy jól sikerült, és nem is alapszintű terminálkezelő eljárás példája. Ismertetése tehát konkrét fejlesztési munkákhoz is jól használható.

Az illusztrációk igen jó minőségűek, a könyv kivitele esztétikus. Sajnálatos viszont, hogy egyes példányok (például a nálam lévő is) hajlamosak a szétesésre.

KOVÁCS P. LÁSZLÓ

Csupa játék ZX-Spectrumra Szerkesztette: Votisky Zsuzsa

(Budapest, 1985.
Műszaki Könyvkiadó,
77 oldal. Ára: 28,— Ft)

A kötetben ismertett játékok programokat középiskolás diákok készítettek. E programok tervezése, megírása, kipróbálása remek alkalom volt, hogy szerzőik programozástechnikai ismereteiket a gyakorlatban is kipróbálják, és a munkájuk során tapasztalt nehézségeket megbirkozva, önálló programok készítésére is vállalkozhassanak.

Az olvasók remélhetőleg a programokat kipróbálva maguk is kedvet kapnak új játékok kitalálására és programjaik megírására. Ehhez az olvasó-programozók a kötetben néhány hasznos tanácsot is találnak.

Reméljük, hogy e kis gyűjteményt, mely a Mikroszámítógép

Magazin mellékleteként jelent meg, továbbiak is követni fogják a többi ismert gépre (Commodore, HT-1080Z stb.) írt játékgrogrammal.

Dusza Árpád —Varga Antal: A BASIC nyelvű programozás ábécéje

(Budapest, 1985.
Műszaki Könyvkiadó,
171 oldal. Ára: 66,— Ft)

A könyv, amely a BASIC nyelv című tankönyv anyagát tartalmazza, a teljesen kezdőket kívánja a számítástechnikai alapfogalmak és a BASIC programozás nyelvi ismereteire megtanítani. Az általános ismereteket tartalmazó fejezeteket a HT-1080Z gépekre alkalmazható BASIC programozásról szóló fejezetek követik, melyekben elsősorban a matematikából és a fizikából ismert példák számítógépes programjai találhatók.

A melléklet a BASIC nyelvű programozás ábécéjének legfontosabb táblázatát gyűjtötte össze.

Dr. Szentcs János: A szoftverminőség és mérése

(Budapest, 1985.
SZÁMALK,
247 oldal. Ára: 107,— Ft)

A rendkívül gazdag irodalomra és tapasztalatra épülő könyv nemzetközi összehasonlításban is egyedülálló áttekintést ad a szoftverminőséget befolyásoló tényezőkről, ezek objektív mérését biztosító módszerek és eszközök kutatásának, gyakorlati alkalmazásának helyzetéről. Az elméleti megalapozottságú rendszeres és tárgyalásmód ellenére a könyv közhelyes. A második részben adott konkrét

útmutatások és eszközismertetés következtében közvetlenül használható bármely olyan szervezetnél, ahol a szoftverminőség javítása érdekében valamit is tenni akarnak.

A mű értékét a gazdag illusztrációk anyag és válogatott irodalomjegyzék mellett emeli még az is, hogy szerzője a terület nemzetközileg ismert szakértője. A részletesen bemutatott hazai fejlesztési minőség-ellenőrző rendszer már negyedik éve tartja magát a világpiacra mint a magyar szoftver egyik legjobban hírverője.

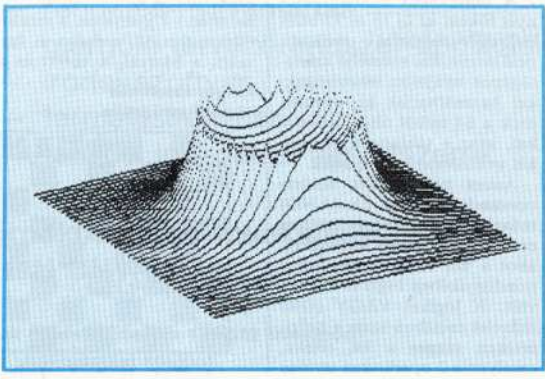
A könyv mindazok számára érdekes olvasmány, akiket egy kicsit is érdekelnek a programozás és a szoftverkészítés aktuális problémái.

Coffron, J. W.: Mikroprocesszorok rendszerek gyakorlati hibakeresése

(Budapest, 1984.
Műszaki Könyvkiadó,
246 oldal. Ára: 80,— Ft)

A szerző foglalkozik a mikroprocesszorok felépítésével, majd leírja a CMS hardverelemző alkalmazását a legismertebb mikroprocesszorok vizsgálatánál. Nagyon ötletes új módszereket mutat be, de tárgyalja az ismert egyszerű vizsgálati eljárásokat is. Hasznos tanácsokat ad a hibakeresés menetére, a vizsgálgató ROM programjára vonatkozólag, végül a TRS-80 személyi számítógépen konkrét hibakereséseket mutat be.

A tartalomról: A mikroprocesszorok rendszerek architektúrája; Hibavizsgáló rendszer (CMS), SST alkalmazása a 8080 mikroprocesszor vizsgálatára; 8085, 6800 és Z80 mikroprocesszorok vizsgálata; Egyszerű hibakereső eszközök; Logikai állapotanalízis; Bitminitavizsgáló; Gyakorlati hibakeresési tanácsok.



Csillagász számítógép

A debreceni Magnitudo amatőr csillagászati kör tagjai az égbolton zajló folyamatok hatékonyabb oktatására számítógépet alkalmaznak. Milyen volt 1985 őszének esti égboltja? Erre a kérdésre a nagylás mértékének és a kívánt időpontnak megadása után kirajzolódik előtűnik az égbolt, majd a Mars, a Jupiter és a Szaturnusz bolygó tűnik fel, és a valóságban megfelelően látjuk mozgásukat.

Számítógépes program segítségével készültek fel a kőr tagjai a Halley-üstökös tavaly novemberi földközeli elhaladására. A november 16-i égboltnak az üstököshöz közeli részét kinagyítva látható, hogy az éppen a Fiasztvíz csillagai alatt halad el.

Az egyik legújabb programtermekük az 1999. augusztus 11-i teljes napfogyatkozás Magyarországról megfigyelhető látványának modellezése. Az égboltra rajzolt Nap előtt elhalad a Hold korongja, és a teljes napfogyatkozás idejére a Nap közelében feltűnnek az Oroszlán csillagkép csillagai.

A programokat ZX-Spectrum házi számítógépre dolgozták ki. A gép színes grafikus lehetőségei elegendőnek bizonyultak, a gép elterjedtsége folytán pedig az oktatási programok széles körben alkalmazhatók.

Név szerint

A francia Tomson cég kifejlesztett egy olyan intelligens telefonot, amely lehetővé teszi, hogy a hívott felet név szerint kérjük. A telefonkagyló mikrofonjába bemozdított nevet egy beszédelfismerő mikro-számítógép elemzi, és kikeresi a névhez tartozó telefonszámot. A szám felhívásáról is a mikrogép gondoskodik.

Biomorzsák

A bioelektronika — szószólónak véleménye szerint — a szilícium alapú számítógépek képességein messze túlmutatva, a szupergepek alapja. A szilícium, germánium és gallium-arszenid egykristályok helyett a bioáramkörök szerves anyagok molekuláiból készülnek. Potenciális előnyeik az elemzésük növekedése, a gyártás egyszerűsödése, a termelésnövekedés, a kisebb költségek, a sugárzásokkal és az elektromágneses hullámokkal szembeni közömbösség. Így a szilícium alapú számítógépek jól ismert gondolj, a hőtermelés, a hibátűrés hiánya és a lassúság leg-

alábbis enyhülnek. A mai félvezetőgyártásban az áramkörü méret (0,4 µm) közeledik a küszöbértékhez; ekkor már majdnem 1 millió tranzistor van egy lapkán. Nagyobb elemsűrűség esetén a vezetékek közelsége olyan problémákat okoz, mint az áthatás, a hőszórás és az elektrosztatikus kapcsolás.

Mindez indokolja a bioelektronika előtérbe kerülését. A Gorham International nemrégiben közzétett tanulmánya szerint a bioelektronikával foglalkozók száma az elmúlt öt évben egy nagyságrenddel növekedett, a befektetések értéke pedig 1990-ig a mai 100 millióról egymilliórd USA-dollárra is növekedhet.

A japánok itt is az élen járnak. A japán külkereskedelmi és iparügyi minisztérium átfogó tervet dolgozott ki a bioanyagok kutatására, és 32 millió dolláros támogatással fogja össze az különböző intézmények ez irányú kutatásait.

Újság

- közvetlen eléréssel

A New Scientist című műszaki tudományos magazin már közvetlen hozzáféréssel is elérhető a Thorn EMI Datasolve hálózaton. Az új számok — melyek terjedelmé mintegy 50 ezer szó — megjelenésük napján kerülnek az adatbázisba, ahol egyébként az 1985. szeptember 5-től kezdve megjelenő számok visszamenőleg is megtalálhatók.

Atari 520 ST

1985 őszén mutatták be a világ eddigi legolcsóbb, 32 bites számítógépet. Az Atari 520 ST valószínűleg a tavalyi év szenzációja, hiszen 750 fontért (Franciáországban 10 ezer frank, az NSZK-ban 3000 márka alatt) kapható a Motorola 68000 típusú mikroprozessor tartalmazó gép, a billentyűzet, a képernyő, az alapszoftver, sőt egy 500 k kapacitású lemez-meghajtó is.

E gép várható fő konkurense a Commodore cég által beharangozott, s az ígértek szerint Európában 1986 elején megjelenő gép, az Amiga lesz. A harc kilátásait még korai jósólni, de az Atari javára billenti a mérleget, hogy kétszer akkora, azaz 512 k memóriájú gépet kínál feleannyiért. Sokat számít a fél esztendővel korábbi piaci megjelenése is, hisz ezalatt tekintélyes szoftvervagyon halmozódhat fel az új Atari 520 ST körül.

A Quelle és a számítógép

A Quelle NSZK áruházi hálózat, melynek csak az anyaországában 25 áruház van, a külföldön létesített üzletek száma pedig meghaladja a 80-at. Éves forgalma csaknem hétélliórd márka, és a tiszta nyereség is 110 millió.

A Quelle-áruháznak széles — Magyarországra is kiterjedő — csomagküldő hálózata van. Ha bárki a közismert Quelle-katalógusok alapján bármilyen árucikket rendel, akkor a következő történik. Ha először rendel a vevő, adatait bevezetik a számítógépes nyilvántartásba, amely egyébként már 12 millió vevő adatait tárolja. A jövőben tehát már lesz egy azonosítási száma, amelyre később hivatkozhat. A rendelést a hivatkozási számot azonosító kulcsként felhasználva számítógépbe viszik, amely figyelembe véve a Quelle hétémeletes központi raktáruházában áruelhelyezését, elemlekre bontja az egyes tételeket. Ezután az előfeldolgozott rendelés egy rezkesbe kerül, és az áruház patermosztérében emeltről emelre elindul útjára. Mire a földszintre ér, már minden árucikk összegyűlt, csak a csomagolás van hátra. Ezt követően a raktári elosztó rendszeren keresztül elindul a csomag az iránykamonokban a vevő felé. A csomagküldő hálózat forgalma: évi 35 millió csomag!

Labdarúgás

1986-tól mikroszámítógép segíti a Magyar Labdarúgó Szövetség munkáját. Számítógépen tartják nyilván a NB I és NB II eseményeit, a válogatottak programját, az edzőtáborokat és nem utolsósorban a mexikói világbajnoki menetrendet.

Gépjármű-vezetés

A járművezetői jogosítvány megszerzéséhez szükséges elméleti tudnivalókat gyors és hatékony oktatására az NSZK-ból kapott megrendelés alapján programcsomagot fejlesztett ki az 5 G Kisszövetkezet. A tananyag tartalmazza a közlekedési szabályokat, műszaki ismereteket gépjármű-kategóriák szerint, sőt segítséget nyújt a hivatásos gépjármű-vezetői vizsga ismereteinek elsajátításához is.

A tananyag fejezetei témakörök szerinti bontásban oktathatók és

gyakoroltathatók. Az egyes témakörök tetszés szerinti sorrendben vehetők és ismételhetők. A program a gyakorlás során elért eredményeket az egyes tanulókra külön-külön és összesítve is nyilván tartja, és az oktató számára hozzáférhetővé teszi. Így az ismeretek el-sajátítási szintje folyamatosan ellenőrizhető.

A gépjármű-oktatási munkaközösségek számára kitűnően használható eszközt a kisszövetkezet Commodore 64 számítógépre készítette el.

A Videoton Afrikában

Irodát nyitott Algírban a Videoton. Először 1984-ben exportált képernyős terminálokat és mátrixnyomatatókat Algériába, és ezek a termékek kivitták az ottani nyelvtudományi intézet elismerését.

A Videoton az export növelése céljából hozta létre új irodáját, melynek feladata számítástechnikai és a szorokozott elektronika körébe tartozó termékek értékesítése, továbbá együttműködés szervezése. Már folynak tárgyalások licencetadásról, szállított hazai billentyűzetek algériai gyártmányú képernyős megjelenítőkhöz, és az idén különféle számítástechnikai berendezéseket is exportálnak.

Az új iroda megnyitásával a Videoton a harmadik földrészt lépett: sűrű európai képvisleti hálózatán kívül már Ázsiában, Kínában is működik bevásárlóháza.

Új adathordozó?

A közelmúltban mutatta be a japán Hudson cég a Bee-cardnak nevezett, 256 kb-át tárolókapacitású, a közönséges hitelkártyának vagy a memóriakártyának megfelelő méretű adathordozóját. Ennek azonban a hitelkártyától eltérően nemcsak olvasható, hanem írható is, változtatásai vannak. Így ez nem csupán a szoftverek forgalmazásában játszhat komoly szerepet, mint abszolút értékben is rendkívül olcsó adathordozó, hanem a memóriakártyához hasonlóan, esetleg azzal kombinálva egy sor új alkalmazás előtt nyitja meg az utat. Ezeknek a technikai realitás azonban elsősorban Japánban van meg, ahol már ma is diagnosztikai adatokat és személyi állományokat visznek ezekre a lapocskákra. Az adatok olvasása és írása az MSX szabványt kielégítő adapter segítségével történik.



NE KÍNLÓDJON TOVÁBB!

A megoldás:

PRINTER BASIC

**A MIKROSZÁMÍTÓGÉP MAGAZIN
AJÁNLATA:**



Egyszerű BASIC utasítások segítségével 480 x 392-es méretű ábrák készíthetők, majd EGYETLEN utasítással kinyomtathatók.

Ezt nyújtja Önnek a
PRINTER BASIC!

Forgalmazza:

NOVOTRADE RT 2C Számítástechnikai Áruház,
Bp. XIII., Balzac u. 35.
Pécs, Zrínyi Miklós Könyvesbolt, Jókai u. 25.
A Fotelektronik Ipari Szövetkezet szakkoltjai:
Bp. V., Múzeum krt. 19.
Bp. XIII., Rajk László u. 46/b.

Miskolc, Korvin Ottó u. 5.
Debrecen, Szabó I. altb. tér 6.
Győr, Bem tér 1.

A termék megrendelhető a következő címen:
HARDSOFT Számítástechnikai GT,
Százhalombatta, Rózsa Ferenc u. 32.

A ROBOTRON 1715 személyi számítógép egyesíti a korszerű mikroszámítógépek előnyeit a hagyományos könyvelőgépek szolgáltatásaival. A ROBOTRON kombinát új berendezése a könyvelési, számlázási munkák mellett előnyösen alkalmazható a gazdasági élet minden területén. A széles körű perifériaválaszték, a két operációs rendszer — CP/M kompatibilis, BROS (R 5110 kompatibilis) — biztosítja a mindenkori feladatnak legjobban megfelelő kiépítést.

Megrendelhető:
MIGERT
számítástechnikai
és ügyvitelgépészeti
osztály
1061 Bp.,
Népköztársaság útja 2.
Tel.: 323-332



Részletes
felvilágosítás:
MIGERT
4. sz. szakuzlet,
1081 Bp.,
Rákóczi út 57/a.
Tel.: 131-440
143-471



Alkalmazástechnikai partnereinkkel vállaljuk kulcsra kész rendszerek szállítását is.
Kedvező ár, rövid szállítási határidő.
Lízinglehetőség.

Most előnyös feltételek mellett lecserélheti ASCOTA,
SOEMTRON könyvelőgépeit.

