



# ELSŐ KÉZBŐL

A TV COMPUTERRŐL

## BILLENTYŰZET RUTINOK

A beépített rutinok leírását a billentyűzetkezelővel folytatjuk. A billentyűzet, valamint a beépített és a külső botkormányok leolvasását végzik. A használt rendszerváltozók a következők:

**PICTURE** 10 byte, címe 2897=0B51h

A billentyűzetmátrix utoljára leolvasott értéke (lásd 1. táblázat)

**OLDPICTURE** 10 byte, címe 2907=0B5Bh

A billentyűzetmátrix előzőleg leolvasott értéke

**DELAY KEY** 1 byte, címe 2917=0B65h

Az automatikus billentyűzetismétlés kezdetéhez szükséges idő 20 ms-os egységben. BASIC-ből a SET DELAY állítja.

**LOCK KEY** 1 byte, címe 2918=0B66h

Az aktuális kurzor (LOCK) állapotát mutatja:

0 normál karakterek (LOCK, normál kurzor)

1 nagybetűk (LOCK+CTRL, inverz C kurzor)

2 folyamatos shift (LOCK+SHIFT, inverz S kurzor)

8 alternatív karakterek (LOCK+ALT, inverz A kurzor)

**RATE KEY** 1 byte, címe 2919=0B67h

Automatikus billentyűismétlés alatt a két karakter megjelenése előtti időt adja 20 ms-os egységben. BASIC-ből a SET RATE állítja.

**HOLD DIS** 1 byte, címe 2920=0B68h

Jelzi, hogy a CTRL+P együttes lenyomásával felfüggeszthető-e a rendszer működése. (Pl. listázás félbeszakítható)

0 HOLD-mód engedélyezve

255 HOLD-mód tiltva

### A rutinok leírása:

**KBD IRQ** hívási kód: 144 (90h)

működés: A billentyűzetmátrix leolvasását végzi.

Felhasználói programból nem ajánlott hívni.

**KBD CHIN** hívási kód: 145 (91h)

output: C=karakter vagy botkormány kódja

A=hibakód

működés: A billentyűzetet leütött karakter, vagy a botkormány elmozdításának kódját adja. Megvárja, amíg érkezik kód.

**KBD STATUS** hívási kód: 147 (93h)

output: C=jelző

0 nincs beolvasható karakter

255 beolvasható karakter

működés: A billentyűzet, ill. botkormányok státusát adja

### A billentyűzet közvetlen leolvasása

A billentyűzet, ill. botkormányok közvetlen leolvasására is szükség lehet egyes programokban.

Az 1. táblázat megadja a billentyűzetmátrixot. Ennek leolvasásához először ki kell választani a mátrix megfelelő sorát (0-9), és ezt a 3. portra az alsó 4 biten kiküldeni. A beolvasás az 58h portról történik. Az a billentyű volt éppen lenyomva, amelyikhez tartozó bit értéke zérus.

A 3. port egyéb biteinek is van jelentése, ezért a sorkiválasztást az utoljára kiküldött byte alsó 4 bite helyébe kell írni, és úgy kiküldeni. A megfelelő rendszerváltozó:

**PORT03** 1 byte, címe 2833=0B11h

bit7, bit6: bővítőártya kiválasztás

bit3-bit0: billentyűzetmátrix sorkiválasztás

Pl. ahhoz, hogy a RETURN billentyű le van-e nyomva, az 1. táblázat alapján az 5. sor 4. bitjét kell vizsgálni. Ezt a vizsgálatot végzi el az 1. program gépi kódú rutinja. Ha visszatéréskor a ZERO FLAG=0, akkor igen, egyébként pedig nem volt a RETURN lenyomva.

Cseh Tibor

1. táblázat: Billentyűzet-mátrix a TV-Computeren.

sor	bitek							
	7.	6.	5.	4.	3.	2.	1.	0.
0	!	'	/	&	"	+	%	
1	=	ö	ó	#	ü	)	(	~
2	R	Q	@	Z	;	W	E	T
3	U	P	ü	{	ö	O	I	}
4	F	A	<	H	\	S	D	G
5	J	é	ő	RET	Á	L	K	DEL
6	V	Y	LOCK	N	SH	X	C	B
7	M	-	SP	CTRL	ESC	.	,	ALT
8			RJL	RJR	RJA	RJF	RJD	RJU
9			LJL	LJR	LJA	LJF	LJD	LJU

A rövidítések jelentése:

RET : RETURN

SP : szóköz

SH : SHIFT

\* RJL : balra

\* RJR : jobbra

RJA : gyorsítás vagy 2.tűz

RJF : tűz

\* RJD : le

\* RJU : föl

beépített (\*) és  
jobboldali külső  
botkormány

LJL : balra

LJR : jobbra

LJA : gyorsítás vagy 2.tűz

LJF : tűz

LJD : le

LJU : föl

baloldali külső  
botkormány

```

10 !
20 !-----!
30 !
40 !           TV-COMPUTER           !
50 !
60 ! RETURN billentyű figyelése     !
70 !   a billentyűzet-mátrixon     !
80 !-----!
90 !
100 !
110 ! A gépi kódú rutin:
120 !
130 !           =0B11   PORT03   EQU   0B11h
140 !
150 ! 243           F3           DI
160 ! 58,17,11     3A 11 0B     LD   A,(PORT03)
170 ! 230,240     E6 F0         AND  0F0h
180 ! 246,5        F6 05         OR   5
190 ! 50,17,11    32 11 0B     LD   (PORT03),A
200 ! 211,3        D3 03         OUT  (3),A
210 ! 219,88      DB 58         IN   A,(58h)
220 ! 203,103     CB 67         BIT  4,A
230 ! 251          FB           EI
240 ! 201          C9           RET
250 !

```

# Egy hasznos segédprogram:

# APPEND

Korábban már ismertettük a MERGE-eljárást programok összefűzésére. Az ott bemutatott módon tetszőleges programot be tudunk illeszteni a memóriában levő programunkba. Csupán egy apró kényelmetlenséggel jár a dolog: a beszúrandó programból lista file-t kell készíteni, és ezt kell szalagról leolvasni. Némi programozási „áldozattal” gyorsabb és egyszerűbb megoldást biztosít a 2. program. A listából csupán az 1. és 2. sort kell begépelni, a többi sor csak a működés könnyebb megértése miatt szerepel.

Az említett programozási áldozat a következő: a programokat úgy kell megírni, hogy az egyes részeknek különböző tartományba essenek a sorszámok. Pl. a főprogram sorszámait 100-999 tartományba esnek, a hozzá tartozó külön megírt szubrutinok pl. az 1000-1999, 2000-2999 stb. tartományt használják. Lényeges, hogy ne legyen átfedés! A használt legkisebb sorszám a 3 lehet, mert az első két sorban van az APPEND programunk. Ezek után jöhet az összefűzés,

sorszámok szerint növekvő sorrendben!

Először betöltjük az APPEND programot és elindítjuk:  
RUN

Az 1. ábra szerinti képet fogjuk látni, a kurzor a 3. sor elején villog. Két RETURN után a kurzor a 7. sorban levő LOAD elején villog. Ha szükséges, beírhatjuk a LOAD mögé idézőjelek között a betöltendő program nevét. Újabb RETURN-re a gép kiírja:

Searching

Most kell a legkisebb sorszámokat tartalmazó programrészletet betölteni. Legyen ez pl. a „FŐPROGRAM” nevű. Ha a betöltés kész (2. ábra), nyomjunk meg néhány RETURN-t, amíg az alul levő sorokat is végrehajtja a BASIC (3. ábra). Most újabb RUN paranccsal kezdve ismétlés, amíg növekvő sorszámok szerint az utolsó programrészletet is be nem olvastuk. Ezután kitöröljük az APPEND programot:

DELETE-2

és kimenthetjük SAVE paranccsal az összefűzött egész programot.

## A program működési elve

a 2. program listájában szereplő rövid gépi kódú program megkeresi az aktuális BASIC programunk végét. A BASIC program kezdőcímét a TEXT rendszerváltozó tartalmazza (címe 5922=1722h). Az első két POKE utasítás átállítja a TEXT változót, így az a programunk végére fog mutatni. Ennyi azonban nem elég a betöltés előtt, mert a LOAD vagy NEW utasítás hatására a TEXT változóba beíródik a BASIC számára használható legkisebb cím. Ezt állítja ugyancsak a program végére a második két POKE. Az említett rendszerváltozó a VLOMEM (címe 5920=1720h). Ezek után a BASIC már „nem látja” az APPEND programot, a betöltött új program fizikailag mégis ennek folytatásaként kerül a memóriába. A harmadik két POKE visszaállítja VLOMEM-et, a negyedik pedig a TEXT-et az eredeti programkezdetre. Újabb RUN-ra az eljárás ismétlődhet.

Cs. T.

```

2. PROGRAM
1 AS="" FOR I=1 TO 13: READ B: AS=AS&CHR$(B): NEXT I:USR(2+VARPTR(AS))
2 TH=INT(I/256): TL=I-256*TH:RCL=PEEK(5922):RCH=PEEK(5923):GRAPH
3 CS:PRINT"Betöltés előtt":PRINT"poke5922,";TL;"":poke592
3 TH:PRINT"poke5920,";TL;"":poke5921;"";TH
2 PRINT"load":PRINTAT18,1:"Betöltés után:"PRINT"poke5
3 RCL:"poke5921";RCH:PRINT"poke5922,";RCL;"":poke5923
3 RCH:PRINTAT2,1:"":END:DATA42,34,23,126,167,200,133,111,48,24
36,24,246
0 |--- Csak az 1. és 2. sort kell begépelni ---|
20 |-----|
30 |-----|
40 |          TV-COMPUTER          |
50 |-----|
60 | BASIC programok összefűzése |
70 |          ( APPEND )          |
80 |-----|
90 |-----|
100 |-----|
110 | A használt gépi kódú rutin. |
120 |-----|
130 |          =1722 TEXT EQU 1722h ;program eleje
140 |-----|
150 | 42,34,23 2A 22 17 LD HL,(TEXT)
160 |-----|
170 |          LOOP:
180 | 7E LD A,(HL)
190 | A7 AND A
200 | C8 RET Z ;program vége
210 | 85 ADD A,L
220 | 6F LD L,A
230 | 30 F9 JR NZ,LOOP
240 | 24 INC H
250 | 1B F6 JR LOOP

```

```

*Betöltés előtt:
ok
poke5922, 97 :poke5923, 27
poke5920, 97 :poke5921, 27
load
*Betöltés után:
poke5920, 239 :poke5921, 25
poke5922, 239 :poke5923, 25
1. ábra

```

```

*Betöltés előtt:
ok
poke5922, 97 :poke5923, 27
ok
poke5920, 97 :poke5921, 27
ok
load
Searching
Reading: FŐPROGRAM
ok
*Betöltés után:
poke5920, 239 :poke5921, 25
poke5922, 239 :poke5923, 25
2. ábra

```

```

*Betöltés előtt:
ok
poke5922, 97 :poke5923, 27
ok
poke5920, 97 :poke5921, 27
ok
load
Searching
Reading: FŐPROGRAM
ok
*Betöltés után:
poke5920, 239 :poke5921, 25
poke5922, 239 :poke5923, 25
ok
3. ábra

```