

# IDŐMÉRÉS

**PRIMO-PROGRAM**

Számítógépünk használata közben gyakran előfordulhat, hogy programunk futása közben szükségünk lenne időmérésre. Például egy programrész lefutási idejére vagy a játékos gondolkodási idejére lehetünk kíváncsiak.

Szerencsére a PRIMO készítői gondoltak erre az igényre, és a gépen megoldható az időmérés. Természetesen külön, igény szerinti programot kell készíteni hozzá.

A RAM terület 16445, 16446 és 16447 címén, három byte-on ábrázolt szám értéke 10 milliszekundumonként eggyel növekszik. A számlálás a gép bekapcsolásakor, illetve a RESET gomb benyomásakor mindig nulláról indul.

Az ábrázoláshoz tudni kell, hogy a legkisebb címen (16445) a szám első byte-ja (LSB=less significant byte), a legnagyobb pedig (16447), a szám legfelső, legértékesebb byte-ja (MSB=more significant byte) helyezkedik el.

Az itt található számot tízes számrendszerben a következő programrészlettel tudjuk kifarítani:

```
20 ? CHR$(6) : REM előtörlés
100 IDO = PEEK(16445) + 256 * PEEK(16446) +
      256 * 256 * PEEK(16447)
105 ? IDO
```

Természetesen ez az érték századszekundumokban van, hogy másodpercekben jelenjen meg:

```
10 CLS : KO = 100.2
110 IDO = IDO/KO
115 ? IDO
```

Óra, perc, másodperc megjelenítésben pedig:

```
120 OH = FIX (IDO/3600)
130 PERC = FIX (IDO/60 - 60 * OH)
140 SEC = FIX (IDO - 3600 * OH - 60 * PERC)
150 ? $ 0, 30, OH; PERC; SEC
160 GOTO 100
```

A 105. és 115. sort érdemes törölni!

Ha óránk nem járná pontosan, a 10. sorban a KO konstanszt kell tízedenként módosítani!

Lehetőség van a tényleges pontos idő beállítására is. Ekkor a következő néhány sorral kell programunkat kiegészíteni:

```
30 IF DI <> 0 THEN 100
40 INPUT "Mennyi a pontos idő (óra, perc, sec) ";
   OH, PERC, SEC
50 ID = KO * (OH * 3600 + PERC * 60 + SEC)
60 MS = FIX (ID/256/256) : MI = FIX ((ID - MS*256*256)/256) :
   LS = FIX (ID - MS*256*256 - MI*256) : DI = ID
70 POKE 16445, LS, MI, MS
```

Ezt a rövid programot, megfelelő átszámozással érdemes szubrutinként felhasználni. A jelenlegi 30-as sor

teszi lehetővé, hogy csak az első meghíváskor kelljen a pontos időt beállítani.

Végezetül óránk felhasználására nézzünk egy egyszerű játékpogramot. A működéshez nem fűzünk magyarázatot. Törje a fejét a kedves Olvasó, hogy hogyan működik, melyik programlépés milyen célt szolgál, milyen érdekességek találhatók benne!

Annyit azért elárulunk, hogy a 300. sor a kurzor (=helyjel) aktuális pozíciójának elmentését, a 380. sor a visszaírását végzi.

```
100 CLEAR 500
110 CLS
120 A = 100 : F = 999 : T = 3
130 POKE 16445, 0, 0, 0
140 ? CHR$(6);
150 ? "Gondoltam egy számot": ? "100 és 999 között.
   Találd ki!";
160 A$ = INKEY$: IF A$ = CHR$(0) THEN S$ = "":
   L = 0: ? $ T, 0, STRING$(5,32): GOTO 160
170 S$ = S$ + A$
180 IF A$ <> "" THEN BEEP 20, 10: L = L + 1
190 ? $ T, L, A$:
200 GOSUB 300
210 IF LEN(S$) < 3 THEN 160
220 ? " ";
230 S = VAL(S$)
240 SS = "": L = 0
250 IF S < A OR S > F THEN ? CHR$(7): GOTO 160
260 IF S - A <= 1 AND F - S <= 1 THEN GOSUB 400:
   GOTO 110
270 IF S - A > F - S THEN F = S: ? CHR$(131); " ennél
   kisebb"; ELSE A = S: ? CHR$(131); " ennél nagyobb";
280 IF T > 14 THEN ? ELSE T = T + 1
290 GOTO 160
300 LS = PEEK(16456) : MS = PEEK(16457)
310 ID = 256*256*PEEK(16447) + 256*PEEK(16446) +
   PEEK(16445)
320 ID = ID/100
330 OH = FIX(ID/3600)
340 PE = FIX(ID/60 - 60*OH)
350 SE = FIX(ID - 3600*OH - 60*PE)
360 ? CHR$(6);
370 ? $ 0, 30, OH; PE; SE;
380 POKE 16456, LS, MS
390 RETURN
400 IF 60*PE + SE < 60*PI + SI OR PI = 0 AND SI = 0
   THEN PI = PE: SI = SE
410 CLS
420 ? CHR$(4); " E l t a l á l t a d "; CHR$(20);
   CHR$(132); S:?: ? PE; "perc és"; SE; "másodperc alatt"
430 GOSUB 490
440 ??: " Játsszunk még (I/N) ? ";
450 J$ = INKEY$: IF J$ = "" THEN 450
460 IF J$ = "I" THEN RETURN ELSE CLS: ? "Köszönöm
   a játékot!"
470 GOSUB 490
480 END
490 ??: ? " Az eddigi legjobb idő:": ? PI; "perc és";
   SI; "másodperc": ?
500 RETURN
```

Takács Gábor  
villamosmérnök

★★★