

- egy hexadecimális szám előtt nincs vezető nulla
- egy címkét többször is definiáltunk, és az első előfordulás nem DEFL utasításban van. A második előfordulásnál jelzi ki a hibát.

NO END INSTR. - nincs END a programban

MEMORY FULL - nincs lehetőség a pufferek felülírására

**Az EDITAS és a SPECTRUM MONITOR együttes alkalmazása:**

Amennyiben rendelkezésünkre áll a SPECTRUM MONITOR (Picturesque/1982) hibakereső és disassembler program, célszerű előbb azt (a 16K-s változatot) betölteni, majd az EDITAS-t (a 48K-s változatot). A SPECTRUM MONITOR az EDITAS-ból az M-paranccsal hívható meg, míg a SPECTRUM MONITOR-ból előbb Y-al ki kell lépni BASIC-be, és a megfelelő RANDOMIZEUSR utasításra az EDITAS ismét aktivizálható.

### Gens-3 Editor/Assembler

HISOFT/1983

Betöltése a LOAD""CODE xxxxxx utasítással történik, ahol xxxxxx tetszőleges RAM cím, de lehetőleg a RAM elején kell elhelyezni, mert mögé kerül a forrásszöveg és a szimbólum-tábla is. Első indítás esetén használjuk a RANDOMIZEUSR xxxxxx utasítást, újraindítás esetén pedig:

```
RANDOMIZEUSR (xxxxxx+56) - hidegindítás
RANDOMIZEUSR (xxxxxx+61) - melegindítás
```

Hidegindításnál törlődik a forrásszöveg és a címtáblák, míg melegindításnál nem.

Az első kérdés a "Buffer Size ?", amelyre a válasz "m", ahol  $0 < m < 9$ , és az alapértéke  $m = 4$  (elég ehhez megnyomni az ENTER-t). Ezek a pufferek 64 byte-osak és a forrásszöveg beolvasásához is szükségesek lehetnek.

Fontos, hogy a program interruptot (megszakítást) nem engedélyez,

így ha visszatérünk a BASIC-hez, arról nekünk kell külön gondoskodni.

A "Table size ?" (az A - EDITOR parancsnál) a szimbólum-tábla nagyságára kérdez rá. Elég itt is, ha csak az ENTER-rel válaszolunk.

Assembláláskor különböző assemblálási kód közül választhatunk. A következő lista a lehetséges opciókat közli. Az itt látható opciókhoz tartozó kódok összegét kell az "option?" kérdésre megadni:

- 1 - az assemblálás végén nem kapunk szimbólum-táblát
- 2 - az assemblálás csak ellenőriz és nem állít elő tárgykódot
- 4 - az assemblálás végén nem kapunk assembly listát
- 8 - az assemblálás végén az assembly listát a Printerre küldjük
- 16 - az assemblálás után a tárgykódot elhelyezi a memóriában
- 32 - ellenőrzés nélküli assemblálást hajt végre, így a művelet valamivel gyorsabb lesz

pl. a 36-os kód megadása jelenti a leggyorsabb assemblálást.

A forrásszöveg sorai a BASIC-hez hasonlóan sorszámot kapnak. A bináris konstansok %-al kezdődnek ( pl. %101 =5), a címkek pedig betűvel (pl. L23).

A címszámlálóra való hivatkozásnál \$ (dollár) -jelet használunk. A műveleteknél a modulus függvényre "?"-jellel hivatkozunk, a logikai vagy (OR) függvényre "@" (kukac) jellel, míg a logikai kizáró vagy (XOR) függvényre "!" (felkiáltó)-jellel.

Assembler direktívák:

ORG n - a tárgykód kezdőcímét jelöli ki. Előtte csak comment (megjegyzés) sorok állhatnak. Az n lehet kifejezés is.

EQU - az EQU kifejezést címkének kell megelőznie. értékadó utasításnak számít.

DEFB kifejezés, kif., kif., ...  
- minden kifejezést 8 bitre értékel ki és konstansként helyezi el.

DEFW kif., kif., ...  
- a fentivel szemben most 2 byte-on teszi le a kifejezést.

- DEFS kif. - a kifejezésnek megfelelő számú byte-ot foglal le.
- DEFN "string"  
- stringet szűr be, max 256 byte hosszon.
- IF kifejezés  
- ha a kifejezés = 0, akkor a forrásszöveg sorokat nem veszi figyelembe, amíg nem találkozik az ELSE vagy az END utasításokkal.
- ELSE - lezárja az IF vezérlését.
- END - lezárja az IF vezérlését és visszakapcsolja az assemblert

## Az assembler parancsai

Az assembler direktívákhoz hasonlóan nincs közvetlen hatásuk a tárgykódra, csupán az assembly lista formáját módosítják. E parancsok a forrásszöveg-sor elején egy "\*" (csillaggjel)-el kezdődnek, ezután betű (nagybetű) következik, és ezt esetleg "+" vagy "-" jelek követhetik. A sor hátralevő része tetszőleges lehet, kivéve az L+, L-, D+, és D- parancsokat. A következő parancsok érhetők el:

- \*E - (eject) három üres sort emel a képernyőn vagy a printeren
- \*Hs - (head) ua. mint a \*E és még az s stringet kinyomtatja.
- \*S - eddig a sorig listáz, a listázás egy tetszőleges billentyűvel folytatható, viszont idő előtt leáll, ha \*L vagy \*S parancsokkal találkozik.
- \*L - e sortól kezdve nem listáz.
- \*L+ - e sortól kezdve listáz.
- \*D- - a címszámlálót ezután HEX számrendszerben írja ki.
- \*D+ - a címszámlálót ezután DEC számrendszerben írja ki.

- \*C- - a sor után rövidített assemblerlistát ad.
- \*C+ - a sor után a teljes assemblersort kiírja.
- \*F - kazettás magnetofonról beolvas egy forrásszöveg fi-  
le-t, amelyet assemblálhatunk, s így a szöveg csak  
minimális helyet foglal le a memóriában, amelyet  
kisebb adagokban olvashatunk be. A betöltött  
szövegfile előzőleg a T paranccsal mentettük ki  
a "Buffer size ?"-nak megfelelően.

Ekkor a szövegfile blokkokban helyezkedik el, és ami-  
kor F paranccsal beolvasunk egy blokkot, akkor azt a-  
zonnal le is fordítjuk. A blokkok beolvasása mindaddig  
tart, amíg be nem olvastuk valamennyi blokkot.

Megjegyzés: a P parancs erre nem jó !

Ha a forrásszöveg nagyon hosszú, akkor először mindig  
vigyük ki azt a T paranccsal, majd olvassuk vissza F-  
el kisebb blokkokban.

#### Az EDITOR parancsai

A GENS-3 indítás után az EDITOR parancsait várja. A további-  
akban a SYMBOL SHIFT-et SS-el, a CAPS SHIFT-et CS-el rövidítjük.

Az editor a parancskéréshez a ">" jelet használja.

- I n,m - sorokat szűrhatunk be az n sortól kezdve m lépték-  
kel. Kilépés a CS+1-re lehetséges. A már létező sor-  
számú sorokat törli (felülírja). Egy szövegsor 64  
karakter lehet (buffer size).
- L n,m - listáz az n sortól az m sorig. Kilépés a CS+1-el  
lehetséges.
- K n - listázáskor n sort jelenít meg.
- D (n,m) - sorokat töröl az n sortól m sorig. Egyetlen sor  
törlése az m = n-el lehetséges.
- M n,m - az n sort az m-be mozgatja át, ha m létezik felül-  
írja azt.
- N (n,m) - átsorszámozza a sorokat. Az első sor sorszáma n, a  
lépésköz pedig m.
- F n,m,f,s - keres egy f stringet az n és m sorok között. Ha ta-

lál illet, leáll. Amennyiben s-t is megadjuk, a megkeresett f string felcserélhető a megadott s stringre az E (EDIT) paranccsal. Ha tovább akarunk keresni, ismét az F parancsot kell megadni.

- E (n) - az n sorszámú sort átmásolja a pufferba, ahol tet-  
szés szerint módosítható, de szükség esetén előhív-  
ható az eredeti sor is.  
Az E (EDIT) parancsnak több alparancsa is van, me-  
lyek a CS-el aktivizálhatók.  
Pl. CS " " (szóköz) esetén meghagyja az eredeti ka-  
raktert, azaz megnöveli a soron belüli pointert, e-  
gészen a sor végéig.
- CSO - karaktert töröl a BASIC-hez hasonlóan.  
CSB - a szöveg-pointert egy tab-pozícióval na-  
gyobbra állítja.  
ENTER - sor kész  
Q - kilép az EDIT-ből és törli a változtatáso-  
kat.  
R - újra betölti az adott sort a pufferba, mi-  
közben a változtatásokat elfelejti.  
L - a sor hátralevő részét meghagyja és a poin-  
tert a sor elejére állítja.  
K - törli az adott karaktert.  
Z - törli az összes karaktert a pillanatnyi po-  
ziciótól kezdve a sor végéig.  
F - a korábban megadott stringet keresi. Ha i-  
lyet nem talál kilép az EDIT módból.  
S - a korábban megadott stringet lecseréli a  
már megadott stringre.  
I - az adott pozíciótól kezdve karaktereket  
szűr be. Kilépés ENTER-rel lehetséges, ezen  
belül a BASIC-hez hasonlóan működik a CSO  
és a CSB is.  
X - a sor végére állítja a pointert és a BASIC-  
hez hasonlóan szerkeszthető a sor.  
D - felülírható az adott karakter. Kilépés az  
ENTER-rel lehetséges, valamint használhatók  
a BASIC szerkesztő utasításai.

#### Adathordozó-kezeléssel kapcsolatos parancsok

- F n,m,s - forrásszöveg-file-t visz ki szalagra az n-sorszámú  
sortól kezdve az m-sorszámú sorig, s néven.
- G s - egy s file-t keres a szalagon és betölti azt. Ha  
már van szövegfile a memóriában, akkor e mögé tölti  
be szalagról miközben a teljes szöveget átsorszámoz-  
za.

T n,m,s - a szövegfile-t blokkokban viszi ki az n sortól kezdve m-ig, s néven. A blokkméretet a "Buffer size?" -nél adtuk meg. A \*F parancsnál olvassuk majd vissza a forrásszöveg-file-t.

Az EDITOR parancsai assembláláshoz és futtatáshoz

A - assemblálás végrehajtása (a forrásszöveg tárgykóddá alakul)

R - az ENT-nek megfelelő címtől futtatja a tárgykódot, amely RET utasítással végződik (ENT= belépési pont a rutinba, melynek értékét az ORG-al határoztuk meg).

Egyéb parancsok

B - visszatérés a BASIC rendszerhez.

C - lehetőség van régebbi GENS programok által írt szövegfile-ok átkonvertálására a GENS-3 -hoz. Ekkor G-vel beolvasunk egy pl. GENS-1-el írt szövegfile-t, majd a C paranccsal a GENS-3-nak megfelelő formára alakítjuk.

d - a delimiter (határolójel) megváltoztatható a d string első karakterével.

p - kinyomtatja az alapértelmezéseket, ahol a sorszámok intervalluma 1-32767 lehet, míg a stringhosszak max. 20 karakteresek.

W (n,m)- kinyomtatja az n és m közé eső sorokat. A K értékének megfelelően bizonyos számú sor kinyomtatása után szünetet tart, tetszőleges billentyű megnyomása után folytatja a műveletet.

X - kinyomtatja a kezdő és a végcímet. Ily módon lehetőség van arra, hogy pl. MONS-3 által létrehozott szövegfile-t a GENS-3 editora átvegyen.

A GENS-3 által használt szövegfile-t az xxxxx+54 és xxxxx+55 végcímek határozzák meg. Tehát, ha egy szövegfile a RAM-ban van, majd betöltjük a GENS-3-at, és BASIC-ből kiadjuk a  
 POKE xxxxx+54,végcím (alsó byte) és  
 POKE xxxxx+55,végcím (felső byte) akkor  
 RANDOMIZE USR xxxxx-től nem jó, hanem  
 RANDOMIZE USR xxxxx+61 utasítással a szövegfile a GENS-3-é!!!

## Hibakódok:

1. hiba ebben a sorban van
2. a mnemonikot nem ismeri fel
3. hibás formátumú utasítás
4. a szimbólumot többször definiáltuk
5. a sorban illegális karakter van
6. az operandusok egyike illegális
7. a szimbólumként használt szó "foglalt"
8. tiltott regiszterhasználat
9. sok a regiszter
10. a 8 bites kifejezés többre sikerült mint 8 bit
11. a JP(IX+n) és JP(IY+n) tiltott
12. az assembler direktíva formája hibás
13. illegális hivatkozás, pl. az EQU-ban olyan változó szerepel, amely még nem kapott értéket
14. osztás zérussal
15. szorzásnál fellépő túlcsordulás

BAD ORG ! - hibás ORG cím fordul elő

Out of Table space ! - a szimbólum-tábla számára kevés a hely

Bad Memory ! - nincs több hely a szövegfile részére

## Zeus Assembler

CRYSTAL COMPUTING LTD./1983

Betöltés után megjelenik a copyright felirat és egy villogó kurzor.

## Parancs lista:

- A x - ASSEMBLE  
a forrásszövegből tárgykódot állít elő (hiba esetén hibaüzenetet jelez ki).
- D xy - DELETE  
megadott x sortól y sorig törli a forrásszöveget.
- F "string" xyz  
- FIND  
keresi x és y között a string z számú előfordulását.