

KKVMF

Matematikai és Számítástechnikai Intézet .

EDI 80

Z-80 EDITOR, ASSEMBLER, MONITOR a HT-1080Z

iskolaszámítógépre

(1984. 10. 30.)

Összeállította: Tick József
Tiszai Tamás

Budapest, 1984

Előszó

A leírás, melyet a kezében tart, azt a célt szolgálja, hogy segítséget nyújtson az EDI-80 program kezeléséhez. Nem tűzi ki célul, hogy megismertessen a gépi kódú programozás részleteivel, a Z-80 mikroprocesszor hatékony felhasználásával. Az erre vonatkozó ismereteket egyéb forrásból kell megszereznie. Megtalálja viszont e leírásban az EDI-80 valamennyi parancsának részletes leírását, az ASSEMBLER szintaktikájának ismertetését, valamint néhány - alapvető fontosságú - HT-1080-Z ROM - rutin belépési címét és paramétereit, melyek megkönnyítik, hogy az EDI-80 segítségével működőképes gépi kódú programokat hozzon létre.

Budapest, 1984.

A szerzők

Tartalomjegyzék

	Oldal
Bevezetés	1.
Az EDI-80 betöltése/indítása	2.
Az EDI-80 által értelmezett parancsok	3.
Editor parancsok	4.
Pointer mozgató parancsok	4.
Szövegbeillesztő parancsok	5.
Szövegtörlő parancsok	7.
Kazetta kezelő parancsok	7.
Egyéb parancsok	10.
Assembler parancsok	11.
Monitor parancsok	14.
Az ASSEMBLER szintaktikája	21.
Címke mező	22.
Utasítás mező	22.
Operandus mező	23.
Megjegyzés mező	25.
Az EDI-80 ASSEMBLER direktívái	26.
ORG Origin	26.
DS Define Storage	26.
EQU Equate	26.
DB Define Byte	27.
DW Define Word	27.
LOAD	28.
END	29.
Az ASSEMBLER hibajelzései	30.
Az EDI-80 parancsok összefoglalása	31.
A HT-1080 Z klaviatúra- és display rutinjai	32.
Gépi kódú rutinok hívása BASIC-ből	34.

BEVEZETÉS

Az EDI-80 egy kazettáról betölthető, gépi kódú program, mely lehetővé teszi, hogy a HT-1080Z felhasználója assembly nyelvű programokat írjon, szerkesszen, fordítson és módosítson a Z-80 mikroprocesszor assembly szintű nyelvén.

A program három részre tagolható:

EDITOR - lehetővé teszi assembly nyelvű programok begépelését, a tárban lévő program kazettára mentését, a kazettán tárolt program tárba töltését, valamint - különféle parancsokkal - a tárban őrzött program szerkesztését, módosítását.

ASSEMBLER - elkészíti a tárban lévő assembly nyelvű program gépi kódú megfelelőjét és azt a képernyőre, printerre, ill. kazettára listázza. A programban Z-80 utasítások, direktívák, szimbólikus, decimális, hexadecimális és oktális operandusok, illetve ezekből műveleti jelekkel képzett kifejezések, valamint megjegyzések állhatnak.

Az assembler jelzi a fordítás során észlelt hibát, valamint - kívánságra - kilistázza a felhasználói szimbólumok táblázatát, vagy annak kiválasztott részét.

MONITOR - alkalmas a tár kijelölt részének módosítására, a Z-80 belső regisztereiben őrzött adatok, illetve a tár tartalmának megjelenítésére, gépi kódú programkazettáról a tárba töltésére, a program indítására.

Az EDI betöltése/indítása:

Az EDI-80 gépi kódú program. Ezért ezt nem a CLOAD, hanem a SYSTEM parancs segítségével tölthetjük be a tárba. A betöltés menete a következő:

Gépelje be a SYSTEM parancsot, majd nyomja meg a NEW LINE billentyűt. A HT-1080Z által megjelölt $\#?$ után írja le az EDI programnevet, majd ismét üsse le a NEW LINE billentyűt. Ekkor megkezdődik a program betöltése, amely kb 1,5 percet vesz igénybe. Ezalatt a képernyő jobb felső sarkában kb 1/4 Hz frekvenciával egy $\#$ karakternek kell villognia. Ha a karakter nem villog, vagy a helyén egy C betű jelenik meg, a betöltést a RESET-gomb segítségével meg kell szakítani, és a műveletet a SYSTEM parancs megadásától újból kell kezdeni. /Ebben az esetben ne feledkezzen meg a kazetta visszacsévéeléséről!/
Ha a program beolvasása sikeresen lezajlott, a képernyő következő sorában ismét megjelenik a $\#?$. Ekkor a /, majd NEW LINE gombok lenyomásával a program elindítható.

Indítás után - és általában, ha a program képes új parancs elfogadására - a képernyő következő sorának elején a $?>$ készenléti jel jelenik meg /ügynevezett prompt/. Amennyiben az EDI-80 bármikor "lefagy" /azaz nem jeleníti meg a promptot/, a RESET gomb megnyomása, majd a SYSTEM parancs begépelése és a $\#?$ -re adott /17152 válasz után a program újra indítható. Ha ez sem vezetne eredményre, a programot ismét be kell tölteni a kazettáról.

Az EDI-80 által értelmezett parancsok.

Az EDI-80 parancsai - két kivételtől eltekintve /RO, WO/ - egybetűsek. A parancsokat mindig a prompt /?>/' után kell begépelni és a NEW LINE billentyűvel kell lezárni. A parancs lezárása - a NEW LINE megnyomása - előtt az esetleges gépelési hibák a ← billentyű használatával még törölhetők. A prompt után közvetlenül lenyomott NEW LINE képernyő törlést eredményez. Ha a rendszer nem képes a parancsot értelmezni, a HUH? hibaüzenet jelenik meg. Néhány parancs után opcionálisan numerikus operandus is megadható. A numerikus operandusokat decimális, hexadecimális /a szám végén H betű/, vagy oktális /a szám végén O betű/ számrendszerbeli számként adhatjuk meg. A program - amennyiben a beírt szám decimális megfelelője nagyobb, mint 65535 - a szám modulo 65536 értékét használja. Az alábbiakban három csoportra - EDITOR, ASSEMBLER, MONITOR - bontva ismeretjük az EDI-80 parancsait. Az egyes parancsok írásmódjának megadásánál a következő jelöléseket alkalmazzuk:

Nagybetű - maga a parancs név

- n - opcionálisan megadható numerikus operandus; ha nem adjuk meg, úgy a rendszer egy feltételezett értékkel dolgozik, amely a D, P, U, Z parancsok esetén = 1, míg az M és Q parancsok esetén a hasonló parancsokban utólsónak előfordult operandus értékkel egyenlő.
- sz - kötelezően megadandó, minimálisan 1, maximálisan 59 karakter hosszú karakterfüzér.
- b - opcionálisan megadható betű.

ZGR

112
14/1/80

D4

4.3.

EDITOR parancsok

Az EDITOR parancsaival az EDI-80 szövegpufferében tárolt forrásprogram - assembly nyelvű program - módosítása, tárolása végezhető el. Az EDITOR sor-orientált, ami azt jelenti, hogy a legkisebb módosítható egység a sor /vagyis egy soron belüli hiba csak a sor újra írásával javítható/. A szövegpufferben lévő sorok közül minden pillanatban létezik egy olyan, amelyet aktuális - kurrens - sornak nevezünk. A kurrens sor az a sor, amelyre az EDITOR pointer - mutatója - éppen rámutat. Az EDITOR működése során az egyes sorokat sorszámmal látja el. Ez a sorszám csak a szövegpufferben belüli könnyebb tájékozódást szolgálja, ténylegesen nem része a sornak.

Az EDITOR parancsai a következők:

Pointer - mozgató parancsok:

B - BOTTOM A pointer-a szövegpuffer utolsó sora mögé áll. Megadásakor - jelezve, hogy a pointer a szövegpuffer végén áll - az EDITOR az EOF üzenetet jeleníti meg.

ln
Dn - Down A pointer n sorral elmozdul a szövegpuffer vége felé. Ha az n értékét nem adtuk meg, úgy azt a rendszer l-nek tekinti. A parancs végrehajtása után a képernyőn megjelenik az új aktuális sor. Ha a szövegpuffer végéig nincs már n darab sor, a parancs úgy működik, mintha a B parancsot adtuk volna meg.

Lsz- Locate A parancs segítségével - az aktuális sor és a szövegpuffer vége közti területen - az sz karaktersorozat első előfordulása kereshető meg. Találat esetén az aktuális sor a keresett karaktersorozatot tartalmazó sor lesz, amely meg is jelenik a képernyőn. Sikertelen keresés esetén a képernyőn az EOF hibajelzés jelenik meg és az L parancs hatása azonos

LTOBB!

nincs más

↑, ↓
azzal, mintha a B parancsot adtuk volna meg.

*listát kezelni
konkret editor
...
... lehet korai*
Pn - Print A parancs az aktuális sortól kezdődő n darab sort a szövegpufferből a képernyőre listázza. Megjelenítés után az új aktuális sor az utolsóként kilistázott sor lesz. Ha n értékét nem adtuk meg, úgy a rendszer 1-nek tekinti. Amennyiben a szövegpuffer végéig nincs már n darab sor, a listázás után - jelezve, hogy a szövegpuffer végén áll - az EDITOR az EOF üzenetet jeleníti meg.

T - TOP A pointer a szövegpuffer első sorára áll./a szövegpuffer első sora lesz az aktuális sor/.

*↑
n*
Un - Up A pointer n sorral elmozdul a szövegpuffer eleje felé. Ha n értékét nem adtuk meg, úgy azt a rendszer 1-nek tekinti. A parancs végrehajtása után a képernyőn megjelenik az új aktuális sor. Ha a szövegpuffer elejéig nincs már n darab sor, a parancs úgy működik, mintha a T parancsot adtuk volna meg.

Szövegbeillesztő parancsok:

E - Enter A parancs segítségével új sort/sorokat illeszthetünk az aktuális sor elé. A parancs lezárása után a képernyő következő sorának elején megjelenik az aktuális sor sorszáma. Ezután gépelhetjük be az új sort. A sor maximálisan 59 karakter hosszú lehet, az 59. karakter megadása után a rendszer újabb karaktert már nem fogad el. Az esetleges gépelési hibát a ← gomb segítségével javíthatjuk. A bebillentyűzött sort a NEW-LINE billentyűvel zárhatjuk le, ekkor megjelenik a következő sorszám, amely mögé újabb sort gépelhetünk, és így tovább...

• KARAKTER
KILÉP

Amennyiben nem kívánunk további sorokat a szövegpufferbe illeszteni, úgy a sorszám megjelenése után a . és a NEW LINE billentyűket nyomjuk meg. Ekkor befejeződik az E parancs végrehajtása és megjelenik a ?> prompt.

Az EDITOR szövegpuffere több, mint 10.000 karakter tárolására képes, mégis előfordulhat, hogy - igen nagy program esetén - betelik. Ebben az esetben az utoljára begépelte sor beírása után nem a következő sorszám, hanem a MEMORY FULL hibáüzenet jelenik meg a képernyőn, és az E parancs végrehajtása befejeződik. /Ilyenkor abban sem lehetünk biztosak, hogy az utoljára begépelte sor teljes egészében bekerült a szövegpufferbe, ezért a puffertartalom képernyőre listázásával ellenőrizni kell a tényleges állapotot!/
N - NEW

Ezzel a paranccsal kettős feladatot hajthatunk végre. A parancs megadása után az EDI-80 kitörli a szövegpufferből az aktuális sort, majd úgy működik, mintha E parancsot adtunk volna meg. Az első sor lezárása után azonban nem a következő sorszám, hanem a ?> prompt jelenik meg, mivel az N paranccsal - az E parancstól eltérően - csak egy sor illeszthető a kitörölt régi helyére.

MEMORY FULL üzenet megjelenése a szövegpuffer beteltét jelenti, a teendő azonos az E parancsnál leírtakkal.

Szövegtörlő parancsok:

- K - Kill A parancs megadása a szövegpufferben őrzött összes sort kitörli.
- Z - Zap Ezzel a paranccsal az aktuális sortól kezdődő n darab sort törölhetjük a szövegpufferből. Ha n értékét nem adtuk meg, úgy azt a rendszer 1-nek tekinti. A parancs végrehajtása után a képernyőn megjelenik az új aktuális sor /az első már nem törölt sor/. Ha a szövegpuffer végéig nincs már n darab sor, a parancs úgy működik, mintha a B parancsot adtuk volna meg.
/ Ha a szövegpufferben igen sok sor van és a Z parancs numerikus operandusa viszonylag nagy - pl. több, mint 50 - a parancs végrehajtása 1/2 - 1 percet is igénybe vehet. Ilyenkor türelmesen várjuk ki a ?>prompt megjelenését!/

Kazetta kezelő parancsok:

A kazetta kezelésénél ne feledkezzünk meg arról, hogy beolvasás előtt a PLAY, felvétel előtt a RECORD és PLAY gombot be kell nyomni! A kazettás egység bal oldalán található Fl jelű gombot hozzuk olyan helyzetbe, hogy a gomb melletti lámpa ne világítson /a kazetta indítását, vagy megállítását az EDI-80 vezérelhesse/.

- R- Read Kazettán tárolt assembly nyelvű program /forrásprogram/ szövegpufferbe töltését végrehajtó parancs. Megadása után a rendszer a NAME> üzenet megjelenítését követően a kazettán rögzített program nevének begépelésére várakozik. Miután pontosan beírtuk a nevet és megnyomtuk a NEW LINE billentyűt, elindul a magnetofon és az EDI-80 az adott nevű

forrásprogram elejét keresi. Ha megtalálja, a képernyőre egy L betűt /load/ ír ki, majd megkezdí a betöltést - a pointer helyzetétől függetlenül - a szövegpufferben őrzött utolsó sor mögő./Az R parancs végrehajtása nem változtatja meg a pointer helyzetét/ A betöltés végeztével megáll a magnetofon és a képernyőn megjelenik a szövegpuffer hexadecimális kezdő és végcíme /lásd H parancs/. Amennyiben a beolvasás alatt hiba történik, a képernyőn egy, -vagy több - * karakter jelenik meg. Ebben az esetben - a RESET gomb megnyomásával - állítsuk meg az EDI-80 működését, majd az újraindítást követően ismét kíséreljük meg a forrásprogram betöltését /ne feledkezzünk meg a kazetta visszacsóvéléséről!/
Amennyiben a NAME > üzenet után csak a NEW LINE billentyűt nyomjuk meg - nem adunk meg nevet - az EDI-80 a kazettán soronkövetkező forrásprogramot tölti be a szövegpufferbe. Ha a megadott név nem azonos a kazettán következő forrásprogram névvel, a rendszer átlépi ezt a programot és a keresést tovább folytatja.

W - Write

A Write paranccsal a szövegpuffer tartalmát lehet későbbi felhasználás céljára a kazettára kiírni. Megadását követően a rendszer a NAME > üzenet megjelenítése után a program nevének begépelésére várakozik. A név maximum 30 karakter hosszúságú, tetszés szerinti karaktereket tartalmazó szöveg lehet.

Ha 30 karakternél hosszabb nevet adunk meg, a rendszer a NAME\ prompt ismételt megjelenítése után a korrekt név megadására várakozik. /Az R és V parancsok esetén a nevet ugyanolyan formában kell megadni, mint ahogy azt a W parancsnál begépeljük. Az EDI-80 különbözőnek tekinti a kis, illetve nagybetűket is./ Miután leirtuk a nevet és megnyomtuk a NEW LINE billentyűt, elindul a magnetofon. Az egyes programok könnyebb elválasztását szolgáló rövid csend szalagra rögzítését követően megkezdődik a program nevének, majd a szövegpuffer tartalmának kiírása. A W parancs a szövegpuffer teljes tartalmát - az első sortól az utolsóig - kiírja a magnóra. A rögzítést követően megáll a magnetofon és megjelenik a ?> prompt. A W parancs végrehajtása nem változtatja meg a pointer helyzetét.

V - Verify

Ezzel a paranccsal állapíthatjuk meg, hogy sikeres volt-e a forrásprogram kazettára írása. A parancs megadása után a rendszer a NAME\ üzenet megjelenítését követően a rögzített forrásprogram nevének begépelésére várakozik. /A név megadásának szabályai azonosak a W parancsnál leirtakkal/. Miután begépeljük a nevet és megnyomtuk a NEW LINE billentyűt, elindul a magnetofon és az EDI-80 az adott nevű forrásprogram elejét keresi. Ha megtalálta, a képernyőre egy L betűt /LOAD/ ír ki, majd megkezdí a forrásprogram ellenőrzését. A vizsgálat a kiirt ellenőrző összegek és a beolvasáskor képzett ellenőrző összegek összehasonlítására épül, így nem szükséges, hogy a szövegpufferben az a program legyen, amelyet ellenőrzünk.

Amennyiben az ellenőrzés hibát észlel, a képernyőn egy - vagy több - x karakter jelenik meg. Ebben az esetben kíséreljük meg ismét a betöltést /az R parancsnál ismertetett módon/. Ha ezután is hibás a beolvasás, a megoldást csak a program újabb szalagra rögzítése jelent. Amennyiben a NAME > üzenet után csak a NEW LINE gombot nyomjuk meg - nem adva meg a nevet - az EDI-80 a kazettán soronkövetkező forrásprogramot ellenőrzi. Ha a megadott név nem azonos a kazettán következő forrásprogram névvel, a rendszer átlépi ezt a programot, és a keresést tovább folytatja.

Egyéb parancsok

H - HOW A parancs megadása után a képernyőn megjelenik a szövegpuffer hexadecimális kezdő-, és aktuális végcime. Ennek segítségével ellenőrizhetjük, hogy még hány karakternyi szöveget illeszthetünk a szövegpufferbe, és pillanatnyilag hány karakter van a szövegpufferben. /A maximális végcim értéke 7FFFH lehet/.

ASSEMBLER parancsok

Az ASSEMBLER segítségével az EDI-80 szövegpufferében tárolt forrásprogram gépi kódra fordítása - assemblálása - hajtható végre. Az EDI-80 ASSEMBLERE a szabványos Z-80 utasítás,-feltételkód- és regiszterneveket - memonikok - ismeri. A programban direktívák - az ASSEMBLER működését befolyásoló utasítások - is alkalmazhatók. /A forrásprogram részletes formai szabályait "Az ASSEMBLER szintaktikája" c. fejezet tartalmazza/.

- A - Assemble A parancs megadása után az EDI-80 a szövegpufferben lévő forrásnyelvi programot - az első END direktíváig - átalakítja tárgyprogrammá /gépi kóddá/. Miután megadtuk az A parancsot, a rendszer az OPTIONS> üzenet megjelenítése után a periféria kijelölésre várakozik. A kijelölhető készülékek a következők:
- V - a fordítási lista a képernyőre készül. Az ASSEMBLER megjeleníti a lista első 15 sorát, majd rövid ideig várakozik. Ezalatt bármely gombot megnyomva a lista bármelyomb újrainyomásáig a képernyőn marad. A várakozási idő letelte után - vagy az engedélyező gombnyomás után a képernyőn a következő 15 sor lesz látható és a leirtak addig ismétlődnek, míg a teljes programlista meg nem jelenik.
 - E - a fordítási lista a sornyomtatón

készül. Ha a számítógéphez nincs nyomtató kapcsolva, az EDI-80 "lefagy" /csak a RESET gomb marad hatásos/.

- C - az EDI-80 tárgy kódú programot ír a kazettára. Ha a C opciót adjuk meg, lista nem készül, a képernyőn az EXEC ADDR> üzenet jelenik meg, melyre válaszul a gépi program indítási címét kell megadni. /Ha nem adjuk meg az indítási címet - az üzenet után a NEW LINE billentyűt nyomjuk meg - a tárgyprogram kazettáról végrehajtott betöltése után - RO parancs - az indítást végző G parancsban a címet feltétlenül meg kell adni, különben annak értéke kiszámíthatatlan!/ Ez után a NAME> üzenet jelenik meg. Most adhatjuk meg a tárgyprogram nevét. A név maximum 59 karakter hosszúságú, tetszés szerinti karaktereket tartalmazó szöveg lehet.

Miután begépeztük a perifériát kiválasztó betűt, nyomjuk meg a NEW LINE billentyűt. Ekkor megkezdődik a forrásprogram assemblálása, melynek során az ASSEMBLER megvizsgálja, hogy az egymást követő forrás-sorok hibátlanok-e, majd a kiválasztott perifériára elkészíti a fordítási listát, ill. a tárgy kódot. /Függetlenül attól, hogy a szövegpuffer pointer hol áll, az ASSEMBLER a fordítást a szövegpuffer el-

ső sorától kezd. / Ha az ASSEMBLER valamelyik sorban hibát észlel, megszakítja a további assemblálást, és a megfelelő hibajelzés után a képernyőn kilistázza a hibás sort. Ezzel egyidőben a szövegbuffer pointerét a hibás sorra állítja, megkönnyítve így a hiba kijavítását. /Ha a program hibátlan, a szövegbuffer pointer az END direktívát tartalmazó sor mögé mutat./

Sb - Sort Az S parancs segítségével az ABC szerint rendezett szimbólumtáblát - a programban alkalmazott címkeket, szimbólumokat és a hozzájuk rendelt hexadecimális értékeket - írathatjuk ki. /Ha a parancs neve után begépelünk még egy betűt, úgy a szimbólumtáblában csak ezen másodiknak begépelte betűvel kezdődő felhasználói szimbólumok fognak megjelenni./

A parancs megadása után a rendszer a OPTIONS> üzenetet írja ki és a periféria kijelölésre várakozik. A lehetséges készülékek a következők:

- V - a szimbólumtábla a képernyőn jelenik meg.
- E - a szimbólumtábla a nyomtatón jelenik meg.

/Ha a számítógéphez nincs nyomtató kapcsolva, az EDI-80 "lefagy"./

A perifériát kiválasztó betű begépelése után nyomjuk meg a NEW LINE billentyűt. Ekkor megkezdődik az assemblálás és a szimbólumtábla összeállítás, majd listázása. Ha az ASSEMBLER valamely sorban hibát észlel, annak következménye azonos az A parancsnál leírtakkal.

MONITOR parancsok

A MONITOR parancsaival a gépi kódú programok futtatása, módosítása valósítható meg. Ügyeljünk arra, hogy az EDI-80 rendszerváltozóit a gépi szintű program "belövése" alatt ne változtassuk meg, mert ez a rendszer hibás működéséhez vezet!

C - Copy

A parancs megadásával a tár kiválasztott területét egy másik tárterületre másolhatjuk. Gépeljük be a C betűt, majd nyomjuk meg a NEW LINE billentyűt. A megjelenő START > üzenet után gépeljük be a másolandó terület kezdőcímét és zárjuk le azt a NEW LINE billentyű megnyomásával. Ekkor megjelenik a STOP > üzenet. Most a másolandó terület végcímét adjuk meg, majd nyomjuk meg a NEW LINE billentyűt. /Ha a STOP > kérdésre adott paraméter-érték kisebb, mint a START - paraméter, a HUH? hibajelzés után az EDI-80 új parancsra vár./ Ezután a DEST > üzenet /Destination, cél/ jelenik meg, és az EDI-80 arra vár, hogy megadjuk azt a címet, ahová a korábban kijelölt tárterületet kell másolnia. A paraméter megadása - és NEW LINE-nal történt lezárása - után a rendszer végrehajtja a parancsot, ezután megjeleníti a ?> prompt, jelezve, hogy képes új parancs fogadására.

START
STOP
DEST

F - Fill

A Fill parancssal a tár kijelölt területét konstans értékkel tölthetjük fel.

A parancs begépelése után - a Copy parancsnál leírt módon - adjuk meg a feltöltendő terület kezdő - és végcímét.

START
STOP
DATA

Ezek,után a DATA> üzenet jelènik meg. Most azt az értéket kell megadni, amelylyel az EDI-80 a kiválasztott területet fel fogja tölteni. A megadható érték \emptyset255. /Ha a megadott paraméter nagyobb, mint 255, a rendszer modulo 256 értékkel dolgozik./

G - Go to

A tárolóban elhelyezkedő gépi kódú program indítását végrehajtó parancs. Begépelése után az ADDR> üzenet jelenik meg, amely után az EDI-80 az indítási cím megadására várakozik. Gépeljük be a címet, majd nyomjuk meg a NEW LINE billentyűt. /Ha nem adunk meg címet - az üzenet után csak a NEW LINE billentyűt nyomjuk meg - a rendszer indítási címként az utoljára végrehajtott G parancsban megadott, ill. az RO parancssal beolvasott címtől indítja a gépi kódú program végrehajtását./ Most a BKPT> üzenet jelenik meg /BKPT = Break Point, Töréspont/. Ha megadunk töréspont címet, úgy a rendszer a kijelölt című byte tartalmát elmenti és a helyére egy RST 38H utasítást helyez, majd elindítja a program futását az ADDR> üzenet után megadott címtől. Amikor a program végrehajtása eljut ahhoz az utasításhoz, amelynek a helyére a rendszer az RST 38H utasítást /töréspont/ helyezte, a vezérlést az EDI-80 veszi vissza; elmenti a Z-80 processzor összes regiszterének pillanatnyi állapotát, visszahelyezi az RST 38H utasítás által felülírt byte-ot a tárolóba,

majd megjeleníti a ?> promptot. A törésponton való kilépés után minden MONITOR parancs használható!

/Ezzel a módszerrel - a program belövése idején - szakaszonként hajthatjuk végre a gépi kódú programot, és egy-egy rész végén ellenőrizhetjük, hogy a regiszterekben és a tárolóban a megkívánt részeredmények alakulnak-e ki.

Figyelem! Töréspont címként csak valamely utasítás első byte-jának címe adható meg, ellenkező esetben az EDI-80 nem kapja vissza a vezérlést! Ne próbáljunk töréspontot helyezni a ROM területre!/
Ha nem adunk meg töréspont címet - a BKPT> üzenet megjelenése után csak a NEW LINE gombot nyomjuk meg - az EDI-80 elindítja a program futását az ADDR> üzenet után megadott címtől.

Mn - Modify

Az M parancs a tároló tartalmának ellenőrzésére/módosítására szolgál. Gépeljük le az M betűt, majd közvetlenül ezután egy tárcímet. /Ha nem adunk meg tárcímet - az M betű után a NEW LINE billentyűt nyomva meg - az EDI-80 címként az utóljára végrehajtott M, ill. Q parancs által módosított/ kijelzett tárcímmel egygel nagyobb címet alkalmazza./ A NEW LINE billentyű lenyomása után megjelenik a megadott tárcím és az adott címen található byte - mindkét érték hexadecimálisan -, majd egy > karakter. Ha ezután a NEW LINE billentyűt nyomjuk meg,

a kijelzett byte értéke nem változik, és megjelenik a soronkövetkező byte. /Ha a soronkövetkező cím 8-cal maradék nélkül osztható, az aktuális cím és byte a következő képernyősorba kerül./

Amennyiben a NEW LINE lenyomása előtt gépelünk egy számot, úgy az EDI-80 a cím régi tartalmát a megadott új értékkel helyettesíti. A leírt módon tehát a tároló egymást követő címein elhelyezkedő értékeket tetszés szerint vizsgálhatjuk, illetve módosíthatjuk. Az M parancsból a ., majd a NEW LINE billentyű lenyomásával lépünk ki.

Qn - Query

A Q parancs segítségével a tároló megadott címtől kezdődő 64 byte-nyi területét lehet megjeleníteni hexadecimális, valamint 7 és 8 bites ASCII kód formájában. Gépeljük be a Q betűt, majd közvetlenül ezután egy tárcímet. /Ha nem adunk meg tárcímet - a Q betű után a NEW LINE billentyűt nyomjuk meg - az EDI-80 címként az utóljára végrehajtott M, illetve Q parancs által módosított, vagy kijelzett tárcímmel egygyel nagyobb címet alkalmazza./ A NEW LINE billentyű lenyomása után megjelenik a tár azon egymást követő 64 byte-nyi területe, amelynek kezdőcíme a megadott címhez legközelebb eső - annál kisebb - nyolccal maradék nélkül osztható cím./Pl, ha a megadott cím 206H volt, akkor a kijelzés 200H-től 23FH-ig terjed./

A megjelenő 8 soron belül a részletes kijelzési formátum a következő:

1. mező - hexadecimális cím
 2. mező - a tároló egymást követő 8 byte-ja hexadecimálisan
 3. mező - a 8 byte 7 bites - a byte legértékesebb bitje = \emptyset - ASCII kódja
 4. mező - a 8 byte 8 bites ASCII kódja
- A 3. és 4. oszlopban az EDI-80 a vezérlő kódok - \emptyset ...LFH - helyén . karaktereket jelenít meg.

RO - Read Object

NAME

DEST

Az RO paranccsal gépi kódú programot /object program, tárgy program/ lehet a kassetáról a tárolóba beolvasni. A parancs megadása után az EDI-80 a NAME > üzenet megjelenítését követően a betöltendő tárgyprogram nevének begépelésére várakozik. Miután pontosan leírtuk a nevet, és megnyomtuk a NEW LINE billentyűt, elindul a magnetofon és az EDI-80 az adott nevű tárgyprogram elejét keresi. Ha megtalálta, a képernyőn egy L betűt /Load/ ír ki, majd megkezdí a betöltést. /A feltöltésre kerülő tárolóterület címét, és a beolvasott program indítási címét az EDI-80 a magnószalagról olvassa le./ A beolvasás végeztével megáll a magnetofon és a képernyőn megjelenik a ?> prompt.

Amennyiben az olvasás alatt hiba történik, a képernyőn egy - vagy több - x karakter jelenik meg. Ebben az esetben az R paranccsnál ismertetett módon kíséreljük meg ismét

a betöltést. Amennyiben a NAME > üzenet után csak a NEW LINE billentyűt nyomjuk meg - nem adva meg a nevet - az EDI-80 a kazettán soronkövetkező tárgyprogramot tölti be a szövegpufferbe. Ha a megadott név nem azonos a kazettán következő tárgyprogram névvel, a rendszer átlépi ezt a programot, és a keresést tovább folytatja.

WO - Write Object

A WO parancs a tároló kiválasztott részének kazettára írását valósítja meg. A parancs megadása után a képernyőn a START > üzenet jelenik meg. Ekkor kell a kazettára kimentendő tárterület kezdő címét begépelni. A NEW LINE billentyű megnyomása után megjelenik a STOP > üzenet. Most a kiírandó terület végcímét adjuk meg, majd nyomjuk meg a NEW LINE billentyűt. /Ha a STOP > kérdésre adott paraméter - érték kisebb, mint a START - paraméter, a HUH? hibajelzés után az EDI-80 új parancsra vár./ Ezután az EDI-80 egy harmadik üzenetet - EXEC ADDR > - ír ki, amelyre válaszul a gépi program indítási címét kell megadnia. /Ha nem adjuk meg az indítási címet, a tárgyprogram kazettáról végrehajtott betöltése után - RO parancs - az indítást végző G parancsban a címet feltétlenül meg kell adni, különben annak értéke kiszámíthatatlan! / Mindezek után az utolsó kérdés - NAME > - jelenik meg. Itt adhatjuk meg a tárgyprogram nevét. A név maximum 59 karakter hosszúságú, tetszés szerinti karaktereket tartalmazó szöveg lehet. /Az RO parancs-

START

STOP

Ustelen⁶ EXEC ADDR

nem is DEPT

nem is WORKB / ADDR

NAME = (LONG DATA

ban a nevet ugyanolyan formában kell majd megadnunk, mint ahogy azt a WO parancsban begépeztük. Az EDI-80 különbözőnek tekinti a kis, illetve nagybetűket./ A név megadása után megtörténik a kazettára írás, majd ennek végeztével megjelenik a ?> prompt.

X - eXamine A parancs segítségével az utoljára végrehajtott töréspont /lásd a G parancsnál/ pillanatában a Z-80 processzor regisztereiben őrzött értékek jeleníthetők meg. A parancs megadása után a képernyőn három sor jelenik meg.

```
HL   DE   BC   AF   RI   IX   IY   SP   PC
XXXX XXXX XXXX XXXX XXXX
XXXX XXXX XXXX XXXX          XXXX XXXX XXXX XXXX
```

A legfelső sor az egyes regiszterek neveit tartalmazza, a következő sor a normál regiszter - csoport hexadecimális értékét szemlélteti, míg a harmadik sorban az alternatív regiszter - csoportban őrzött adatok, valamint az indexregiszterek, a stack pointer és a program-számláló hexadecimális értéke látható. Mivel a regiszter-értékek tárolása csak egy töréspont végrehajtásakor történik meg, ezért a megjelenített regisztertartalmakban változás csak a program továbbindítása és újabb törésponton történő kilépése után látható.

Az ASSEMBLER szintaktikája

A szövegpufferben elhelyezkedő forrásprogramoknak az alábbiakban felsorolt formai - szintaktikai - szabályoknak kell eleget tennie. Amennyiben a program megszegi a szabályokat, az ASSEMBLER hibajelzést ad, majd megszakítja a további assemblálást.

A forrásprogram sorok összessége. Az egyes sorok mindegyike maximum 56 karakter hosszú lehet és négy különféle mezőre bontható. A lehetséges mezők a következők:

- 1, címke mező
- 2, utasítás mező
- 3, operandus mező
- 4, megjegyzés /comment/ mező

Egy soron belül nem kötelező valamennyi mezőt szerepeltetni, azok bármelyike elhagyható, de ha előfordulnak, úgy az előző felsorolásbeli sorrendben kell állniuk. /Illegális azonban az olyan sor, amelyben utasítás mező nem, de operandus mező található./

Az ASSEMBLER működése során elkészített fordítási lista - függetlenül attól, hogy a forrásprogramban a listakép esztétikus kialakítása érdekében alkalmazunk-e, vagy sem betűközöket - formátumozott. Ez azt jelenti, hogy a képernyőn, illetve a sornyomtatón megjelenő listában a címke, utasítás, operandus és megjegyzés mezők rendre 8, 5, 7, illetve 25 karakter hosszúak, hacsak valamelyik mezőben nincs a felsorolt számú karakternél több, mert ebben az esetben a megfelelő mező hossza értelemszerűen a benne szereplő karakterek számával lesz egyenlő. Ha egy sor olyan hosszú, hogy

nem fér el a lista egy sorban, akkor a még hátralévő karakterek a következő sorban jelennek meg.

Címke mező

A címke mezőben az úgynevezett címkék állhatnak. A címkék a program bizonyos részeinek - tárolórekeszek, ciklusok, programszakaszok kezdetének - könnyebb azonosítását szolgálják. Az ASSEMBLER megengedi, hogy címként szimbólumokat - szimbólikus neveket - alkalmazzunk. A felhasználni kívánt szimbólumokat egyszer - de csak egyszer - szerepeltetni kell a címke mezőben. A megjelölt szimbólumhoz - amelynek utolsó karaktere után egy : karaktert kell írni - az ASSEMBLER a helyszámláló /location counter/ aktuális értékét - vagyis azt a tárcímet, amelyre a címkét követő első utasítás kerül majd - rendel. /A programban a felhasználói szimbólumokat operandusként még definiálásuk előtt felhasználhatjuk, tehát nemcsak pre-definit, hanem post-definit szimbólumokat is alkalmazhatunk.

Utasítás mező

Az utasítás mezőben a Z-80 67 szabványos utasításnéve, illetve az EDI-80 ASSEMBLER által értelmezett 7 direktíva neve állhat.

Az utasítás, illetve direktíva nevét a soronkövetkező operandustól - amennyiben az adott név után áll operandus - legalább egy betűköz karakternek kell elválasztani. Az utasítás és direktíva nevek csak nagybetűvel írhatók.

/Az EDI-80 két Z-80 utasítás megadását nem a megszokott formában várja. E két utasítás:

Szabványos írásmód	EDI-80 írásmód
RLD	RLD /HL/
RRD	RRD /HL/

Az összes többi utasítás írásmódja megegyezik a más assemblereknél szokásosan alkalmazottal./

Operandus mező:

Az operandus mezőben adhatjuk meg az adott utasítás, illetve direktiva által megkivánt operandusokat. Operandusként regiszternevek, feltételkódok, nevek, szimbólumok, konstansok, literálok, a helyszámláló aktuális értéke, illetve az utóbbi négy tagból és műveleti jelből képzett kifejezések alkalmazhatók.

Regiszternevek: a Z-80 szabványos regiszternevei - A, B, C, D, E, H, L, I, R, AF, BC, DE, HL, SP, IX, IY. Ha a regiszterneveket kerek zárójel közé tesszük, az a regiszter tartalmára utal.

Feltételkód nevek: a Z-80 szabványos feltételkód nevei - NZ, Z, NC, C, PO, PE, P, M.

Szimbólumok: Egy felhasználói szimbólum tetszés szerinti hosszúságu - de praktikusán 7 karakternél nem hosszabb -, nagybetűvel kezdődő, majd kis -, ill. nagybetűvel, számmal, vagy %, #, <, >, ?, =, ! karakterrel folytatódó karaktersorozat lehet. Utasítások, direktívák, feltételkódok és regiszterek neveit azonban felhasználói szimbólumként nem alkalmazhatja! /Ha 7 karakternél hosszabb szimbólumokat írunk, azok már nem férnek el a címke mezőben, ami felborítja a szabályos listaformátumozást/.

Csal hexa vezető Ø vel

Konstansok: Decimális, hexadecimális /a szám végén H betű/, vagy oktális /a szám végén O betű/ számokat alkalmazhatunk. Az ASSEMBLER, ha a leírt szám decimális megfelelője nagyobb, mint 65535, a megadott érték modulo 65536 eredményét használja. /Ha egy hexadecimális konstans nem számjeggyel kezdődik, úgy azt egy vezető Ø-val kell kiegészítenie. Pl. ØDE/

Literálok: /szövegkonstansok/: idézőjelek /"/, vagy aposztrófok /'/ közé zárt karaktersorozatokat alkalmazhat. A határoló jelek között valamennyi - a számítógépben alkalmazható - karakter előfordulhat, kivéve magát a határoló jelet. A határoló jelek között legalább egy karakternek állnia kell, ami betűköt is lehet. Ha a literál elején határoló jelként az idézőjelet /"/ alkalmaztuk, úgy a szövegkonstans végén is azt kell szerepeltetni. Ez értelemszerűen igaz az aposztrófra /'/ is. Ha a szövegkonstansban idézőjelet is fel kívánunk használni, akkor határoló jelként az aposztróft alkalmazzuk, és fordítva, aposztróf alkalmazása esetén a határoló jel az idézőjel legyen.

Helyszámláló aktuális értéke: a # minden pillanatban a helyszámláló aktuális értékét képviseli. Ez az érték egyenlő azon utasítás első byte-jének tárcímével, amely utasítás operandus mezejében a \$ szimbólumot felhasználja. Pl. a JP \$ utasítás önmagára ugrást eredményez.

Kifejezősek: felhasználói szimbólumokból, konstansokból, literálokból és a \$ szimbólumból műveleti jelek felhasználásával kifejezések képezhetők. Egy műveleti jel előtt és mögött is operandusnak kell állnia, az unáris +, illetve - művelet nem megengedett. A műveletek precedenciája azonos, a precedenciát zárójelzéssel nem befolyásolhatjuk /zárójelek a kifejezés-

ben nem alkalmazhatók./ Egy kifejezésben tetszőlegesen sok tag állhat, a kiértékelést az ASSEMBLER balról jobbra végzi. Ha a kifejezés értéke meghaladja a decimális 65535-öt, az eredmény a modulo 65536 lesz. A felhasználható műveleti jelek és értelmezésük a következő:

+	összeadás
-	kivonás
*	szorzás
/	egész osztás
&	logikai ÉS
.	logikai VAGY

Megjegyzés mező:

A forrásprogramban a jobb megértés kedvéért megjegyzéseket /comment/ alkalmazhatunk. A megjegyzés kezdetét a ; karakter jelöli, a megjegyzés vége a sor vége.

Az EDI-80 ASSEMBLER direktívái

ORG - Origin

A direktíva segítségével az ASSEMBLER helyszámlálójának adhatunk értéket. Az operandus mezőben álló kifejezés értéké lesz a helyszámláló új értéke, tehát a program soronkövetkező utasítását az ASSEMBLER erre a címre fogja assemblálni. Az ORG direktívát egy programon belül többször is alkalmazhatjuk, de a program első - nem megjegyzés - sorában kötelezően ennek a direktívának kell állnia. Ha a direktíva operandus mezejében álló kifejezés értéke nagyobb, mint 65535, a rendszer a modulo 65536 értékkel számol.

DS - Define Storage

A DS direktívával a helyszámláló aktuális értékét az operandus mezőben álló kifejezés értékével megnövelhetjük, vagyis adott darabszámú byte számára helyet foglalhatunk a tárban. Ha a helyszámláló megnövekedett értéke nagyobb lenne mint 65535, akkor az ASSEMBLER a modulo 65536 értékkel számol.

EQU - Equate

Egy felhasználói szimbólumhoz az EQU direktíva segítségével rendelhetünk értéket. A címke mezőben álló - kettősponttal lezárt - szimbólumhoz az operandus mezőben álló kifejezés értékét rendeli az ASSEMBLER. Ha az operandusként megadott kifejezés értéke nagyobb, mint 65535, a rendszer a modulo 65536 értékkel számol. Amennyiben a címke mezőben olyan felhasználói szimbólum szerepel, amely korábban már kapott értéket, az ASSEMBLER

a DOUBLE SYMBOL jelzéssel figyelmeztet a hibára./Szimbólum újradefiniálása ugyanis nincs megengedve./

DB - Define Byte

A DB direktívával 8 bites konstansokat, illetve egy, vagy több karakterből álló szövegkonstansokat helyezhetünk el a gépi kódú programban. A direktíva operandusa a helyszámláló aktuális értéke által kijelölt című byte-ba kerül. Ha több - egymástól vesszővel elválasztott - operandust szerepeltetünk, azok rendre a tároló egymást követő címéire kerülnek. Pl. a

DB 32,7FH,"Alfa"

forrásprogram sor hatására az ASSEMBLER a gépi kódú programba a helyszámláló által kijelölt címtől kezdődően a következő hexadecimális értékű byte-ókat helyezi:

20,7F,41,6C,66,61

Amennyiben a direktíva valamelyik operandusa meghaladja a 255-ös értéket, az ASSEMBLER OPND hibajelzést generál.

DW - Define Word

A DW direktívával 16 bites konstanst /pl. címkonstanst/ helyezhetünk el a gépi kódú programban. A direktíva operandusának kevésbé értékes 8 bitje /LSB/ a helyszámláló aktuális értéke által kijelölt címre, míg az értékesebb 8 bit /MSB/ az eggyel nagyobb tárcímre kerül. Ha a direktíva operandusként megadott kifejezés értéke nagyobb, mint 65535, a rendszer a modulo 65536 értékkel számol.

LOAD

A LOAD direktíva segítségével az assemblálás során keletkező gépi kódot közvetlenül betölthetjük a tárolóba. Nem feltétlenül szükséges tehát a tárgyprogramot /gépi kódot/ kazettára kimenteni, majd a Monitor RO parancsával a tárba tölteni.

A LOAD direktívát - a következő ORG direktíváig - követő gépi kódokat az ASSEMBLER a LOAD operandusaként megadott címtől kezdődően az Assemblálás ideje alatt a tárolóba írja.

Ha a direktíva operandusaként megadott kifejezés értéke nagyobb, mint 65535, a rendszer a modulo 65536 értékkel számol.

A direktíva működésének jobb megértése érdekében nézzünk meg egy példát!

```
START:      ORG 7000H           ; Kezdőcím
            LOAD 7000H         ; Betöltés címe
            LD A,B
            NOP
            JP START
CIMKE:      ORG 7005H           ; Betöltés megszakad
            JP CIMKE
            END                 ; Program vége
```

A példában látható forrásprogram assemblálása alatt az ASSEMBLER a LD A,B : NOP : JP START utasítások kódját - a LOAD hatására - közvetlenül betölti a tároló 7000H címétől kezdődően, míg ha egyidőben tárgyprogram szalag generálását is kértük az ASSEMBLER-től, úgy az a

```
LD A,B : NOP : JP START : JP CIMKE
```

utasítások gépi kódjait fogja tartalmazni. Mint látható a LOAD direktíva hatása a következő ORG-ig terjed.

E direktíva segítségével a gépi programok kifejlesztését jelentősen meggyorsíthatjuk, hisz nem kell a hosszadalmas kazettára mentéssel lassítani munkánkat. Elegendő a tárgyprogramot akkor kazettára írni, amikor már az egymást követő futtatások és javítások után megbizonyosodtunk arról, hogy programunk hibátlan. Annak érdekében azonban, hogy a gépi kód tárba töltésével az EDITOR szövegpufferében őrzött forrásprogramot, illetve magát az EDI-80-at tönkre ne tegyük, ügyeljünk a LOAD operandusaként megadott betöltési cím helyes megválasztására. Ezért betöltési címként ne adjunk meg olyan értéket, amely 17152 és a szövegpuffer aktuális végcíme közé esik. Ez utóbbi értéket a Monitor H parancsának segítségével határozhatjuk meg./ Vegyük figyelembe azt is, hogy a 16 Kbyte RAM tárolóval felszerelt HT-1080 Z számítógépben elérhető legmagasabb tárcím értéke 32767 /7FFFH/. Amennyiben a HT-1080 Z kezelését megkönnyítő NEW KEYBOARD ROUTINE-t használjuk, az elérhető max. tárcím 32671-re csökken!

END

A forrásprogram utolsó sorában kötelezően az END direktívának kell állnia. Ez a direktíva jelzi az ASSEMBLER számára, hogy elért a forrásprogram végére. Ha a forrásprogram END direktíva nélkül fejeződik be, az ASSEMBLER EOF hibajelzéssel figyelmeztet a hibára.

Az ASSEMBLER hibajelzései

Az ASSEMBLER működése során a felismert hibákra különféle figyelmeztető jelzésekkel reagál. Egy hiba észlelése után az assemblálás megszakad, és az EDITOR pointere a hibajelzést okozó forrásprogram sor elejére mutat.

Az ASSEMBLER hibajelzései, és azok jelentése a következő:

DOUBLE SYMBOL	a programban két, azonos nevű szimbólum található /illegális szimbólum újradefiniáció/.
EOF	üres a szövegpuffer, vagy nincs a forrásprogram végén END direktíva.
FULL	betelt a szimbólumtábla; a programban túl sok szimbólumot alkalmaztunk. /A szimbólumtábla maximális mérete 605 byte. Ezen a területen pl. 67 db, egyenként 7 karakter hosszú szimbólum tároható./
HUH?	szintaktikai hiba
ORG	a forrásprogram első - nem megjegyzés - sorában nem ORG direktíva áll.
OPND	az utasítás, illetve direktíva operandusa hiányzik, vagy a megadott operandus hibás.
UNDEF	a forrásprogramban definiálatlan szimbólum található.

Az EDI-80 parancsok összefoglalása

Parancsok	oldalszám
A - Assemble	11
B - Bottom	4
C - Copy	14
D - Down	4
E - Enter	5
F - Fill	14
G - Goto	15
H - How	10
K - Kill	7
L - Locate	4
M - Modify	16
N - New	6
P - Print	5
Q - Query	17
R - Read	7
Ro - Read Object	18
S - Sort	13
T - Top	5
U - Up	5
V - Verify	9
W - Write	8
WO - Write Object	19
X - eXamine	20
Z - Zap	7

A HT 1080 Z klaviatúra- és display rutinjai

A következőkben megadjuk a klaviatúra és display kezeléséhez szükséges szubrutinok belépési pontjait és hívási feltételeit.

Klaviatúra:

Hívási cím: 002BH

Funkció: INKEY rutin

Működése: Lekérdezi a klaviatúrát, amennyiben nincs gomb lenyomva, akkor az A-regiszter tartalma zéró, ha van megnyomott gomb, akkor az A-regiszter a gomb kódját tartalmazza.

Hatása: A = gomb ASCII kódja
Flagek, és a D, E regiszterpár tartalmát rontja

Hívási cím: 0049H

Funkció: INPUT rutin

Működése: Lekérdezi a klaviatúrát és addig vár, amíg egy gombot meg nem nyomunk. Visszatéréskor az A-regiszter tartalmazza a megnyomott gomb kódját.

Hatása: A = gomb ASCII kódja
Flagek, és a D, E regiszterpár tartalmát rontja.

Display:

Hívási cím: 0033H

Funkció: DISPLAY rutin

Működése: Az A-regiszterben átadott ASCII kódnak megfelelő karaktert megjeleníti a képernyőn az aktuális kurzor pozíción.

Hatása: Flagek, D, E, IY-regiszterek tartalmát rontja.

Hívási cím: 01C9H 457
Funkció: CLS rutin
Működése: Törli a képernyőt, a kurzort a bal felső sarokba viszi.
Hatása: Flagek, és az A-regiszter tartalmát rontja

Hívási cím: 28A7H
Funkció: PRINT rutin
Működése: A Display aktuális kurzor pozíciójától kezdődően megjeleníti azt a tárban lévő szöveget, melynek kezdőcímét a HL-regiszterpárban kell átadni. A szöveget egy 00H, vagy 0DH értékű byte-tal kell zárni.
Hatása: Flagek, és a A, H, L-regiszterek tartalmát rontja.

Gépi kódu rutinok hívása BASIC-ből

A gépi kódu rutinok BASIC programba illesztéséhez, illetve meghívásához a következő utasítások és függvények adnak segítséget.

POKE tárcim,adat

Az utasítás a megadott tárcimre tárolja az utasításban megadott adatot. Az adat értéke 0-255 értéket vehet fel.

változó = PEEK /tárcim/

A függvény az adott tárcim található byte értékének decimális megfelelőjét adja át a változónak.

változó = USR /0/

Az utasítás elindítja azt a gépi szubrutint, melynek kezdőcímét előzőleg letároltuk.

A kezdőcímet két POKE utasítással a 16526 és 16527-es címekre kell elhelyezni, méghozzá úgy, hogy először a cím alacsonyabb, majd magasabb helyiértékű byte-ját.

példa:

A 7000H - 7000H tárterületre nyugodtan írhatunk gépi kódu rutinokat, feltéve, ha a BASIC programunk nem hosszabb 12 Kbyte-nál és a CLEAR parancs operandusában nem adtunk meg 2000-nél nagyobb számot.

Tegyük fel tehát, hogy a rutin kezdő címe 7015H = = 28693. akkor a kezdőcím elhelyezése a következőképpen néz ki.

POKE 16527, INT /28693/256/: POKE 16526,

28693 - 256 x PEEK /16527/

A 7015H átkonvertálását decimális számmá most csak azért végeztük el, hogy decimális érték esetére is példát mutassunk.

Ha a kezdőcím értéke hexadecimálisan áll rendelkezésünkre, /vagy átkonvertáltuk/ akkor természetesen a kezdőcím letárolása lényegesen egyszerűbb.

Vegyük az előző 7015 hexadecimális értéket. A cím

alsó byte-ja: $15H = 1 \cdot 16 + 5 = 21$

felső byte-ja: $70H = 7 \cdot 16 + 0 = 112$

tehát a cím letárolása:

POKE 16526, 21: POKE 16527, 112

Az USR függvény a vezérlést tehát arra a címre adja, amit előzőleg a 16526 és 16527-es byte-okba letároltunk. A gépi szubrutinból a vezérlést a BASIC-be egy RET utasítással /C9H kód/ adhatjuk vissza.