

EDICE ELEKTRONIKY SVAZARMU



ASSEMBLER-MONITOR PRO  
PMD-85  
**DAM-0000**

Karel Šuhajda

ŘADA 5



výpočetní  
technika

EDICE  
ELEKTRONIKY  
SVAZARMU

---

Karel Šuhajda

---

ASSEMBLER-MONITOR PRO  
**DAM-1**  
PMD-85



ÚSTŘEDNÍ VÝBOR  
SVAZARMU

---

PRAHA 1987



4004/482 ZO Svazarmu 1986

DAM je program určený k práci ve strojovém kódu na počítači PMD-85 1. Tvoří kompletní systém umožňující tvorbu a ladění programů vlastních i zpětný překlad programů již hotových. Takto jsou také rozděleny jeho jednotlivé části:

DEBUGGER - umožňuje krokování programů mnoha různými způsoby (po jedné instrukci, podprogramy v reálném čase, rychlá či pomalá simulace ...).

ASSEMBLER - umožňuje tvorbu, záznam a překlad zdrojových textů v JSA MHB 8080.

MONITOR - částečně nahrazuje služby OS PMD, navíc obsahuje rutinu pro vyhledání zadané posloupnosti byte v paměti a zpětný překlad obsahu paměti.

Prvním krokem při použití DAMu je jeho nahrání do počítače. Na rozdíl od obvyklého tvaru příkazu MGLD se za číslo bloku zadává ještě horní polovina nahrávací (a spouštěcí) adresy. Např. "MGLD 00 24" nahraje DAM od adresy 2400H. Napišete-li "MGLD 00", nahraje se DAM od adresy 0000H. Pokud zadáte adresu větší než 58, ohlásí zavaděč chybu "\*\*\* FILE ERROR \*\*\*" a program se nenahráje. Stejné hlášení se objeví v případě, že se program nahraje s chybou. Pokud se DAM nahraje správně, sám se také spustí. Znovu jej lze spustit od nahrávací adresy. Při novém spuštění se provádí inicializace systémových proměnných DAMu, ale zdrojový text zůstává v paměti zachován a lze s ním dále pracovat.

S možností nahrání DAMu od libovolné adresy souvisí také způsob, jakým si DAM rozvrhuje paměť. Tato je rozdělena do částí podle následující tabulky:

Počáteční adresa	obsah	poznámka
0000H	uživ. paměť	volné
XX00H	DAM	chráněno
XX00H+1800H	print-buffer	chráněno
XX00H+1900H	uživ. paměť	volné
7600H	zásobník DAMu	chráněno
7700H	lokace OS klíčů	chráněno
7F00H	prac. oblast OS	chráněno
8000H	ROM	chráněno
C000H	syst. prom. OS	chráněno
C230H	syst. prom. DAM	chráněno
C340H	zbytek obrazovky	volné
FFFFH	KONEC PAMĚTI	

XX00 zde značí adresu, od které jste si DAM nahráli.

Pokud jste si DAM nahráli od adresy 0000H, zaniká první volná oblast paměti. Mezi chráněné oblasti přísluší také oblast zdrojového textu a oblast tabulky návěští. Tyto si můžete umístit libovolně ve volných oblastech, pouze zdrojový text nelze ukládat do obrazovky. Standardně je zdrojový text ukládán těsně za print-buffer, tabulka návěští těsně za zdrojový text.

Po nahrání vstupuje DAM do volící smyčky, ze které lze vstupovat do jeho jednotlivých částí. Stiskem libovolné klávesy kromě 'Q', 'EOL' a 'C-D' se přepínají jednotlivé varianty. Stiskem 'EOL' provedete skok do zvolené části, stiskem 'Q' se smyčka opouští (v základní smyčce se provede skok do OS PMD, ze smyček vyšších úrovní se vyskakuje do základní smyčky). Klávesa 'C-D' se používá k nastavení speciálních funkcí - viz dále.

Základní smyčka obsahuje tyto varianty:  
ASSEMBLER - MONITOR - DEBUGGER

## ASSEMBLER

Je jediná část programu, která rozlišuje hexadecimální, dekadická a binární čísla. Všechna čísla musí začínat číslicí '0' - '9', hexadecimální čísla končí znakem 'H', binární 'B'.

Obsahuje tuto smyčku:

EDIT - TRANSLATE - SAVE - VERIFY - LOAD - MERGE

## EDIT

Slouží k vytváření a úpravám zdrojových textů. Je řádkově orientován. POZOR! Čísla řádků se neuchovávají. Po jakékoli změně se provede přečíslování všech řádků ve zdrojovém textu se zadaným krokem (například pokud vložíte mezi řádky s čísly 20 a 30 řádek s číslem 25, má po přečíslování začleněný řádek číslo 30 a bývalý řádek 30 má číslo 40). Po spuštění DAMu je hodnota kroku nastavena na deset.

Po spuštění editoru se do dialogového řádku vypíše text "ENTER LINE OR 'Q' TO QUIT". Stiskem 'Q' se vyskakuje do smyčky ASSEMBLERu. Pokud stisknete cokoli jiného (opět s výjimkou 'C-D'), vstoupíte do režimu vstupu řádku, který se opouští stiskem 'EOL'. Nyní je možné zadat řádek zdrojového textu nebo příkaz editoru. Při vstupu řádku jsou všechny počáteční mezery ignorovány.

Řádek zdrojového textu má mít tento tvar:

číslo\_řádku   pole\_návěští   pole\_instrukce   pole\_argumentu

Pokud se číslo řádku shoduje s číslem některého z řádků ve zdrojovém textu, dojde k přepsání tohoto řádku řádkem zadaným, jinak dojde pouze k vložení tohoto řádku.

Mezi číslem řádku a návěštím se musí vyskytovat právě jedna mezera. Pokud za číslem řádku následují dvě nebo více mezer, je tento řádek považován za řádek bez návěští. Návěští nemá být ukončeno dvojtečkou, tato by se vyhodnotila jako platný znak návěští. Návěští musí začínat písmenem, uvnitř návěští lze použít libovolné znaky. Nedoporučují se znaky '+', '-' a '\'. Jsou povolena návěští delší než šest znaků, ale při výpisu a opravách jsou přepisována zbytkem řádku.

V přijatém řádku se provádí částečná syntaktická kontrola, t.j. pokud se v poli instrukce najde instrukce assembleru, kontroluje se dále, zda této instrukci odpovídají i parametry v poli argumentu. Pokud ne, řádek není přijat a ohlásí se "\*\*\* LINE INPUT ERROR \*\*\*". Lze jej opravit po stisku klávesy 'RCL' nebo lze stiskem 'Q' opustit editor. Pokud není v poli instrukce nalezena instrukce, je pole instrukce považováno za prázdné a řádek chápán jako datový. Řádek nemusí být přijat také v případě, že by zdrojový text přetekl mimo oblast vyhrazenou uživateli - v takovém případě se vypíše hlášení "\*\*\* MEMORY OVERFLOW \*\*\*".

Příkazy editoru se vyznačují tím, že se nemezerový znak vyskytuje těsně za číslem řádku, nebo není číslo řádku uvedeno vůbec (může být nulové). Pokud zadané číslo řádku neukazuje na žádný řádek v textu, pak se bere řádek s nejbližším vyšším číslem.

1/ Jednoduché příkazy - znak následuje bezprostředně za číslem řádku

- L - výpis textu od zadaného čísla řádku. Ovládání výpisu bude popsáno později.
- E - žádost o opravu řádku daného čísla - řádek se vypíše do editační oblasti.
- S - žádost o změnu intervalu číslování řádků. Např. při použití "5S" budou řádky číslovány 5, 10, 15, 20 ... Pokud je zadaná hodnota 0 nebo není uveden žádný parametr, objeví se hlášení "\*\*\* LINE INPUT ERROR \*\*\*".



D - maže řádek daného čísla.

N - ruší všechny řádky od zadaného až do konce textu.

Příklad použití:

1250L vypisuje program od řádku 1250

5S mění krok číslování řádků na pět

2/ Blokové příkazy - za číslem řádku následuje pomlčka, pak další číslo řádku a pak znak příkazu, za ním popřípadě ještě číslo řádku.

110-230M40 zkopíruje řádky 110 až 230 před stávající řádek číslo 40. Při kopírování větších bloků textu je tato funkce značně časově náročná. Pokud by konec zdrojového textu po přenosu zasáhl do chráněných oblastí, přenos se neprovede a objeví se hlášení "\*\*\* MEMORY OVERFLOW \*\*\*".

120-250D maže řádky s čísly 120 až 250.

3/ Zvláštní příkazy - bez čísla řádku, popřípadě může předcházet libovolný počet nul. Těsně za znakem může následovat parametr.

N - ruší celý zdrojový text.

T - přenáší začátek zdrojového textu na uvedenou adresu (například "T5000H" na adresu 5000H). Pak se provede kontrola, jestli se na této adrese nenachází jiný zdrojový text. Pokud ano, je přiřazen (stejná kontrola se provádí po nahrání DAMu a při každém jeho dalším spuštění).

L - vypisuje text od řádku označeného zadaným návěštím (například "LSTART" vypisuje od řádky označené návěštím "START").

Překladač nezná obvyklé pseudoinstrukce, ty jsou však nahrazeny speciálními symboly, nebo jinou syntaxí zápisu zdrojového textu:

ORG: adresu je nutno zapsat tak, jako by se jednalo o návěští, například:

```
10 6000H ;  
10 4000H PUSH H
```

EQU: do pole instrukce za návěštěm se napíše znak '=', za ním následuje výraz vyčíslitelný během prvního průchodu překladu (všechna v něm uvedená návěští mu musí předcházet). Např.:

```
20 CONST =12+CONST2\ -1
```

DW: nahrazeno znakem '\*', následuje výraz.

```
30 WORD *0C32H-'AB'
```

DS: nahrazeno znakem '#', následuje číselná konstanta.

```
40 POLE #400
```

DB: nahrazeno dvěma různými způsoby:

1/ přímé zadání hexadecimálních dvojčíslí a znakových řetězců ukončené středníkem.

```
50 'WRITE 'AHOJ''' 0D 01 0A4050;
```

2/ číselné zadání - údaje začínají a jsou navzájem odděleny čárkami, ukončeny jsou opět znakem ';'.  
;

```
60 ,1,2,3,4,20H,01110001B;
```

Za středníkem ukončujícím data může následovat libovolný komentář. Pokud požadujete pouze komentář, lze napsat prázdný soubor dat:

```
70 ; KOMENTAR
```

Ostatní pseudoinstrukce nebyly implementovány, místo nich byly zavedeny speciální znaky pro další funkce:

[ ... ]

Každá ze závorek musí být uvedena jako instrukce na zvláštním řádku a může být následována komentářem. Závorky mezi sebou uzavírají část textu, jež se nepřekládá. Levá závorka může být označena návěštím, jež se ještě překládá. Případné návěští u pravé závorky je ignorováno. Další levé závorky uvnitř označeného bloku jsou ignorovány také. Výskyt samostatné pravé závorky je při překladu indikován jako chyba.

```
80 CONST  =0F6H
90        [  VERZE PRO PP
100 CONST  =0C9H
110        ]
```

@ ... @

Specifikace souboru pro překlad na MG pásku. První (a každý lichý) znak '@' soubor otevírá. Může za ním následovat mezera, 'H' nebo 'N'. Mezeru následuje číslo bloku (0-63) a jeho název. Při druhém průběhu překladu je vypsán požadavek "START TAPE, THEN PRESS ANY KEY". Pak je generována hlavička bloku.

Uvedení znaku 'H' odstraňuje generování hlavičky (vytváří se bezhlavičkový blok), 'N' odstraňuje i požadavek "START TAPE ... ".

Sudý znak '@' uzavírá soubor, může za ním následovat znak '-', který odstraní záznam kontrolního součtu bloku.

Pokud není prováděn překlad na MG pásku, jsou řádky označené znaky '@' ignorovány. Znaků '@' musí být v textu vždy sudý počet, lichý počet vede při překladu k chybě "\*\*\* FILE NOT CLOSED \*\*\*". Pokud mezi sebou znaky '@' uzavírají prázdný soubor, dojde k chybě "\*\*\* NO FILE \*\*\*".

```
120      @ 03 SOUBOR
130      LXI H,0
140      @
150      @N
160      MOV A,L
170      @-
```

W

Slouží k umělému vytváření časových mezer při překladu na MG pásku. Při použití bez parametru nebo s parametrem menším než 256 je generována časová prodleva asi 0,5 s. Jiné parametry umožňují jemnější dělení tohoto časového intervalu (256 je nejkratší).

```
180      @H
190      FFFFFFFFFFFFFFFF00; HLAVICKA
200      @
210      W
220      @N
      ...
900      @
```

V aritmetických výrazech ( v instrukcích i pseudoinstrukcích ) se počítá se šestnáctibitovými hodnotami. Jako operandy jsou povolena návěští, číselné konstanty a znakové konstanty (v apostrofech, vyhodnoceny jsou vždy jen dva poslední znaky). Operátory existují binární - '+' a '-', a unární '\'. Tento operátor způsobí prohození vyššího a nižšího byte zatím dosaženého výsledku. Vyhodnocování výrazu probíhá vždy zleva doprava, všechny operace mají stejnou prioritu.

Příklad práce s editorem - zápis programu a opuštění editoru. Znaky uzavřené mezi "<" a ">" znamenají bezprostřední stisk daného tlačítka.

zapsáno	odpověď
.....	ASSEMBLER
<EOL>	EDIT
<EOL>	ENTER LINE OR 'Q' TO QUIT
<C-D> <K0>	ENTER LINE OR 'Q' TO QUIT
10 6000H ;	0020
0020 MVI A, 'A' <EOL>	0030
0030 JMP 8500H <EOL>	0040

zapsáno	odpověď
<CLR> 1L	0010 6000H ;
	0020 MVI A, 'A'
	0030 JMP 8500H
	ENTER LINE OR 'Q' TO QUIT
20E	0020 MVI A, 'A'
0020 MVI A, 'B'	0030
<CLR> <EOL>	** LINE INPUT ERROR **
<Q>	EDIT

## TRANSLATE

Umožňuje překlad zdrojového textu v několika režimech. Po spuštění se táže "OPTIONS ?" a čeká na stisk hexadecimální klávesy ( '0' - '9' nebo 'A' - 'F' ). Při stisku 'Q' vyskakuje do smyčky ASSEMBLERu, při stisku jiných tlačítek (s výjimkou 'C-D') neprovádí nic. Zadaná hexadecimální číslce nastavuje čtyři podmínkové bity překladu a spouští překlad.

Při stisku '0' dochází k normálnímu překladu všech částí programu za současného výpisu návěstí a jejich hodnot.

Při použití '1' si můžete zadat polohu tabulky návěstí v paměti - vypíše se otázka "TABLE ADDRESS ?". Odpověď lze stiskem 'Q' (opuštění překladu) nebo napsáním adresy tabulky. Pokud nenapíšete žádnou adresu, vyhodnotí se jako nula. Pokud zasahuje vámi zadaná adresa do chráněných oblastí, program se táže znovu.

Použitím '2' lze provádět překlad na jinou lokaci, než je zadáno ve zdrojovém textu. Po proběhnutí první části překladu (vytvoření tabulky) se překladač táže "START ADDRESS ?" a čeká na zadání adresy, od které bude ukládán generovaný kód (bez ohledu na ORG). Během překladu

se vypisují zadávané hodnoty ORG a vedle nich adresy, od kterých jsou dané bloky skutečně umístěny. Po skončení překladu se vypíše otázka "TRANSFER ?" a čeká se na stisk 'Y' nebo 'N'. Při stisku čehokoli jiného (včetně 'Q', výjimku tvoří 'C-D' ) otázku opakuje. Při stisku 'N' ukončuje překlad, při stisku 'Y' vypíše otázku "RUN LABEL ?" a čeká na zadání návěští, od něhož má program odstartovat po provedení přenosu. Lze použít také pseudonávěští "DAM" nebo "OS". Pokud se ve vašem textu vyskytují takováto návěští, mají pseudonávěští přednost. "DAM" umožňuje návrat do DAMu, "OS" se vrací do OS PMD. Pokud se vámi uvedené návěští nevyskytuje ve zdrojovém textu, překladač opakuje otázku. Z tohoto režimu lze vyskočit stiskem 'Q'. Po přijetí návěští je jeho hodnota podle tabulky zaznamenána coby návratová adresa a celý blok vygenerovaného kódu je přenesen na adresu podle prvního ORG v textu, a to směrem od nižších adres k vyšším.

Stisk '4' ruší výpis návěští během překladu.

Stisk '8' provádí překlad na MG pásku. Ruší podmínku 2 (t.j. pokud jste stiskli 'A', je činnost stejná, jako při stisku '8'). Je nutná spolupráce s pseudoinstrukcemi ve zdrojovém textu (všechny instrukce mimo specifikované bloky jsou sice přeloženy, ale nedojde k jejich záznamu).

Režimy při překladu odpovídající kombinacím uvedených možností získáte prostým součtem kódů požadovaných dílčích činností.

Pokud by měly být při překladu přepisovány chráněné oblasti paměti, vypíše se "\*\*\* MEMORY OVERFLOW \*\*\*" (první průchod překladače, tabulka se nevejde na danou lokaci), nebo "\*\*\* BAD ORG \*\*\*" (špatně směřovaný zdrojový text). Pokud je v textu nalezen řádek neobsahující instrukci ani data, vypíše se "\*\*\* ERROR IN DATA \*\*\*". Při ostatních chybách (chyba v aritmetickém výrazu, nevyskytující se návěští, dvakrát deklarované návěští atd.) se nevypíše nic.

Při libovolné chybě se překlad ukončuje a do dialogového řádku se vypíše řádek, na němž k chybě došlo (toto se neprovede pouze při chybě "\*\*\* FILE NOT CLOSED \*\*").

## SAVE

Ušchovává zdrojový text na magnetofonovou pásku. Po spuštění se vypíše "ENTER NAME" a čeká se na zadání názvu. Je zpracováváno pouze osm prvních znaků, pokud zadáte kratší název, je doplněn do délky mezerami. Pak se vypíše "ENTER SEGMENT NUMBER OR 'Q' TO QUIT" a čeká se na zadání čísla bloku. Je-li zadané číslo větší než 63, vypíše se "\*\*\* IT'S TOO HIGH \*\*\*" a program čeká na nové zadání čísla bloku. Funkci lze opustit stiskem 'Q'. Po zadání čísla se vypíše "START TAPE, THEN PRESS ANY KEY" a po stisku libovolného tlačítka (s výjimkou 'Q' a 'C-D') začne nahrávat. Současně se do dialogového řádku vypíše obsah hlavičky. Po skončení nahrávání vypíše "MG STOP !" a po stisku libovolného tlačítka se vstupuje do smyčky ASSEMBLERu.

## VERIFY

Funkce shodná s MGEND v operačním systému PMD. Žádá zadání čísla bloku stejně, jako SAVE. Potom kontroluje blok na MG pásce a v případě chyby vypíše "\*\*\* FILE ERROR \*\*\*". Pokud nebyla dosud nalezena zadaná hlavička, lze funkci opustit stiskem "STOP".

## LOAD

Nahrává nový zdrojový text. Obsluha je stejná, jako u VERIFY. Pokud by se nahrávaný zdrojový text nevešel do vyhrazené oblasti paměti, nenahrává se a vypíše se "MEMORY OVERFLOW \*\*\*". Pokud při nahrávání došlo k chybě, vypíše se "\*\*\* FILE ERROR \*\*\*" a zdrojový text je z paměti "vymazán" (první znak textu se přepíše koncovým znakem 0FDH).

## MERGE

Přihrává zdrojový text z magnetofonu za konec současného zdrojového textu. Obsluha je stejná, jako u LOAD. Při chybě na pásku "smaže" nahranou část, starý zdrojový text ponechá na místě.

POZOR ! - pokud během nahrávání stisknete RESET a znovu spustíte DAM, může se stát, že zdrojový text zmizí.

## MONITOR

Slouží k zobrazení a úpravě obsahu paměti.  
Obsahuje smyčku: EDIT - FIND - LIST - DISASSEMBLER

## EDIT

Spojuje vlastnosti příkazů MEM a SUB v operačním systému PMD. Po spuštění se vypíše "ENTER ADDRESS OR 'Q' TO QUIT" a program očekává zadání adresy, od níž si přejete měnit obsah paměti. Pokud stisknete 'Q', dostanete se zpět do smyčky monitoru. Zadáte-li čtyři hexadecimální číslice a stisknete 'EOL', zobrazí se do dialogového řádku znovu zadaná adresa a za ní obsah šestnácti odpovídajících paměťových míst. Pokud za adresou napíšete data v hexadecimální nebo znakové formě (znakové řetězce je nutno uzavřít mezi apostrofy, případný apostrof lze zapsat pomocí dvou apostrofů za sebou, data je možno ukončit středníkem), tato se zapíše do paměti a do dialogového řádku se vypíše následující adresa. Pokud zadáte data chybně, objeví se nápis "\*\*\* ERROR IN DATA \*\*\*". Řádek lze opravit po stisku 'RCL' nebo lze pomocí 'Q' editor opustit. Stejným způsobem lze editor opustit, pokud byla chybně zadána adresa a svítí nápis "\*\*\* ERROR IN ADDRESS \*\*\*".



## FIND

Vyhledává v paměti zadanou posloupnost byte. Vždy ji nalezne nejméně jednou (nejspíše na adrese 7F82H, kde je obvyklá adresa textového bufferu). Prohledávání pokračuje stále dokola, je třeba jej zastavit ručně - klávesou 'STOP'. Zadání se provádí stejně, jako u EDITu, adresa označuje místo, od něhož se začne vyhledávat. Po zastavení prohledávání lze pokračovat stiskem 'EOL', opustit funkci stiskem 'Q', nebo zadat novou adresu a data. Funkci lze výhodně použít k nalezení konce zdrojového textu - prohledávání začíná na počáteční adrese textu, vyhledávaným byte je FD.

## LIST

Vypisuje obsah paměti v řádcích po osmi byte stejným způsobem, jako DUMP u OS PMD, to jest na začátku řádku se vypíše adresa, za ní následuje osm hexadecimálních kódů a pak tytéž kódy ve znakové reprezentaci uzavřené mezi znaky "!", netisknutelné znaky jsou nahrazeny tečkami. Výpis lze řídit stejně, jako u FIND - po zastavení lze ve výpisu pokračovat, opustit režim nebo zadat novou adresu.

## DISASSEMBLER

Zobrazuje obsah paměti ve formě zpětného překladu. Zadání adresy i řízení výpisu se provádí stejně, jako v předchozích případech. Na začátku řádku se vypíše adresa instrukce, za ní její hexadecimální reprezentace v paměti a dále následuje vlastní instrukce v mnemonice JSA 8080, přičemž kódy, jež nejsou instrukčními kódy, se zde ještě jednou zopakují.

## DEBUGGER

Po spuštění této části programu se smaže obrazovka a zaplní se následujícím způsobem:

V levé horní části obrazovky se vypisují obsahy registrů ve tvaru RR XXXX BBBB BBBB ZZ, kde RR je název registru (AF pro PSW), XXXX hexadecimální, BBBB... binární a ZZ znaková reprezentace jeho obsahu. Nad indikátorovou částí registru AF se píší názvy indikátorů (M..MSB, Z..ZERO, H..HALF CARRY, P..PARITY, C..CARRY). Na pravé straně obrazovky se vypíše nejbližších šestnáct položek na zásobníku. Pod obsahy registrů se zobrazí i hexadecimální a znaková reprezentace obsahu paměti adresované registry BC, DE a HL. O něco níže se vypíše mnemonika instrukce, na níž ukazuje PC a jež se má právě simulovat. Pod ní se vypíše obsah "okénka" do paměti (32 byte). Do dialogového řádku se vypíše text "ENTER COMMAND" a program čeká na příkazy, které se zadávají stiskem některého tlačítka.

Obsah dvojice registrů lze měnit stiskem počátečního písmene registru - např. stiskem 'B' se do dialogového řádku vypíše "BC=". Nyní lze zadat hexadecimální čtyřčísli a provede se jeho přiřazení do registru. Stejným způsobem lze změnit i polohu okénka v paměti - stiskem 'W'.

Další příkazy DEBUGGERu se týkají simulace strojových instrukcí. Stiskem 'EOL' se provede simulace jedné strojové instrukce a to té, na kterou před stiskem ukazoval PC registr a která byla vypsána ve spodní části obrazovky. Stisk klávesy 'C' způsobí simulaci jedné instrukce, stejně, jako 'EOL', pouze je-li simulovanou instrukci CALL, provede se volaný podprogram v reálném čase. POZOR! Následují-li za instrukcí CALL parametry volaného podprogramu, je nutné tento podprogram krokovat jiným způsobem. Při stisku 'U' se zavolá podprogram, jehož adresa je uložena v PC s tím, že všechny registry včetně SP jsou nastaveny na hodnoty, které byly na čelním panelu. Po návratu z tohoto podprogramu (např. instrukcí RET) se

vypisují nové hodnoty registrů, PC zůstává na původní hodnotě. Stisk 'G' způsobí totéž, co 'U' s tím rozdílem, že se na zásobník neukládá návratová adresa, tedy návrat je nutno provést jiným způsobem, například do OS PMD. Další příkazy lze sestavit do následující tabulky:

simulace	rychlá	pomalá
nekonečná	Y	R
do následující instrukce	X	O
do návratové adresy	Z	F

V pravém sloupci tabulky jsou klávesy simulující pomalu – po každé instrukci se znovu vypíší obsahy registrů, zásobníku a okénka. Rychlá simulace znamená, že se smaže obrazovka a obsahy registrů se vypíší až po zastavení simulace. Klávesy v první řádce způsobují nekonečnou simulaci, t.j. simulace není omezena žádnou podmínkou. V druhé řádce jsou klávesy způsobující simulaci do následující instrukce – vhodné při krokování cyklů, nebo podprogramů. Při stisku klávesy ve třetí řádce se přečte nejbližší položka na zásobníku a simulace se zastaví v okamžiku, kdy je obsah PC roven tomuto číslu. Na případných následujících změnách na zásobníku tedy nezáleží. Krokování tímto způsobem je vhodné při potřebě "dokrokování" podprogramu, pokud na zásobníku je jen návratová adresa. Stisknete-li klávesu pro simulaci se shiftem, provádí se podprogramy v reálném čase. Simulaci lze zastavit stiskem 'STOP', pokud se právě neprovádí podprogram v reálném čase. Má-li být simulován kód, který neodpovídá žádné instrukci, je simulace zastavena a vypíše se hlášení "\*\*\* I CAN'T DO STEP \*\*\*". POZOR! V programu není ošetřena simulace instrukce HLT, která tedy může způsobit zastavení mikroprocesoru a porušení obsahu paměti! Ještě jedno upozornění: pro ukazatel zásobníku (SP) je nejvhodnější adresa kolem 7500H.

## SPECIÁLNÍ FUNKCE

Speciální funkce lze navolit kdykoli v "zobrazovacím režimu", t.j. když je v dialogovém řádku napsán text DAMu ( například "ENTER LINE ..." ). Volí se stiskem klávesy 'C-D' následovaným stiskem jedné z klíčových kláves 'K0' - 'K11', popřípadě se shiftem. Stisk klávesy bez shiftu způsobí nastavení příslušného indikátoru na 1, se shiftem nastavení na 0. Jsou to tyto funkce:

'K0' - zapíná automatické řádkování v EDITu ASSEMBLERu. Po odeslání řádky zdrojového textu se vypíše číslo o N vyšší, kde N je stupeň číslování řádek. Funkci je vhodné použít při přepisování, dopisování a vepisování větších částí textu.

'K1' - připojuje tiskovou funkci paralelně k funkci pro výpis na obrazovku. Je automaticky vypínána při vstupu do DEBUGGERu, po návratu se funkce obnoví. Týká se všech výpisů.

'K2' - uplatňuje se jako čítač řádek pro automatické stránkování při tisku. Nastavení na 1 způsobí vypnutí stránkování, nastavení na 0 odstraní a zapne stránkování. Jako jediný je inicializován do stavu 1.

'K3' - Způsobuje vyprazdňování print buffer na konci výpisu.

'K4' - Vypíná výstup na obrazovku (při tisku delších výpisů, kdy by pomalé obrazovkové operace tiskárnu zdržovali).

'K5' - vypíná hlášení "START TAPE, THEN PRESS ANY KEY".

Klávesy 'K6' - 'K11' reprezentují také indikátory, nemají však žádnou funkci. Lze je používat ke sdělování jednoduchých parametrů laděným programům v DEBUGGERu.

## TISKOVÁ FUNKCE

Tisková funkce není zaměřena na konkrétní tiskárnu, ale vytváří bloky o délce 256 byte na MG pásku v nestandardním formátu tak, aby bylo možno s pomocí jednoduchého programu jejich obsah vytisknout. Z toho důvodu je za DAMem vyhrazeno 256 byte bufferu, do něhož se výpis provádí. Po jeho zaplnění se vypíše hláška "START TAPE ... " a pak je blok nahrán. Tvar bloku je tento:

Na začátku je synchronizační záznam šestnáctkrát FFH a šestnáctkrát 77H, za ním následuje bezprostředně zápis 256 byte tisku ukončený kontrolním součtem těchto 256 byte počítaným pomocí instrukce ADD. Poslední blok je do délky doplněn nulovými znaky (00H). Řádky jsou ukončovány vždy pouze znakem 0DH, při automatickém odstránkování se provádí pětkrát za sebou odřádkování. Automatické stránkování se provádí (je-li zapnuto) po každých šedesáti pěti řádkách tisku.

## MOŽNOSTI UŽIVATELSKÝCH ÚPRAV

DAM-0000 je uzpůsoben tak, aby si k němu uživatel mohl připsat vlastní tiskový podprogram. K tomu slouží následující tři vektory, které lze měnit. Uvedené adresy se týkají DAMu nahraného od adresy 0000H.

"Hlavní" vektor je umístěn na adrese 0004H a je volán vždy při tisku jednoho znaku, který dostává v akumulátoru. Standardní funkce je zaplňování print buffer, po zaplnění se volá další vektor, který způsobuje jeho vyprázdnění. Rutina se nachází na adresách 17C7H - 17D2H.

Další vektor je na adrese 0007H a je volán vždy při "násilném" vyprazdňování print bufferu na konci výpisu. Zaplňuje jej kódy 00H tím, že volá výše popsanou rutinu. Vlastní rutina se nachází na adresách 17B5H - 17C6H.

Poslední vektor je na adrese 000AH a ukazuje na podprogram, který nahrává obsah bufferu na pásku. Tento se nachází na adresách 17D3H - 17FFH.

Na adrese 000CH se nachází konstanta určující, po kolika řádkách tisku dojde k odstránkování při tisku.

Uvedené podprogramy jsou umístěny na samém konci DAMu a proto je lze nahradit i podstatně delší rutinou, která může zaplnit i print buffer, který následuje a má délku 256 byte.

Pokud se rozhodnete udělat zásah do tiskových rutin, je nutné opatrně zacházet s vektory a přebytečné z nich vždy namířit na instrukci RET, neboť mohou být volány z mnoha míst hlavního programu (týká se především vektoru pro "násilné" vyprazdňování bufferu).

Systémové proměnné DAMu jsou umístěny "vedle" obrazovky na adresách C230H - C340H. Na adrese C2F0H je lokace programových klíčů 'K0' - 'K11', hodnota 0 je reprezentována kódem 00H, hodnota 1 kódem FFH. Na adrese C335H je umístěn podprogram pro TRANSFER po překladu.

Pokud Váš DAM nebude fungovat tak, jak je zde popsáno, obraťte se laskavě přímo na nás.

4004/482 ZO Svazarmu  
Klub digitální a měřicí techniky  
Lopatecká 615/5  
140 00 Praha 4

Následuje výpis zdrojového textu DAMu v podobě, v jaké jej lze napsat "sám do sebe". Toto má hned dva důvody. Za prvé může tento výpis sloužit jako příklad použití většiny možností DAMU uvedených v předcházejícím manuálu. Druhým důvodem je to, že DAM-0000 vznikl bez použití jiného assembleru a to tak, že jeho první verze byla napsána přímo ve strojovém kódu, verze další byly pak psány ve verzích předchozích. Proto se ve zdrojovém textu na několika místech nacházejí části programu napsané přímo v hexadecimálních kódech, které vznikly při přepisování první verze do zdrojového textu a později již nebyly odstraněny z toho důvodu, že tento způsob zápisu je v některých případech kratší, než běžný zdrojový text. Pro ilustraci lze uvést, že je-li v paměti počítače přítomen DAM i jeho zdrojový text, zbývá přibližně dvě stě byte volné paměti. Proto je pochopitelné, že zdrojový text neobsahuje ani žádné komentáře. Pro lepší orientaci je za výpisem programu uveden ještě výpis návěstí s jejich fyzickými adresami vztahujícími se k DAMu od adresy 0000H.

```

0001 7FEDH ;
0002 @ 00 DAM-0000
0003 XQZAC JNZ 8C40H
0004 LXI H,7000H
0005 LXI D,XQX1-XQX2-1
0006 JMP 8DC4H
0007 *0
0008 *XQZAC
0009 *7000H
0010 @
0011 7000H ;
0012 @N
0013 XQX2 ;
0014 JZ XSTRXX
0015 XERR LXI H,8F22H
0016 SHLD 0C074H
0017 CALL 8A89H
0018 LXI H,8C40H
0019 JMP XGO
0020 XTABLX *ZACATA+2
0021 *WEC1+2
0022 *WEC5+2
0023 *WEC3+2
0024 *C1+1
0025 *C2-1
0026 *C2+1
0027 *C2+3
0028 *C2+5
0029 *C2+7
0030 *C2+9
0031 *C2+11
0032 *C2+13
0033 *C2+15
0034 *C2+17
0035 *C2+19
0036 *C2+21
0037 *C2+23
0038 *C2+27
0039 *C2+29
0040 *C2+31
0041 *C2+33
0042 *C2+35
0043 *C2+37
0044 *C2+39
0045 *C2+41
0046 *C2+43
0047 *C2+49
0048 *C2+51
0049 *C2+53
0050 *C2+55
0051 *C2+57
0052 *C2+59
0053 *C3+2
0054 *C4+2
0055 *C5+2
0056 *C6+2
0057 *C7+2
0058 *BOXER+2

```



```

0059      *C8+2
0060      *C9+2
0061      *C17+2
0062      *OVERAL+2
0063      *NUMBRB+2
0064      *C10+2
0065      *LOUT1+4
0066      *C11+2
0067      *C12+2
0068      *C13+2
0069      *C14+2
0070      *C15+2
0071      *C16+2
0072      *0-1
0073 XSTRXX LHLD 0C072H
0074      DCX  H
0075      MOV  A,M
0076      INX  H
0077      CPI  13
0078      JZ   XSTAR
0079      CALL 80F7H
0080      JNC  XSTRRR
0081 XSTAR  XRA  A
0082 XSTRRR CPI  59H
0083      JNC  XERR
0084      MOV  H,A
0085      MVI  L,0
0086      MVI  A,1
0087      STA  XLEG
0088      SHLD XADDR
0089      SHLD XTRTAD
0090      LXI  H,0
0091      SHLD XLENI
0092      LXI  H,XTABLX
0093      SHLD XTABUK
0094      XRA  A
0095      STA  XSUMA
0096      STA  0C171H
0097      LXI  H,77B0H
0098      SHLD 0C1FAH
0099      LXI  H,0C3C1H
0100      SHLD 0C1FCH
0101      LXI  H,XWAN
0102      SHLD 0C1FEH
0103      LXI  H,0C000H
0104      LXI  D,16
0105 XCLS1A MVI  A,48
0106 XCLS2A MOV  M,D
0107      INX  H
0108      DCR  A
0109      JNZ  XCLS2A
0110      CALL XWAIN
0111      DAD  D
0112      JNC  XCLS1A
0113      LXI  H,XTABLE
0114      CALL XP
0115      MVI  A,0A8H
0116      STA  0C1FAH

```

```

0117 XPOCET =100
0118     LXI  H,XTAB
0119     MVI  B,XPOCET+XPOCET
0120     LXI  D,8000H
0121 XLP0  LDAX D
0122     MOV  M,A
0123     INX  H
0124     CALL XWAIN
0125     INX  D
0126     DCR  B
0127     JNZ  XLP0
0128     MVI  A,XPOCET
0129     LXI  H,XTAB
0130 XLP7  INX  H
0131     PUSH PSW
0132     CALL XBODHL
0133     POP  PSW
0134     INX  H
0135     DCR  A
0136     JNZ  XLP7
0137 XLP1  LXI  H,XTAB
0138     MVI  A,XPOCET
0139 XLP2  PUSH PSW
0140     INX  H
0141     CALL XBODHL
0142     DCX  H
0143     MOV  A,M
0144     CALL XUPR
0145     MOV  M,A
0146     INX  H
0147     MOV  A,M
0148     CALL XUPR
0149     MOV  M,A
0150     CALL XBODHL
0151     INX  H
0152     POP  PSW
0153     DCR  A
0154     JNZ  XLP2
0155     CALL 8C74H
0156     JNZ  XLP1
0157     RET
0158 XUPR  SUI  80H
0159     PUSH PSW
0160     CC   XMINUS
0161     MOV  B,A
0162     INR  B
0163     RRC
0164     RRC
0165     RRC
0166     RRC
0167     ANI  15
0168     ADD  B
0169     ADI  128
0170     CC   XEND
0171     MOV  B,A
0172     POP  PSW
0173     MOV  A,B
0174     RNC

```

```

0175 XMINUS CMA
0176         INR  A
0177         RET
0178 XEND     ADI  80H
0179         MOV  B,A
0180         INR  E
0181         LDAX D
0182         ANI  31
0183         SUB  B
0184         JMP  XMINUS
0185 XBODHL   MOV  A,M
0186         STA  0C170H
0187         DCX  H
0188         MOV  A,M
0189         STA  0C172H
0190         INX  H
0191         JMP  8C7DH
0192 XBYTES   CALL XCOMP
0193         CALL XIN
0194         CZ   XADD
0195         CALL XZAPIS
0196         LDA  XLENI+1
0197         CPI  5
0198         JNZ  XENB
0199         LXI  H,XCODES
0200         SHLD XJUMP+1
0201 XENB     POP  B
0202         POP  D
0203         POP  H
0204         3E;
0205 XWAIN     PUSH PSW
0206         IN   1FH
0207         ANI  2
0208         JNZ  XWWIN
0209         POP  PSW
0210         RET
0211 XWAN      POP  D
0212         POP  H
0213         JMP  XWAIN
0214 XWWIN     PUSH H
0215         PUSH D
0216         PUSH B
0217 XJUMP     JMP  XBYTES
0218 XCODES    LDA  XLEG
0219         DCR  A
0220         STA  XLEG
0221         JNZ  XINST
0222         CALL XIN
0223         PUSH PSW
0224         CALL XZAPIS
0225         POP  PSW
0226         CALL XLEGT
0227         STA  XLEG
0228         STA  XLEE
0229         JMP  XENB
0230 XINST     DCR  A
0231         JNZ  XLOGGER
0232         LDA  XLEE

```

0233		CPI	3
0234	XLOGER	CALL	XIN
0235		CZ	XAD2
0236		CALL	XZAPIS
0237		JMP	XENB
0238	XAD2	MOV	B,A
0239		CALL	XCOMP
0240		MOV	A,B
0241		RZ	
0242	XADD	CPI	0BDH
0243		JZ	XADD1
0244		CPI	1AH
0245		RNC	
0246		MOV	B,A
0247		LDA	XBASE
0248		ADD	B
0249		RET	
0250	XADD1	MOV	B,A
0251		LDA	XBASE
0252		SUB	B
0253		CMA	
0254		INR	A
0255		RET	
0256	XCOMP	LHLD	XLENI
0257		XCHG	
0258		LHLD	XTABUK
0259		MOV	A,M
0260		CMP	E
0261		RNZ	
0262		INX	H
0263		MOV	A,M
0264		CMP	D
0265		RNZ	
0266		INX	H
0267		SHLD	XTABUK
0268		RET	
0269	XIN	IN	1EH
0270		PUSH	PSW
0271		MOV	B,A
0272		LDA	XSUMA
0273		ADD	B
0274		STA	XSUMA
0275		POP	PSW
0276		RET	
0277	XLEGT	MVI	B,0
0278		MOV	C,A
0279		CPI	0C3H
0280		JZ	XB3
0281		CPI	0CDH
0282		JZ	XB3
0283		ANI	0EFH
0284		CPI	22H
0285		JZ	XB3
0286		CPI	2AH
0287		JZ	XB3
0288		ANI	0CFH
0289		CPI	1
0290		JZ	XB3

0291	ANI	0C7H
0292	CPI	0C2H
0293	JZ	XB3
0294	CPI	0C4H
0295	JZ	XB3
0296	MOV	A,C
0297	ANI	0F7H
0298	CPI	0D3H
0299	JZ	XB2
0300	ANI	0C7H
0301	CPI	6
0302	JZ	XB2
0303	CPI	0C6H
0304	JZ	XB2
0305	JMP	XB1
0306	XB3	INR B
0307	XB2	INR B
0308	XB1	INR B
0309	MOV	A,B
0310	RET	
0311	XTABUK	#2
0312	XLENI	#2
0313	XADDR	#2
0314	XSUMA	#1
0315	XLEG	#1
0316	XLEE	#1
0317	XP	MOV B,M
0318		INX H
0319	XLOOP1	MOV C,M
0320		INX H
0321		MOV A,M
0322		DCR A
0323		RLC
0324		STA 0C170H
0325		INX H
0326		MOV A,M
0327		DCR A
0328		RLC
0329		STA 0C172H
0330		INX H
0331	XLOOP2	MOV A,M
0332		DCR A
0333		RLC
0334		STA 0C173H
0335		INX H
0336		MOV A,M
0337		DCR A
0338		RLC
0339		STA 0C174H
0340		INX H
0341		PUSH H
0342		PUSH B
0343		CALL 8CD0H
0344		POP B
0345		POP H
0346		DCR C
0347		JNZ XLOOP2
0348		DCR B

```

0349      JNZ  XLOOP1
0350      RET
0351  XTABLE 33;
0352      030A840A0F140F140505A8051E051E0A280A280F1E0F;
0353      06BC053205320A370A320A320F3C0F06;
0354      468F4605500550;
0355      0A460A4B0A500F015D05620A05F8056E056E0A780A78;
0356      0F6E0F032D2D232323282D300323281E231923;
0357      2D3203192319281E282F34041E28232D41412332232D;
0358      0423321E374141119371E370319371432142D1E310514;
0359      2D19321E321E2D192D1E2F05192D14281423191E1E1E;
0360      23230A37373C324141373732372D322D23321E;
0361      371E3C234141013C233C3205372D3223372337323232;
0362      322306461E46374141461E551E5523502B055523;
0363      4B234B285028502D4B3303502D4B2D4B374141014B37;
0364      463705643769374141643764234141056423;
0365      5F234B3748355F1E5F23035F1E6E1E6E236926026E;
0366      2369236937051864145F144B4141194B144B03;
0367      194B195A1B551E54051C551E5A1E4B4141234B1E4B05;
0368      234B235F2D552D561F64235F041F641C5F1B5F1864;
0369      1C610632642D642D50324B4141374B324B0437;
0370      4B3C5041413C643C50053C643764375F325F3264;
0371      345F053752325A32503750375A325A;
0372      075064505F4B5A4B6446644141464B466402464B50;
0373      4B414105504B5550555505A555F515903555F556450;
0374      644D5C054C50505550504B504B555055;
0375      0553575F645F4B41416E4B5F4B056E4B6E506450;
0376      645569556453056955695A645A645F6E5F675A;
0377      026E5F6E645F64030AF20A780F780F7D0419;
0378      73197D1E7D1E731973042873287D2D7D2D7328;
0379      730337F237783C783C7D014BF2467D0355F25578;
0380      5A785A7D066478697869736473647D697D;
0381      697805F273787378787378737D787D;
0382  XZAPIS LHL D XADDR
0383      MOV  M,A
0384      INX  H
0385      SHLD XADDR
0386      LHL D XLENI
0387      INX  H
0388      SHLD XLENI
0389      MOV  A,H
0390      CPI  18H
0391      RNZ
0392      CALL 8E0EH
0393      LDA  XSUMA
0394      MOV  B,A
0395      IN   1EH
0396      CMP  B
0397      JNZ  XERR
0398  XST1  LXI  H,0
0399  XGO   PUSH H
0400      LXI  H,0E1D1H
0401      SHLD 0C1FDH
0402      MVI  A,0C9H
0403      STA  0C1FFH
0404      LXI  H,7000H
0405  XRUN  STC
0406      RNC

```

```

0407      XRA  A
0408      MOV  M,A
0409      INX  H
0410      JMP  XRUN
0411 XBASE  =XST1+2
0412 XTRTAD  =XST1+1
0413 XQX1    ;
0414      @
0415 XTAB     =XP
0416 0       ;
0417      @N
0418 ZACATA  JMP  START
0419 WEC1    JMP  WECCO
0420 WEC5    JMP  WROT1
0421 WEC3    JMP  WECOUT
0422 PAGE   ,65;
0423 LOCDM   'DAM  ';
0424 C1      *ASSEMB
0425      'OS '408CFF;
0426 DAMWRD  0D'DAM-0000 UNIVERSAL MONITOR'0D;
0427      0D'BY 4004/482 SVAZARM ORGANISATION'0D;
0428      'AUTHOR K.SUHAJDA'0D0D00;
0429 TAB1     4E4FD04A4DD04AC34ADA4A4EC34A;
0430      4EDA4A50C54A50CF4AD04ACD4D4FD64D56C9;
0431      494ED24443D2494ED84443D8535441D8;
0432      4C4441D84C484CC453484CC44C44C153;
0433      54C1414EC14F52C15852C1434DD04144;
0434      C45355C25342C24144C3414EC94F52C9;
0435      5852C95342C95355C94350C94144C941;
0436      43C943414CCC43DA434EDA43C3434EC3;
0437      4350C54350CF43D043CD5245D452DA52;
0438      4EDA52C3524EC35250C55250CF52D052;
0439      CD434DC1434DC3504FD0505553C84441;
0440      C4584348C7585448CC524CC35252C352;
0441      41CC5241D245C944C9535048CC504348;
0442      CC4441C15253D449CE4F55D45354C348;
0443      4CD44C58C923;
0444 TAB2     B0B1B2B3B4B5B6B723C2C4C85053;
0445      D723C2C4C853D023C2C3C4C5;
0446      C8CCDC123C2C423;
0447 TAB3     FF764D10FF000110FFC30230FFDA;
0448      0330FFCA0430FFD20530FFC20630FFEA0730;
0449      FFE20830FFF20930FFFA0A30C0400B12;
0450      3330C7060C2133C7040D1133C7050E11;
0451      33CF030F1124CF0B101124EF02111144;
0452      EF0A121144FF2A1330FF221430FF3A15;
0453      30FF321630F8A0171130F8B0181130F8;
0454      A8191130F8B81A1130F8801B1130F890;
0455      1C1130F8981D1130F8881E1130FFE61F;
0456      20FFF62020FFEE2120FFDE2220FFD623;
0457      20FFFE2420FFC62520FFCE2620FFCD27;
0458      30FFCC2830FFC42930FFDC2A30FFD42B;
0459      30FFEC2C30FFE42D30FFFE42E30FFFC2F;
0460      30FFC93010FFC83110FFC03210FFD833;
0461      10FFD03410FFE83510FFE03610FFF037;
0462      10FFF83810FF2F3910FF3F3A10CFC13B;
0463      1114CFC53C1114CF093D1124FFEB3E10;
0464      FFE33F107F074010FF0F4110FF174210;

```

```

0465      FF1F4310FFFB4410FFF34510FFF94610;
0466      FFE94710FF274810C7C7491103FFDB4A;
0467      20FFD34B20FF374C10CF014E31240000;
0468 LOCFLG 40010480;
0469      *ROZBEH
0470 C2      ;
0471 ASSTAB  *EDAD
0472      *EDITAS
0473      *TRAD
0474      *TRANAS
0475      *SAAD
0476      *SAVEAS
0477      *VERA
0478      *VERIFY
0479      *LOAD
0480      *LOADAS
0481      *MEGD
0482      *MERGE
0483      *Ø
0484      *ROZBEH
0485 MONTAB  *EDAD
0486      *EDITMO
0487      *FIAD
0488      *FIND
0489      *LIAD
0490      *LISTMO
0491      *DIAD
0492      *DISAMO
0493      *Ø
0494      408C;
0495 BASTAB  *ASAD
0496      *ASSEMB
0497      *MOAD
0498      *MONITR
0499      *DEAD
0500      *DEBUGR
0501      *Ø
0502 DEAD   'DEBUGGER'ØD;
0503 MOAD    'MONITOR'ØD;
0504 VERA    'VERIFY'ØD;
0505 DIAD    'DIS';
0506 ASAD    'ASSEMBLER'ØD;
0507 FIAD    'FIND'ØD;
0508 LIAD    'LIST'ØD;
0509 EDAD    'EDIT'ØD;
0510 TRAD    'TRANSLATE'ØD;
0511 LOAD    'LOAD'ØD;
0512 MEGD    'MERGE'ØD;
0513 SAAD    'SAVE'ØD;
0514 STAP    'START TAPE,THEN PRESS ANY KEY'ØD;
0515 EAOQ    'ENTER ADDRESS OR ''Q'' TO QUIT'ØD;
0516 PSOQ    'PRESS ''STOP'' TO STOP LISTING'ØD;
0517 PETC    'PRESS ''EOL'' TO CONTINUE, ''Q'' TO QUIT'ØD;
0518 ECOM    'ENTER COMMAND'ØD;
0519 PSTO    'PRESS ''STOP''ØD;
0520 WARN    '** I CAN'T DO STEP **'ØD;
0521 ELOQ    'ENTER LINE OR ''Q'' TO QUIT'ØD;
0522 LIER    '** LINE INPUT ERROR **'ØD;

```



```

0523 ESNQ      'ENTER SEGMENT NUMBER OR ''Q'' TO QUIT'0D;
0524 ERSO      '** FILE NOT CLOSED **'0D;
0525 NOFI      '** NO FILE **'0D;
0526 ENAM      'ENTER NAME'0D;
0527 ITOH      '** IT'S TOO HIGH **'0D;
0528 BORG      '** BAD ORG **'0D;
0529 OPTS      'OPTIONS ?'0D;
0530 TADD      'TABLE ADDRESS ?'0D;
0531 SADD      'START ADDRESS ?'0D;
0532 TRND      'TRANSFER ?'0D;
0533 RLAB      'RUN LABEL ?'0D;
0534 WDAD      'WD';
0535 REGERS     'AFBCDEHLSPPC->MZ H P C';
0536 START     LXI  H,OUTDAM
0537           PUSH H
0538 CLS       LXI  H,0C000H
0539 C3        LXI  D,16
0540 CLSLP1    MVI  A,30H
0541 CLSLOP    MOV  M,D
0542           INX  H
0543           DCR  A
0544           JNZ  CLSLOP
0545           DAD  D
0546           JNC  CLSLP1
0547           RET
0548           NOP  VOLNY BYTE
0549 DEBOPS     3E20CD0085790E08;
0550 DEBOP2     17F53E00CE30CD0085F10D;
0551           JNZ  DEBOP2
0552           RET
0553 VYPSA1     SHLD 0C03EH
0554 VYPS1      1A13CD00851A13C30085;
0555 VYPS4      PUSH D
0556           CALL VYPS2
0557           POP  D
0558           PUSH H
0559           PUSH D
0560 C4         LXI  D,1400H
0561           19D1223EC03E28E1CD0085;
0562           CALL VYPS1
0563           3E29CD00853E3DC30085;
0564 VYPS2      CALL VYPSA1
0565           3E3DC30085;
0566 DEBREG     LXI  H,0C704H
0567           XRA  A
0568           STA  FTISK
0569           STA  FPRIN
0570           LXI  D,REGERS
0571 C5         LXI  B,400H
0572           CALL VYPS2
0573           DAD  B
0574           CALL VYPS4
0575           DAD  B
0576           CALL VYPS4
0577           DAD  B
0578           CALL VYPS4
0579           DAD  B
0580           CALL VYPS2

```

0581	E51B1B211FE5;
0582	CALL VYPSA1
0583	POP H
0584	DAD B
0585	CALL VYPS2
0586	LXI H,0E521H
0587	CALL VYPSA1
0588	LXI H,0C515H
0589	CALL VYPSA1
0590	CALL VYPS1
0591	CALL VYPS1
0592	CALL VYPS1
0593	LXI H,0C707H
0594	LXI D,REGLOC
0595	MVI B,6
0596	DEBZNO C5D5E5223EC0EB5E2356;
0597	CALL DEOUT
0598	MOV C,D
0599	CALL DEBOPS
0600	MOV C,E
0601	CALL DEBOPS
0602	3E20CD00852A3EC07A;
0603	CALL KYU
0604	7B23;
0605	CALL KYU
0606	E124242424D11313C105;
0607	JNZ DEBZNO
0608	LXI D,REGLOC+2
0609	2109DF0E03;
0610	DEBZN2 D5E5223EC0EB5E23561ACD2581;
0611	3E20CD00851A2A3EC0C5;
0612	CALL KYU
0613	C1E124242424;
0614	D113130D;
0615	JNZ DEBZN2
0616	LHLD REGLOC+8
0617	EB2123E50E10;
0618	DEBC3C 223EC0C51A4F131ACD258179CD;
0619	2581C11325250D;
0620	JNZ DEBC3C
0621	2100EB223EC00E30;
0622	DEBLOP 3E20CD00850D;
0623	JNZ DEBLOP
0624	SHLD 0C03EH
0625	LHLD REGLOC+10
0626	SHLD PROM2
0627	CALL DEBDIS
0628	2100F6223EC0;
0629	LHLD WINDOW
0630	SHLD PROM2
0631	LXI H,0ED00H
0632	C6 LXI B,300H
0633	CALL LIMEX1
0634	LIMEX1 CALL LIMEX
0635	LIMEX DAD B
0636	SHLD 0C03EH
0637	PUSH H
0638	PUSH B

```

0639      LHLD PROM2
0640      CALL LIMEDI
0641      POP  B
0642      POP  H
0643      RET
0644 DOSTEP LHLD REGLOC+10
0645      7EFEC0;
0646      JC   DEBSET
0647      CPI  0E9H
0648      JZ   DEBCHL
0649      CPI  0C9H
0650      JZ   DEBRET
0651      CPI  0C3H
0652      JZ   DEBJMP
0653      CPI  0CDH
0654      JZ   DEBCAL
0655      2FE6C7;
0656      JZ   DEBRST
0657      RRC
0658      JNC  DEBSET
0659      ANI  3
0660      JZ   DEBSET
0661      7E0F0F0F0FE603;
0662      INR  A
0663      LXI  D, LOCFLG-1
0664 LOFG  INX  D
0665      DCR  A
0666      JNZ  LOFG
0667      1A4F7EE608;
0668      LDA  REGLOC
0669      JNZ  DEBUAL
0670      CMA
0671 DEBUAL ANA  C
0672      JZ   DEBSET
0673      7EE607;
0674      JZ   DEBRET
0675      CPI  2
0676      JZ   DEBJMP
0677 DEBCAL LDA  WARI+1
0678      ANA  A
0679      JNZ  DEBSET
0680      EBD5131313;
0681      LHLD REGLOC+8
0682      2B722B73;
0683      SHLD REGLOC+8
0684      POP  H
0685 DEBJMP 235E2356;
0686 DEBJDE XCHG
0687 DEBJHL SHLD REGLOC+10
0688      RET
0689 DEBCHL LHLD REGLOC+6
0690      JMP  DEBJHL
0691 DEBRET LHLD REGLOC+8
0692      5E235623;
0693      SHLD REGLOC+8
0694      JMP  DEBJDE
0695 DEBWAR MOV  A,H
0696      POP  H

```

```

0697      STA  WARI
0698      RET
0699 DEBRST XCHG
0700      LHLD REGLOC+8
0701      132B722B731B1AE638;
0702      SHLD REGLOC+8
0703      6F2600;
0704      JMP  DEBJHL
0705 DEBSET MOV  C,M
0706      PUSH H
0707      CALL ASKIN
0708      JC   DEBWAR
0709      2323237E;
0710 C7    210000F0F0F0FE60347;
0711      SHLD DEBOX2
0712      LXI  D,DEBOX1
0713      POP  H
0714 DEBSTA 7E12231305;
0715      JNZ  DEBSTA
0716      SHLD REGLOC+10
0717 BOXER  21000039;
0718      SHLD PROM0
0719      LXI  SP,REGLOC
0720      F1C1D1E1E1F9;
0721      LHLD REGLOC+6
0722 DEBOX1 NOP
0723 DEBOX2 *0
0724      SHLD REGLOC+6
0725      DCX  SP
0726      DCX  SP
0727      POP  H
0728      PUSH PSW
0729      XTHL
0730      POP  PSW
0731      SHLD REGLOC
0732 C8    21000039;
0733      SHLD REGLOC+8
0734      LXI  SP,REGLOC+6
0735      PUSH D
0736      PUSH B
0737      LHLD PROM0
0738      SPHL
0739      RET
0740 DEBUGR CALL CLS
0741      MVI  A,255
0742      STA  0C137H
0743      LDA  FTISK
0744      STA  PFX1
0745 DEBR1  CALL DEBREG
0746      XRA  A
0747      STA  WARI+1
0748      MOV  H,A
0749      MOV  L,H
0750      SHLD FLAGS
0751      LDA  WARI
0752      ANA  A
0753      JZ   DEBCAA
0754 WWARNI SHLD WARI

```

```

0755      LXI  H,WARN
0756      CALL ZOBEEP
0757 DEBCAA LXI  H,ECOM
0758      CALL ZOBUNB
0759      2100FB223EC0;
0760      JZ   ROZBE1
0761      CPI  89H
0762      JNZ  DEBOTH
0763      CALL DOSTEP
0764      JMP  DEBR1
0765 DEBOTH CPI  'C'
0766      JNZ  DEBOT2
0767      STA  WARI+1
0768      JMP  DEBOTH-6
0769 DEBOT2 CPI  ' '
0770      JC   DEBBBB
0771      STA  WARI+1
0772      SUI  32
0773 DEBBBB CPI  'X'
0774      JC   DEBCCC
0775      STA  FLAGS
0776 DEBCCC CPI  'R'
0777      JZ   ISPOL
0778      CPI  'E'
0779      JZ   EDITM1
0780      CPI  'Y'
0781      JZ   ISPOL
0782      LHLD REGLOC+8
0783      MOV  E,M
0784      INX  H
0785      MOV  D,M
0786      XCHG
0787      SHLD PROM1
0788      STA  FLAGS+1
0789      CPI  'Z'
0790      JZ   ISPOL
0791      CPI  'F'
0792      JZ   ISPOL
0793      PUSH PSW
0794      LHLD REGLOC+10
0795      MOV  C,M
0796      PUSH H
0797      CALL ASKIN
0798      2323237EE6F00F0F0F16005FE119F1;
0799      SHLD PROM1
0800      CPI  'O'
0801      JZ   ISPOL
0802      CPI  'X'
0803      JZ   ISPOL
0804      CPI  'G'
0805      JZ   DEBURE+1
0806      CPI  'U'
0807      JNZ  DEBDDD
0808      MVI  A,0CDH
0809 DEBURE LXI  H,0C33EH
0810      LHLD REGLOC+10
0811      SHLD DEBOX2
0812      STA  DEBOX1

```

```

0813      CALL CLS
0814      CALL BOXER
0815      JMP  DEBR1
0816 DEBDDD LXI  H,WDAD-2
0817      MOV  B,A
0818 DE1    23237EFE'-';
0819      JZ   DEBR1
0820      CMP  B
0821      JNZ  DE1
0822      XCHG
0823      LHLD 0C030H
0824      D51A7723131A77CDEE89CD6C893E'=';
0825      3234C1CDEE8B2A78C0232323CD0981E1;
0826      JC   DEBR1
0827      LXI  B,WINDOW-WDAD
0828      DAD  B
0829      MOV  M,E
0830      INX  H
0831      MOV  M,D
0832      JMP  DEBR1
0833 ISPOL  LDA  FLAGS
0834      ANA  A
0835      CNZ  CLS
0836      LXI  H,PSTO
0837      CALL ZOBRAZ
0838 ISPOL1 CALL DOSTEP
0839      LDA  WARI
0840      ANA  A
0841      JNZ  DEBR1
0842      LDA  FLAGS
0843      ANA  A
0844      CZ   DEBREG
0845      CALL 8C74H
0846      JZ   DEBR1
0847      LDA  FLAGS+1
0848      ANA  A
0849      JZ   ISPOL1
0850      LHLD REGLOC+10
0851      XCHG
0852      LHLD PROM1
0853      CALL CODEHL
0854      JNZ  ISPOL1
0855      JMP  DEBR1
0856 OUTDAM ;
0857      LXI  H,PROM0
0858      XRA  A
0859 QWERTZ  3600233D;
0860      JNZ  QWERTZ
0861      STA  0C03EH
0862      LXI  H,DAMWRD
0863      MOV  A,M
0864 OUTDA1 CALL 8500H
0865      INX  H
0866      MOV  A,M
0867      ANA  A
0868      JNZ  OUTDA1
0869      MVI  A,25H
0870      STA  0C136H

```

0871		CALL	88A3H
0872		MVI	A, 10
0873		STA	PROM4
0874		MVI	A, 255
0875		STA	FPAGE
0876		LXI	H, KONECA
0877		SHLD	PROM5
0878		LXI	H, PRBUFF
0879		SHLD	WECTOR
0880	ROZBE1	LDA	PFX1
0881		STA	FTISK
0882	ROZBEH	LXI	H, BASTAB
0883	ROVINA	SHLD	PROM1
0884		LXI	SP, STACK
0885		CALL	8A73H
0886	ROZB1	LHLD	PROM1
0887		SHLD	PROM2
0888	ROZB2	LHLD	PROM2
0889			7E5F23B6;
0890		JZ	ROZB1
0891			56232323;
0892		SHLD	PROM2
0893		XCHG	
0894		CALL	ZOBUNB
0895		JZ	ROZB3
0896		CPI	89H
0897		JNZ	ROZB2
0898		LHLD	PROM2
0899	ROZB4		2B562B5EEBE9;
0900	ROZB3	LHLD	PROM1
0901		JMP	ROZB4
0902	TAPES	LDA	FSTAP
0903		ANA	A
0904		RNZ	
0905		CALL	88A3H
0906		LXI	H, STAP
0907	ZOBAS	CALL	ZOBUNB
0908		RNZ	
0909	ASSEMB	CALL	WEC5
0910		LHLD	PROM5
0911		CALL	TEXTP
0912		LXI	H, ASSTAB
0913		JMP	ROVINA
0914	ODRD	MVI	A, 13
0915		JMP	S
0916	ZOBEEP	PUSH	H
0917		CALL	88A3H
0918		POP	H
0919	ZOBUNB	SHLD	0C074H
0920		CALL	8A8CH
0921		LDA	0C134H
0922		CPI	08AH
0923		JNZ	CONQ
0924		CALL	8A8CH
0925		LDA	0C134H
0926		CPI	0D0H
0927		JC	CONQ
0928		CPI	0ECH

0929	JNC	CONQ
0930	CPI	0E0H
0931	MVI	B,0
0932	JNC	CANZ
0933	DCR	B
0934	CANZ	LXI H,PFXE
0935	ANI	15
0936	MOV	E,A
0937	MVI	D,0
0938	DAD	D
0939	MOV	M,B
0940	JMP	ZOBUNB+3
0941	LLIST	LXI H,PSOQ
0942	CALL	ZOBRAZ
0943	LLIST1	MVI A,0C3H
0944	STA	DEBOX1
0945	CALL	ODRD
0946	LHLD	PROM2
0947	CALL	DEBOX1
0948	SHLD	LOC3
0949	CALL	8C74H
0950	JNZ	LLIST1
0951	LXI	H,PETC
0952	CALL	ZOBUNB
0953	CPI	89H
0954	JZ	LLIST
0955	CALL	ODRD
0956	CALL	WEC5
0957	CONQ	LDA 0C134H
0958	CPI	'Q'
0959	RET	
0960	ZOBRAZ	2274C0CD2D893A3CC15F16002A3AC119;
0961	ZOBLAB	3620233E301CBB;
0962	JNZ	ZOBLAB
0963		2A3AC1CD5888AF323CC1C9;
0964	LISTMO	LXI H,LIMEDE
0965	CALL	LISTER
0966	MONITR	LXI H,MONTAB
0967	JMP	ROVINA
0968	CPIR	PUSH B
0969		PUSH D
0970		PUSH H
0971	CPIR1	LDAX D
0972		CMP M
0973	JNZ	CPIR2
0974	INX	H
0975	INX	D
0976	DCX	B
0977	MOV	A,B
0978	ORA	C
0979	JNZ	CPIR1
0980	CPIR2	POP H
0981		POP D
0982		POP B
0983		RET
0984	FIND	LXI H,EAQO
0985		CALL ZOBUNB
0986	FINQ	JZ MONITR



```

0987      CALL COMPRI
0988      JC  FINQ
0989      LXI  H,FINF
0990      SHLD DEBOX2
0991      MOV  H,B
0992      MOV  L,C
0993      SHLD PROM2
0994      CALL LLIST
0995      JMP  FINQ
0996  FINF  MOV  B,H
0997      MOV  C,L
0998      LHLD 0C078H
0999      XCHG
1000      LHLD LOC3
1001  FINX  CALL 8C74H
1002      RZ
1003      INX  H
1004      CALL CPIR
1005      JNZ  FINX
1006      CALL HLOUT
1007      XCHG
1008      RET
1009  LISTER SHLD DEBOX2
1010  LISING LXI  H,EA0Q
1011      CALL ZOBUNB
1012  ACONQ  RZ
1013      CALL EDISX
1014      JC  LISING
1015      XCHG
1016      SHLD PROM2
1017      CALL LLIST
1018      JMP  ACONQ
1019  LIMEDE CALL HLOUT
1020      EB0E08E5;
1021  MEMLI2 MVI  A,32
1022      CALL S
1023      MOV  A,M
1024      CALL S1
1025      230D;
1026      JNZ  MEMLI2
1027      3E20323EC0E13C;
1028      CALL S
1029      0E08;
1030  MEMLI3 7E23;
1031      CALL KYU
1032      DCR  C
1033      JNZ  MEMLI3
1034      SHLD PROM2
1035      MVI  A,33
1036      JMP  S
1037  KYU    CPI  32
1038      JC  KYV
1039      CPI  60H
1040      JC  S
1041  KYV    MVI  A,'.'
1042      JMP  S
1043  EDISX  CDEE8B2A78C0C30981;
1044  EDITM1 LDA  PFX1

```

```

1045          STA  FTISK
1046 EDITMO LXI  H,EA0Q
1047          CALL ZOBUNB
1048 EDIWCQ JZ   MONITR
1049 EDITEX CALL COMPRI
1050          JC   EDIWCQ
1051          INR  C
1052          DCR  C
1053          LHLD LOC3
1054          INX  H
1055          XCHG
1056          JZ   MEMOUT
1057          LHLD 0C078H
1058          CALL LDIR
1059          CALL CURSOU
1060          JMP  EDITEX
1061 CURSOU 2A30C07ACD3B81237BCD3B81CD;
1062        EE89CD6C89CD6C89CD6C89C35588;
1063 ERINDA LXI  H,807EH
1064 ERAD1  CALL ZOBEEP
1065          STC
1066          RET
1067 ERAD  LXI  H,8092H
1068          JMP  ERAD1
1069 COMPRI CALL EDISX
1070          JC   ERAD
1071          XCHG
1072          DCX  H
1073          SHLD LOC3
1074 ASSCOM LHLD 0C078H
1075          XCHG
1076 C9     LXI  B,0
1077 COMPRI MVI  A,32
1078          CALL OVER
1079          MOV  A,M
1080          CPI  13
1081          RZ
1082          CPI  0DDH
1083          JZ   ERINDA
1084          CPI  ';'
1085          RZ
1086          CPI  ''''
1087          JZ   STRING
1088          PUSH B
1089          CALL 80F7H
1090          POP  B
1091          JC   ERINDA
1092          STAX D
1093          INX  D
1094          INX  H
1095          INX  B
1096          JMP  COMPR1
1097 MEMOUT PUSH D
1098          CALL CURSOU
1099          POP  D
1100          LHLD 0C030H
1101 C17    LXI  B,3
1102          CALL 81EBH

```

```

1103      JMP  EDITEX
1104 STRING 237EFE0D;
1105      JZ   ERINDA
1106      CPI  0DDH
1107      JZ   ERINDA
1108      CPI  27H
1109      JNZ  EDIST2
1110      INX  H
1111      CMP  M
1112      JNZ  COMPR1
1113 EDIST2 STAX D
1114      INX  D
1115      INX  B
1116      JMP  STRING
1117 DISAMO LXI H,DISASM
1118      CALL LISTER
1119      JMP  MONITR
1120 BWORD  LXI H,TAB1
1121 BPRVE  DCR  B
1122      RZ
1123 BPRVE1 7E1723;
1124      JNC  BPRVE1
1125      JMP  BPRVE
1126 BTBLE  LXI H,TAB2
1127 BTBLE1 DCR  B
1128      RZ
1129 BTBLE2 7E23FE23;
1130      JNZ  BTBLE2
1131      JMP  BTBLE1
1132 ASKIN  LXI H,TAB3
1133 ASKIN1 7EA73FC8A123BE2BC82323237EE60F3C;
1134 ASKIN2 INX  H
1135      DCR  A
1136      JNZ  ASKIN2
1137      JMP  ASKIN1
1138 PRINH1 INX  H
1139      CALL CHOUT
1140 PRINHL 7E07A71F;
1141      JNC  PRINH1
1142 CHOUT  JMP  S
1143 DISASM ;
1144 DEBDIS CALL HLOUT
1145      EB3E05323EC04E23;
1146      SHLD PROM2
1147 DIAID  CALL ASKIN
1148      E5F53E01;
1149      JC   DISUNK
1150      F1F52323237E1F1F1F1FE60F;
1151 DISUNK LHLD PROM2
1152      DCX  H
1153      MOV  D,A
1154 DISCO2 MOV  A,M
1155      CALL NUOUT
1156      INX  H
1157      DCR  D
1158      JNZ  DISCO2
1159      3E0C323EC0F1E179;
1160      JC   NUOUT

```

```

1161      B6AE4FC52323E546;
1162      CALL BWORD
1163      CALL PRINHL
1164      3E11323EC0E1C1237EE60F;
1165      JZ  DISNOI
1166      PUSH H
1167  DISPPP F5C5237EE60F47047907;
1168  DISPP  RRC
1169      DCR  B
1170      JNZ  DISPP
1171      E607E5F57EE6F00F0F0F0F4704;
1172      CALL BTBLE
1173      F14704;
1174      CALL BPRVE
1175      CALL PRINHL
1176      E1C1F1FE02;
1177      JNZ  DISGON
1178      MVI  A,2CH
1179      CALL CHOUT
1180      XRA  A
1181      JMP  DISPPP
1182  DISGON E17EE6F0DE10C83E2C;
1183      CALL CHOUT
1184  DISNOI 7EE6F0DE10C8E610;
1185      LHLD PROM2
1186      MOV  A,M
1187      INX  H
1188      JZ   DISTWO
1189      SHLD PROM2
1190      JMP  NUOUT
1191  DISTWO 577E23;
1192      SHLD PROM2
1193      CALL NUOUT
1194      MOV  A,D
1195  NUOUT  JMP  S1
1196  EDITAN ;
1197      LDA  FLINE
1198      ANA  A
1199      JZ   EDITAS
1200      LHLD LNOP
1201      LDA  PROM4
1202      MOV  E,A
1203      MVI  D,0
1204      DAD  D
1205      CALL LOUT1
1206      CALL CURSOU
1207      JMP  ASSED3
1208  EDITE  CDEE8B2A78C0E5545DCD3789EB;
1209      362023360DE1C9;
1210  EDITAX CALL ODRD
1211      CALL WEC5
1212  EDITAS LXI  H,ELOQ
1213  ASSED2 LXI  SP,STACK
1214      CALL ZOBAS
1215  ASSED3 XRA  A
1216      STA  0C137H
1217      CALL EDITE
1218      CALL ASLINE

```

1219		XCHG
1220		SHLD LNOP
1221		XCHG
1222		7BB27E;
1223		JZ LIIER1
1224		CPI 'S'
1225		JNZ ASSOTH
1226		MOV A,E
1227		STA PROM4
1228		JMP EDITAS
1229	LIIER1	CPI 'N'
1230		JNZ LIIER2
1231		LHLD PROM5
1232	LIERN	CALL YES
1233		JMP EDITAS
1234	LIIER2	CPI 'T'
1235		JNZ LIIER3
1236		INX H
1237		CALL NUMIN
1238		XCHG
1239		CALL TLES3
1240		JC LIIERR
1241		MOV A,H
1242		RLC
1243		JC LIIERR
1244		SHLD PROM5
1245		CALL TEXTP
1246		JMP EDITAS
1247	TLES1	PUSH D
1248		PUSH B
1249		XCHG
1250		LHLD LOC1
1251		MOV C,L
1252		MOV B,H
1253		LHLD LOC2
1254		XCHG
1255		CALL UNIT
1256		POP B
1257		POP D
1258		RC
1259	TLES2	PUSH D
1260		PUSH B
1261		XCHG
1262		LHLD PROM5
1263		MOV C,L
1264		MOV B,H
1265		LHLD PROM6
1266		XCHG
1267		CALL UNIT
1268		POP B
1269		POP D
1270		RC
1271	TLES3	PUSH D
1272		PUSH B
1273		LXI B,ZACATA
1274		LXI D,KONECA-1
1275		CALL UNIT
1276		C1D1D8D5C5;

```

1277      LXI  D,CEND
1278      LXI  B,STACK-128
1279      CALL UNIT
1280      C1D1C9;
1281 UNIT  INX  D
1282      CALL CODEHL
1283      CMC
1284      RNC
1285      MOV  E,C
1286      MOV  D,B
1287 CODEHL MOV  A,D
1288      CMP  H
1289      RNZ
1290      MOV  A,E
1291      CMP  L
1292      RNZ
1293      STC
1294      RET
1295 ASSOTH CPI  'E'
1296      JZ   ASLINO
1297      CPI  'L'
1298      JZ   LISTAS
1299      PUSH H
1300      CALL ASLIDE
1301      E37E23FE'D';
1302      JZ   AMT
1303      CPI  '-'
1304      JZ   DEL
1305      XTHL
1306      CPI  'N'
1307      JZ   LIERN
1308      XTHL
1309      FE207E;
1310      JNZ  LIIERR
1311      CPI  0DH
1312      JNZ  ASSNEM
1313 LIIERR LXI  H, LIER
1314 LIAZ   PUSH H
1315      CALL 88A3H
1316      POP  H
1317      JMP  ASSED2
1318 DEL    CALL ASLINE
1319      XTHL
1320      PUSH H
1321      CALL ASLIDE
1322      CNC  NEXL
1323      POP  D
1324      CALL CODEHL
1325      JNC  LIIERR
1326      XTHL
1327      MOV  A,M
1328      CPI  'M'
1329      JZ   MOVE
1330      CPI  'D'
1331      JNZ  LIIERR
1332      POP  H
1333      CALL MAZ
1334      JMP  EDITAS

```

```

1335 MOVE    INX    H
1336         XTHL
1337         MOV    A, L
1338         SUB    E
1339         MOV    C, A
1340         MOV    A, H
1341         SBB    D
1342         MOV    B, A
1343         LHLD   PROM6
1344         INX    H
1345         PUSH   H
1346         DAD    B
1347         CALL   TLES3
1348         JC     MEOVS
1349         POP    H
1350         CALL   LDIRX
1351         XCHG
1352         MVI    M, 0FDH
1353         POP    H
1354         CALL   ASLINE
1355         CALL   ASLIDE
1356         XCHG
1357 MOVE1     ;
1358         LHLD   PROM6
1359         INX    H
1360         MOV    A, M
1361         CPI    0FDH
1362         JZ     EDITAS
1363         PUSH   D
1364         XCHG
1365         LXI    H, STACK
1366         PUSH   H
1367         XCHG
1368         CALL   TRANSX
1369         POP    H
1370         POP    D
1371         PUSH   D
1372         CALL   PRENOS
1373         POP    H
1374         CALL   NEXL
1375         XCHG
1376         JMP    MOVE1
1377 AMT       POP    H
1378         CALL   ASSMAZ
1379         JMP    EDITAS
1380 ASSMAZ    7EFEFDC8E5;
1381         CALL   OVERA
1382         POP    D
1383 MAZ       MVI    B, 0FDH
1384         CALL   TRANS
1385         XCHG
1386         SHLD   PROM6
1387         RET
1388 PRENOS    PUSH   H
1389         CALL   OVERA
1390         LHLD   PROM6
1391         PUSH   H
1392         DAD    B

```

1393		SHLD	PROM6
1394		MOV	B,H
1395		MOV	C,L
1396		POP	H
1397		DCX	D
1398		CALL	MVMT
1399		INX	D
1400		POP	H
1401	TRANSX	MVI	B,0DDH
1402	TRANS		7E122313B8;
1403		JNZ	TRANS
1404		RET	
1405	MVMT	CALL	CODEHL
1406		RZ	
1407		MOV	A,M
1408		STAX	B
1409		DCX	B
1410		DCX	H
1411		JMP	MVMT
1412	TEXTP	SHLD	PROM5
1413	TXT1	MOV	A,M
1414		CPI	0FDH
1415		JZ	YES
1416		CALL	OVERA
1417		MOV	A,B
1418		ANA	A
1419		JNZ	NO
1420		MOV	A,C
1421		CPI	80
1422		JC	TXT1
1423	NO	LHLD	PROM5
1424	YES	MVI	M,0FDH
1425		SHLD	PROM6
1426		RET	
1427	OVERA	MVI	A,0DDH
1428	OVERAL	LXI	B,1
1429	OVERAK	CMP	M
1430		INX	H
1431		RZ	
1432		INX	B
1433		JMP	OVERAK
1434	OVER	CMP	M
1435		RNZ	
1436	OVE	INX	H
1437		JMP	OVER
1438	TABWS	LXI	H,TAB3+2
1439	OLT		7E23B97EC8E60F3C;
1440	RWT	INX	H
1441		DCR	A
1442		JNZ	RWT
1443		INX	H
1444		INX	H
1445		JMP	OLT
1446	FIEND	XRA	A
1447		STA	0C03EH
1448		STA	TB1
1449	NEXLT	LHLD	PROM0
1450		LDA	PROM4



```

1451      MOV  C,A
1452      MVI  B,0
1453      DAD  B
1454      SHLD PROM0
1455      LHLD LOC3
1456 NEXL  MVI  A,0DDH
1457 CROS  CMP  M
1458      INX  H
1459      JNZ  CROS
1460      RET
1461 ASLIDE LHLD PROM5
1462      LDA  PROM4
1463      CMA
1464      MOV  C,A
1465      MVI  B,255
1466 NWL   7EFED37C8EB09EB13D0;
1467      CALL NEXL
1468      JMP  NWL
1469 EDIN  CALL EDITE
1470 NUMIN MVI  A,32
1471      CALL OVER
1472      PUSH H
1473      DCX  H
1474 NUMLPL INX  H
1475      MOV  A,M
1476      CALL 80E0H
1477      JNC  NUMLPL
1478      MOV  A,M
1479      CPI  'H'
1480      DCX  H
1481      MOV  A,M
1482      POP  H
1483 NUMBRB LXI  D,0
1484      XCHG
1485      JZ   NUMHEX
1486      CPI  'B'
1487      JZ   NUMB
1488 NUMOPT 1A13D630;
1489      JC   ET
1490      CPI  10
1491      JNC  ET
1492      294D442929094F060009;
1493      JMP  NUMOPT
1494 ASLINE MVI  A,32
1495      CALL OVER
1496      ORI  255
1497      JMP  NUMBRB
1498 NUMB   1A13D630;
1499      JC   ETD
1500      CPI  2
1501      JNC  ETD
1502      29856F;
1503      JMP  NUMB
1504 NUMHEX 1A13CDE080;
1505      JC   ETD
1506      29292929B56F;
1507      JMP  NUMHEX
1508 ETD    INX  D

```

```

1509 ET      1BEB C9;
1510 ASSNEM ;
1511         PUSH D
1512         LXI D,STACK
1513         MVI B,32
1514         CALL TRANS
1515         PUSH D
1516         CALL OVER
1517         MOV A,M
1518         CPI 13
1519         JZ LIERR
1520         CALL WIN
1521         16082272C0;
1522         JC AND
1523         MVI A,32
1524         CALL OVER
1525         SHLD 0C072H
1526         CALL TABWS
1527         2B2B562323;
1528 OTHOP    PUSH PSW
1529         ANI 0FH
1530         JZ AENOP
1531         0620FE02;
1532         JZ CAR
1533         F1F5FE20;
1534         JC UNC
1535 CAR      MVI B,', '
1536 UNC     F13DF5237EE5D5F50F0F0F0FE60FC54704;
1537         CALL BTBLE
1538         C10EFFEB2A72C0;
1539         CALL CTL
1540         JC LIERR
1541         F1E60F3C47790F;
1542 ASSROL   RLC
1543         DCR B
1544         JNZ ASSROL
1545         D1B2572272C0E1F1;
1546         JMP OTHOP
1547 AENOP    POP PSW
1548 AND      POP H
1549         7223EB2A72C0CD37893EDD12D17BB2E1;
1550         JNZ VLOZ
1551         PUSH H
1552         CALL ASSMAZ
1553         POP H
1554 VLOZ     PUSH H
1555         LXI H,STACK
1556         PUSH H
1557         CALL OVERA
1558         LHLD PROM6
1559         DAD B
1560         CALL TLES3
1561 MEOVS    LXI H,80C3H
1562         JC LIAZ
1563         POP H
1564         POP D
1565         CALL PRENOS
1566         JMP EDITAN

```

```

1567 WIN      LXI   D,TAB1
1568 C10      LXI   B,2000H
1569          JMP   CTL
1570 EFD      CMP   M
1571          JNZ   BLE
1572          INX   H
1573          INX   D
1574 MA       1A07A71F;
1575          JNC   EFD
1576          CMP   M
1577          JZ    EDED
1578 BLE      POP   H
1579 NX       1A13A7;
1580          JP    NX
1581          1AFE2337C8;
1582 CTL      PUSH  H
1583          INR   C
1584          JMP   MA
1585 EDED      237EB8;
1586          JNZ   BLE
1587          D123C9;
1588 CHOUT2   E52A3EC023223EC02B267F77E1C9;
1589 NUOUT2   E52A3EC023223EC02B2B267FCD3B81E1C9;
1590 LIIER3   CPI   'L'
1591          JNZ   LIIERR
1592          INX   H
1593          PUSH  H
1594          MVI   A,32
1595          CALL  OVERAL
1596          POP   D
1597          MVI   H,0
1598          LDA   PROM4
1599          MOV   L,A
1600          SHLD  PROM0
1601          LHLD  PROM5
1602 FIN1     SHLD  LOC3
1603          MOV   A,M
1604          CPI   0FDH
1605          JZ    EDITAS
1606          CALL  CPIR
1607          JZ    FINE
1608          PUSH  B
1609          CALL  NEXLT
1610          POP   B
1611          JMP   FIN1
1612 LISTAS   CALL  READY
1613 FINE     LXI   H,FIANUP
1614          SHLD  DEBOX2
1615 LISTAD   CALL  LLIST
1616          JZ    ASSEMB
1617          JMP   ASSED3
1618 READY    PUSH  D
1619          CALL  ASLIDE
1620          JC    EDITAS
1621          E37B2F5F7A2F571319;
1622          SHLD  PROM0
1623          POP   H
1624          SHLD  LOC3

```

```

1625          RET
1626 FIANUP    LHLD LOC3
1627          MOV  A,M
1628          CPI  0FDH
1629          JZ   EDITAX
1630          PUSH H
1631          LHLD PROM0
1632          CALL LOUT
1633          E13E05323EC01620;
1634 FIA2       7E23BAF5;
1635          CNZ   CHOUT
1636          POP   PSW
1637          JNZ   FIA2
1638          3E10323EC07AFEDD;
1639          JZ    FIEND
1640          4EE53EC9;
1641          STA   NUOUT
1642          CALL  DIAID
1643          MVI   A,0C3H
1644          STA   NUOUT
1645          E12316DD;
1646          JMP   FIA2
1647 LOUT1      0E10110000;
1648 LOREPT     297B8F275F7A8F27570D;
1649          JNZ   LOREPT
1650          RET
1651 LOUT       CALL LOUT1
1652          XCHG
1653 HLOUT      XCHG
1654 DEOUT      MOV  A,D
1655          CALL  NUOUT
1656          MOV  A,E
1657          JMP   NUOUT
1658 ASLINO     CALL  READY
1659          LXI   H,CHOUT2
1660          SHLD  CHOUT+1
1661          LXI   H,NUOUT2
1662          SHLD  NUOUT+1
1663          CALL  FIANUP
1664          LXI   H,S
1665          SHLD  CHOUT+1
1666          LXI   H,S1
1667          SHLD  NUOUT+1
1668          CALL  8855H
1669          JMP   ASSED3
1670 INIMG      MVI   A,0
1671          STA   TBX
1672          3E7ED35F3EABD35D3E06D35D3E40D31FC3B98D;
1673 ENTSEC     CALL  INIMG
1674          LXI   H,ESNQ
1675 ENTS1      CALL  TXIN
1676          MOV  A,L
1677          CPI   64
1678          RC
1679          CALL  88A3H
1680          LXI   H,ITOH
1681          JMP   ENTS1
1682 SAVE      LXI   D,0C1B8H

```

```

1683      MVI  B,8
1684 SAVL1  MOV  A,M
1685      CPI  13
1686      JZ   TOEN
1687      12231305;
1688      JNZ  SAVL1
1689      RET
1690 TOEN   MVI  A,32
1691 SAVL3  121305;
1692      JNZ  SAVL3
1693      RET
1694 SAVEAS LXI  H,ENAM
1695      CALL ZOBAS
1696      CALL 8BEEH
1697      LHLD 0C078H
1698      CALL SAVE
1699      CALL ENTSEC
1700      STA  0C1B2H
1701      MVI  A,'% '
1702      STA  0C1B3H
1703      LHLD PROM5
1704      SHLD 0C1B4H
1705      XCHG
1706      LHLD PROM6
1707      MOV  A,E
1708      CMA
1709      MOV  E,A
1710      MOV  A,D
1711      CMA
1712      MOV  D,A
1713      DAD  D
1714      INX  H
1715      SHLD 0C1B6H
1716      CALL TAPES
1717      CALL HDD
1718      CALL 8F60H
1719      LXI  H,8F33H
1720      CALL ZOBEEP
1721      JMP  ASSEMB
1722 HDD     11B2C11ACD738E2A30C04F1F1F1F1F;
1723      E60FF630772379E60FF6307723362F;
1724      13231A7723362011B8C10608;
1725 HDD1   231A771305;
1726      JNZ  HDD1
1727      JMP  8855H
1728 LOADAS LHLD PROM5
1729      SHLD PROM6
1730      21;
1731 VERIFY XRA  A
1732      21;
1733 MERGE  MVI  A,'% '
1734      STA  0C1B1H
1735      CALL ENTSEC
1736      STA  0C1B0H
1737      LXI  H,8F22H
1738      SHLD 0C074H
1739      LXI  H,ME01
1740      PUSH H

```

1741		XRA	A
1742	LOAD1	CNZ	88A3H
1743		CALL	8DE2H
1744		CALL	HDD
1745		2AB0C13AB2C1BD;	
1746		JNZ	LOAD1
1747		3AB3C125244C;	
1748		JZ	HDX
1749		CMP	H
1750		JNZ	LOAD1
1751	HDX	LHLD	PROM6
1752		XCHG	
1753		LHLD	0C1B6H
1754		PUSH	H
1755		DAD	D
1756		LDA	0C1B1H
1757		ANA	A
1758		CNZ	TLES3
1759		JC	MEOV
1760		POP	H
1761		XCHG	
1762		CALL	8DC4H
1763		JZ	ME01
1764		LHLD	PROM6
1765		MVI	M,0FDH
1766	ME01	CALL	8EC1H
1767		JMP	ASSEMB
1768	MEOV	LXI	H,80C3H
1769		CALL	ZOBEEP
1770		JMP	ASSEMB
1771	ZFNB	MVI	B,32
1772	ZFNAV	XCHG	
1773		LHLD	LOC1
1774		XCHG	
1775		JMP	ZFIN
1776	ZBLE	MVI	A,32
1777		XCHG	
1778		CALL	CROS
1779		2323EBE1;	
1780	ZFIN	1AFEFF37C8E5;	
1781	ZMA	LDAX	D
1782		CPI	32
1783		JZ	ZCEN
1784		CMP	M
1785		JNZ	ZBLE
1786		INX	H
1787		INX	D
1788		JMP	ZMA
1789	ZCEN	MOV	A,M
1790		CMP	B
1791		JNZ	ZBLE
1792		E5EB235E2356E1C1C9;	
1793	OPIN	MOV	A,M
1794		CPI	27H
1795		JNZ	ZOTHER
1796	C11	LXI	D,0
1797	ZLOPL	237EFEDD;	
1798		JZ	ERROR!

1799		CPI	27H
1800		JZ	ZAPS
1801	ZCIN	MOV	D,E
1802		MOV	E,A
1803		JMP	ZLOPL
1804	ZAPS		237EFE27;
1805		JZ	ZCIN
1806		RET	
1807	ZOTHER	CPI	48
1808		JC	ERROR!
1809		CPI	3AH
1810		JC	NUMIN
1811		CALL	ZFNB
1812		RNC	
1813		MVI	B, '+'
1814		CALL	ZFNAV
1815		RNC	
1816		MVI	B, '-'
1817		CALL	ZFNAV
1818		RNC	
1819		MVI	B, '\'
1820		CALL	ZFNAV
1821		RNC	
1822	ERROR!	CALL	88A3H
1823	ERROR	CALL	WEC5
1824		CALL	8A73H
1825		LHLD	PROM0
1826		MVI	A, 83H
1827		STA	0C134H
1828		XCHG	
1829		JMP	ASLINO
1830	CISIN	LDA	PROM4+1
1831		ANA	A
1832		RZ	
1833	EQU	CALL	OPIN
1834	YNOXT	MVI	A, 32
1835		CALL	OVER
1836		MOV	A,M
1837		CPI	0DDH
1838		INX	H
1839		RZ	
1840		CPI	'+'
1841		JZ	YPLUS
1842		CPI	'-'
1843		JZ	YPLUS
1844		CPI	'\'
1845		JNZ	ERROR!
1846		MOV	A,E
1847		MOV	E,D
1848		MOV	D,A
1849		JMP	YNOXT
1850	YPLUS		D5F53E20;
1851		CALL	OVER
1852		CALL	OPIN
1853			F1FE2D;
1854		JNZ	YPL
1855			7B2F5F7A2F5713;
1856	YPL		E319EBE1;

1857		JMP	YNOXT
1858	ZAZNAM	LDA	PROM4+1
1859		ANA	A
1860		RNZ	
1861		PUSH	H
1862		MVI	B, 32
1863		CALL	ZFNAV
1864		JNC	ERROR!
1865		POP	H
1866		LDA	TRFLG
1867		ANI	4
1868		JNZ	TR5
1869		PUSH	H
1870	TR6	MOV	A, M
1871		INX	H
1872		CPI	32
1873		JZ	TR7
1874		CALL	S
1875		JMP	TR6
1876	TR7	LHLD	PROM1
1877			3E0A323EC0;
1878		CALL	HLOUT
1879		CALL	ODRD
1880		POP	H
1881	TR5	XCHG	
1882		LHLD	LOC2
1883			EBE5D53E20;
1884		CALL	OVERAL
1885		XCHG	
1886		DAD	B
1887		CALL	TLES2
1888		JC	MOVAA
1889			D1E10620;
1890		CALL	TRANS
1891		LHLD	PROM1
1892			EB7323722336FF;
1893		SHLD	LOC2
1894		RET	
1895	UNES	MOV	A, H
1896		CMA	
1897		ANI	0C0H
1898		RNZ	
1899		LDA	TRFLG
1900		ORI	4
1901		STA	TRFLG
1902		RET	
1903	MOVAA	LXI	H, 80C3H
1904		JMP	BADORG+3
1905	ZAPIS	LHLD	LENX
1906		INX	H
1907		SHLD	LENX
1908		LHLD	PROM1
1909		INX	H
1910		SHLD	PROM1
1911		LDA	PROM4+1
1912		ANA	A
1913		LDA	TRFLG
1914		RZ	



1915		ANI	24
1916		CPI	8
1917		RZ	
1918		JNC	ZPSM
1919		CALL	TLES1
1920		JC	BADORG
1921			2B70C9;
1922	END2	INX	H
1923		MOV	A,M
1924		CPI	' - '
1925		RZ	
1926	END3	LDA	SUMA
1927		MOV	B,A
1928	ZPSM	CALL	8DB1H
1929		MOV	A,B
1930		OUT	1EH
1931		LDA	SUMA
1932		ADD	B
1933		STA	SUMA
1934		RET	
1935	BADORG	LXI	H,BORG
1936		CALL	ZOBEEP
1937		JMP	ERROR
1938	TXIN	CALL	ZOBAS
1939		CALL	EDIN
1940		XCHG	
1941		RET	
1942	TRANAS	LXI	H,OPTS
1943		MVI	A,5
1944		STA	0C137H
1945		CALL	ZOBAS
1946		CALL	80E0H
1947		JC	TRANAS
1948		MOV	B,A
1949		ANI	8
1950		RRC	
1951		RRC	
1952		CMA	
1953		ANA	B
1954		STA	TRFLG
1955		ANI	1
1956		LHLD	PROM6
1957		INX	H
1958		JZ	TR1
1959	TR2	LXI	H,TADD
1960		CALL	TXIN
1961		CALL	TLES2
1962		JC	TR2
1963	TR1	SHLD	LOC1
1964		SHLD	LOC2
1965		MVI	M,255
1966		CALL	UNES
1967		LDA	PROM4
1968		MOV	L,A
1969		MVI	H,0
1970		SHLD	PROM4
1971		SHLD	PROM0
1972		CALL	PROC

1973		LDA	TRFLG
1974		CPI	16
1975		JNC	ERSOU
1976		ANI	11
1977		STA	TRFLG
1978		ANI	2
1979		CZ	INIMG
1980		JZ	TR3
1981	TR4	LXI	H, SADD
1982		CALL	TXIN
1983		CALL	TLES1
1984		JC	TR4
1985		SHLD	PROM1
1986		SHLD	ZACP
1987		CALL	UNES
1988	TR3	ORI	5
1989		LHLD	PROM4
1990		STA	PROM4+1
1991		SHLD	PROM0
1992		LHLD	LOC1
1993		CALL	UNES
1994		CALL	PROC
1995		CALL	88A3H
1996		LDA	TRFLG
1997		ANI	2
1998		JZ	ASSEMB
1999	TRNS	LXI	H, TRND
2000		CALL	ZOBUNB
2001		CPI	'N'
2002		JZ	ASSEMB
2003		CPI	'Y'
2004		JNZ	TRNS
2005	TRRE	LXI	H, RLAB
2006		CALL	ZOBUNB
2007		JZ	ASSEMB
2008		CALL	EDITE
2009		LXI	D, LOC DM
2010		MVI	B, 32
2011		CALL	ZFIN
2012		JNC	TRR1
2013		CALL	ZFNB
2014		JC	TRRE
2015	TRR1	PUSH	D
2016		LXI	H, MOVES
2017		LXI	D, LOC AND
2018	C12	LXI	B, 11
2019		CALL	LDIR
2020		LHLD	LENX
2021		MOV	B, H
2022		MOV	C, L
2023		MOV	A, L
2024		ORA	B
2025		JZ	ASSEMB
2026		LHLD	ZACP
2027		XCHG	
2028		LHLD	STACK
2029		JMP	LOC AND
2030	LDIRX	XCHG	

2031	LDIR	MOV	A,B
2032		ORA	C
2033		RZ	
2034		MOV	A,M
2035		STAX	D
2036		INX	H
2037		INX	D
2038		DCX	B
2039		JMP	LDIR
2040	MOVES	LDAX	D
2041		MOV	M,A
2042		INX	H
2043		INX	D
2044		DCX	B
2045		MOV	A,B
2046		ORA	C
2047		JNZ	LOCAND
2048		RET	
2049	ERSOU	LXI	H,ERSO
2050		CALL	ZOBEEP
2051		JMP	ASSEMB
2052	PROC	LXI	H,STACK
2053		SHLD	UKAZ
2054	C13	LXI	H,0
2055		SHLD	LENX
2056		SHLD	FGFT
2057		LHLD	PROM5
2058	PRO	SHLD	LOC3
2059		MOV	A,M
2060		CPI	0FDH
2061		RZ	
2062		CALL	P
2063		CALL	8C74H
2064		JNZ	TR8
2065	UXO	LXI	H,PETC
2066		CALL	ZOBAS
2067		CPI	89H
2068		JNZ	UXO
2069	TR8	CALL	NEXLT
2070		JMP	PRO
2071	IFX	MVI	A,32
2072		CALL	CROS
2073		MOV	A,M
2074		CPI	8
2075		RNZ	
2076		INX	H
2077		MOV	A,M
2078		CPI	'J'
2079		RNZ	
2080		XRA	A
2081		STA	FGFT
2082		RET	
2083	P	LDA	FGFT
2084		ANA	A
2085		MOV	A,M
2086		JNZ	IFX
2087		SUI	48
2088		JC	NOTH

2089		CPI	10
2090		JNC	ZA
2091		LDA	PROM4+1
2092		ANA	A
2093		JZ	TR9
2094		LDA	TRFLG
2095		ANI	2
2096		JNZ	NOTHA
2097	TR9	CALL	NUMIN
2098		XCHG	
2099		SHLD	PROM1
2100		XCHG	
2101		JMP	NOTH
2102	NOTHA	PUSH	H
2103		CALL	NUMIN
2104		LHLD	UKAZ
2105		MOV	M, E
2106		INX	H
2107		MOV	M, D
2108		INX	H
2109		SHLD	UKAZ
2110		LDA	TRFLG
2111		ANI	4
2112		JNZ	NOHA
2113		CALL	DEOUT
2114			3E06323EC0;
2115		LHLD	PROM1
2116		CALL	HLOUT
2117		CALL	ODRD
2118	NOHA	POP	H
2119		JMP	NOTH
2120	ZA	PUSH	H
2121		PUSH	H
2122		LHLD	PROM1
2123		XTHL	
2124		PUSH	H
2125		MVI	A, 32
2126		CALL	CROS
2127		MOV	A, M
2128		CPI	8
2129		JNZ	OKAY
2130			237EFE'=';
2131		JNZ	OKAY
2132		INX	H
2133		CALL	EQU
2134		XCHG	
2135		SHLD	PROM1
2136	OKAY	POP	H
2137		CALL	ZAZNAM
2138		POP	H
2139		SHLD	PROM1
2140		POP	H
2141	NOTH	MVI	A, 32
2142		CALL	CROS
2143			46EB78FE08;
2144		JNZ	BOOM
2145			EB237EFE2A;
2146		JZ	BOM1

2147		CPI	'#'
2148		JNZ	BOM2
2149		INX	H
2150		CALL	NUMIN
2151		LHLD	PROM1
2152		DAD	D
2153		LDA	TRFLG
2154		ANI	16
2155		JNZ	TT2
2156		SHLD	PROM1
2157		LHLD	LENX
2158		DAD	D
2159		SHLD	LENX
2160		RET	
2161	TT2	MVI	B,0
2162	T1	MOV	A,D
2163		ORA	E
2164		RZ	
2165		CALL	ZAPIS
2166		DCX	D
2167		JMP	T1
2168	BOM4	CPI	'W'
2169		JNZ	BOM3
2170		LDA	PROM4+1
2171		ANA	A
2172		RZ	
2173		INX	H
2174		CALL	NUMIN
2175		JMP	88B5H
2176	BOM3	CPI	'='
2177		RZ	
2178	ZDATA	XCHG	
2179		CALL	ASSCOM
2180		JC	ERROR
2181		LHLD	0C078H
2182		MOV	D,B
2183		MOV	E,C
2184	ZDATA1	MOV	A,D
2185		ORA	E
2186		RZ	
2187		DCX	D
2188		MOV	B,M
2189		INX	H
2190		PUSH	H
2191		CALL	ZAPIS
2192		POP	H
2193		JMP	ZDATA1
2194	BOM2	CPI	'['
2195		JNZ	BOM2A
2196		STA	FGFT
2197		RET	
2198	BOM2A	CPI	', '
2199		JNZ	BOM2B
2200	BOMXX	INX	H
2201		CALL	NUMIN
2202		MOV	B,E
2203		PUSH	H
2204		CALL	ZAPIS

2205		POP	H
2206		MOV	A,M
2207		CPI	','
2208		JZ	BOMXX
2209		CPI	','
2210		RZ	
2211		CALL	ERINDA
2212		JMP	ERROR
2213	BOM2B	CPI	'@'
2214		JNZ	BOM4
2215		LDA	TRFLG
2216		MOV	B,A
2217		ANI	8
2218		RZ	
2219		MOV	A,B
2220		XRI	16
2221		STA	TRFLG
2222		LDA	PROM4+1
2223		ANA	A
2224		MOV	A,B
2225		JNZ	PR2
2226		ANI	16
2227		JNZ	END
2228	C14	LXI	H,0
2229		SHLD	LENX
2230		RET	
2231	END	LHLD	LENX
2232		MOV	A,H
2233		ORA	L
2234		JZ	NOFILE
2235		XCHG	
2236		LHLD	UKAZ
2237		MOV	M,D
2238		INX	H
2239		MOV	M,E
2240		INX	H
2241		SHLD	UKAZ
2242		RET	
2243	NOFILE	LXI	H,NOFI
2244		JMP	BADORG+3
2245	OUTH2	MVI	A,255
2246	OUTH1	MVI	C,16
2247		MOV	B,A
2248	OTH1	CALL	8DB1H
2249		MOV	A,B
2250		OUT	1EH
2251		DCR	C
2252		JNZ	OTH1
2253		RET	
2254	PR2	ANI	16
2255		JNZ	END2
2256		XRA	A
2257		STA	SUMA
2258		INX	H
2259		MOV	A,M
2260		CPI	'N'
2261		PUSH	PSW
2262		INX	H

2263		CALL	NUMIN
2264		MOV	A, E
2265		CPI	64
2266		JNC	ERROR!
2267		STA	0C1B2H
2268		MVI	A, 32
2269		CALL	OVER
2270		PUSH	H
2271		CALL	NEXL
2272		DCX	H
2273		MVI	M, 13
2274		XTHL	
2275		CALL	SAVE
2276		POP	H
2277		MVI	M, 0DDH
2278		MVI	A, '?'
2279		STA	0C1B3H
2280		LHLD	UKAZ
2281		MOV	D, M
2282		INX	H
2283		MOV	E, M
2284		INX	H
2285		SHLD	UKAZ
2286		XCHG	
2287		DCX	H
2288		SHLD	0C1B6H
2289		LHLD	PROM1
2290		SHLD	0C1B4H
2291		POP	PSW
2292		PUSH	PSW
2293		CNZ	TAPES
2294		POP	PSW
2295		RZ	
2296		CPI	'H'
2297		RZ	
2298		CALL	HDD
2299		CALL	OUTH2
2300		XRA	A
2301		CALL	OUTH1
2302		MVI	A, 55H
2303		CALL	OUTH1
2304		LXI	H, 0C1B2H
2305	C15	LXI	D, 13
2306		CALL	8D6CH
2307	C16	LXI	D, 0
2308		JMP	88B5H
2309	BOM1	INX	H
2310		CALL	CISIN
2311		JMP	BRUM
2312	BOOM	CALL	ZAPIS
2313			EB4E23E5;
2314		CALL	ASKIN
2315			2323237EE630FE10E1C8F5;
2316		CALL	CISIN
2317			F1FE3043;
2318		JNZ	ZAPIS
2319	BRUM	MOV	B, E
2320		CALL	ZAPIS

2321		MOV	B,D
2322		JMP	ZAPIS
2323	S1	MOV	B,A
2324		RRC	
2325		RRC	
2326		RRC	
2327		RRC	
2328		ANI	15
2329		CALL	814FH
2330		CALL	S
2331		MOV	A,B
2332		ANI	15
2333		CALL	814FH
2334	S	PUSH	D
2335		PUSH	H
2336		PUSH	B
2337		MOV	B,A
2338		LDA	FTISK
2339		ANA	A
2340		MOV	A,B
2341		PUSH	B
2342		CNZ	QQA
2343		POP	B
2344		LDA	FPRIN
2345		ANA	A
2346		MOV	A,B
2347		JZ	8503H
2348		LHLD	0C03EH
2349		CPI	13
2350		JNZ	SW1
2351		LXI	H,0FB00H-1
2352	SW1	MOV	A,L
2353		ANI	3FH
2354		CPI	30H
2355		JZ	8526H
2356		JMP	8522H
2357	QQA	PUSH	PSW
2358	QQA1	LDA	TB1
2359		MOV	B,A
2360		LDA	0C03EH
2361		CPI	30H
2362		JZ	QQA2
2363		CMP	B
2364		JZ	QQA2
2365		SBB	A
2366		ANI	237
2367		ADI	32
2368	QQA3	CALL	OUTX
2369		JMP	QQA1
2370	QQA2	POP	PSW
2371	OUTX	PUSH	PSW
2372		CPI	13
2373		LDA	TB1
2374		JNZ	QQA4
2375		LDA	FPAGE
2376		INR	A
2377		JZ	QQA7
2378		DCR	A



2379	JZ	QQA9
2380	DCR	A
2381	STA	FPAGE
2382	JNZ	QQA7
2383	QQA9 LDA	PAGE
2384	STA	FPAGE
2385	MVI	B,4
2386	QQA8 PUSH	B
2387	MVI	A,13
2388	CALL	QQA6
2389	POP	B
2390	DCR	B
2391	JNZ	QQA8
2392	QQA7	;
2393	QQA5 MVI	A,255
2394	QQA4 INR	A
2395	STA	TB1
2396	POP	PSW
2397	QQA6 JMP	WEC1
2398	WROT1 LDA	FTAPE
2399	ANA	A
2400	RNZ	
2401	LHLD	WECTOR
2402	MOV	A,L
2403	ANA	A
2404	RZ	
2405	XRA	A
2406	CALL	WEC1
2407	JMP	WROT1
2408	WECCO LHLD	WECTOR
2409	MOV	M,A
2410	INR	L
2411	SHLD	WECTOR
2412	RNZ	
2413	JMP	WEC3
2414	WECOUT LHLD	0C074H
2415	PUSH	H
2416	LDA	0C134H
2417	PUSH	PSW
2418	CALL	TAPES
2419	POP	PSW
2420	STA	0C134H
2421	CALL	OUTH2
2422	MVI	A,77H
2423	CALL	OUTH1
2424	LHLD	WECTOR
2425	XRA	A
2426	STA	SUMA
2427	WECT1 MOV	B,M
2428	CALL	ZPSM
2429	INR	L
2430	JNZ	WECT1
2431	CALL	END3
2432	POP	H
2433	JMP	ZOBRAZ
2434	@	
2435	1800H	;
2436	PRBUFF	#256

```

2437 KONECA ;
2438 STACK =7600H+128
2439 0C230H ;
2440 PROM0 #2
2441 PROM1 #2
2442 PROM2 #2
2443 PROM3 #2
2444 PROM4 #2
2445 PROM5 #2
2446 PROM6 #2
2447 FLAGS #2
2448 #48
2449 WINDOW #2
2450 REGLOC #12
2451 WARI #2
2452 #48
2453 TB21 #1
2454 WECTOR #2
2455 TBX #1
2456 PFX1 #1
2457 TB1 #1
2458 LOC1 #2
2459 LOC2 #2
2460 LOC3 #2
2461 TRFLG #2
2462 FGFT #1
2463 #1
2464 #48
2465 PFXE ;
2466 FLINE #1
2467 FTISK #1
2468 FPAGE #1
2469 FTAPE #1
2470 FPRIN #1
2471 FSTAP #1
2472 #6
2473 LNOP #2
2474 LENX #2
2475 #48
2476 UKAZ #2
2477 ZACP #2
2478 SUMA #1
2479 LOCAND #11
2480 CEND ;

```

XQZAC	7FED
XQX2	7000
XERR	7003
XTABLX	7012
XSTRXX	707C
XSTAR	708D
XSTRRR	708E
XCLS1A	70CC
XCLS2A	70CE
XPOCET	0064
XLP0	70EE
XLP7	70FE
XLP1	7109
XLP2	710E
XUPR	712F
XMINUS	7147
XEND	714A
XBODHL	7155
XBYTES	7162
XENB	717C
XWAIN	7180
XWAN	718A
XWWIN	718F
XJUMP	7192
XCODES	7195
XINST	71B3
XLOGGER	71BC
XAD2	71C8
XADD	71CE
XADD1	71DC
XCOMP	71E4
XIN	71F7
XLEGT	7204
XB3	7247
XB2	7248
XB1	7249
XTABUK	724C
XLENI	724E
XADDR	7250
XSUMA	7252
XLEG	7253
XLEE	7254
XP	7255
XLOOP1	7257
XLOOP2	7267
XTABLE	7285
XZAPIS	74C7
XST1	74E7
XG0	74EA
XRUN	74F9
XBASE	74E9
XTRTAD	74E8
XQX1	7501
XTAB	7255

ZACATA	0000
WEC1	0003
WEC5	0006
WEC3	0009
PAGE	000C
LOCDM	000D
C1	0011
DAMWRD	0019
TAB1	006A
TAB2	0150
TAB3	0172
LOCFLG	02C2
C2	02C8
ASSTAB	02C8
MONTAB	02E4
BASTAB	02F8
DEAD	0306
MOAD	030F
VERA	0317
DIAD	031E
ASAD	0321
FIAD	032B
LIAD	0330
EDAD	0335
TRAD	033A
LOAD	0344
MEGD	0349
SAAD	034F
STAP	0354
EAQ	0372
PSQ	038F
PETC	03AC
ECOM	03D1
PSTO	03DF
WARN	03EC
ELOQ	0402
LIER	041C
ESNQ	0433
ERSO	0457
NOFI	046D
ENAM	047B
ITOH	0486
BORG	049A
OPTS	04A8
TADD	04B2
SADD	04C2
TRND	04D2
RLAB	04DD
WDAD	04E9
REGERS	04EB
START	0501
CLS	0505
C3	0508
CLSLP1	050B
CLSL0P	050D
DEBOPS	0519
DEBOP2	0521
VYPSA1	0530

VYPS1	0533
VYPS4	053D
C4	0544
VYPS2	055F
DEBREG	0567
C5	0574
DEBZNO	05B5
DEBZN2	05F0
DEBC3C	0620
DEBLOP	063F
C6	0663
LIMEX1	0669
LIMEX	066C
DOSTEP	067B
LOFG	06B2
DEBUAL	06C3
DEBCAL	06D2
DEBJMP	06E9
DEBJDE	06ED
DEBJHL	06EE
DEBCHL	06F2
DEBRET	06F8
DEBWAR	0705
DEBRST	070B
DEBSET	0721
C7	072D
DEBSTA	073E
BOXER	0749
DEBOX1	075C
DEBOX2	075D
C8	076B
DEBUGR	077C
DEBR1	078A
WWARNI	079D
DEBCAA	07A6
DEBOTH	07C0
DEBOT2	07CB
DEBBBB	07D5
DEBCCC	07DD
DEBURE	0835
DEBDDD	084A
DE1	084E
ISPOL	088A
ISPOL1	0897
OUTDAM	08C5
QWERTZ	08C9
OUTDA1	08D7
ROZBE1	08FE
ROZBEH	0904
ROVINA	0907
ROZB1	0910
ROZB2	0916
ROZB4	0936
ROZB3	093C
TAPES	0942
ZOBAS	094D
ASSEMB	0951
ODRD	0960

ZOBEEP	0965	DISUNK	0BDE
ZOBUNB	096A	DISC02	0BE3
CANZ	0990	DISPPP	0C14
LLIST	099D	DISPP	0C1E
LLIST1	09A3	DISGON	0C4D
CONQ	09CB	DISNOI	0C59
ZOBRAZ	09D1	DISTW0	0C6F
ZOBLAB	09E1	NUOUT	0C79
LISTM0	09F6	EDITAN	0C7C
MONITR	09FC	EDITE	0C96
CPIR	0A02	EDITAX	0CAA
CPIR1	0A05	EDITAS	0CB0
CPIR2	0A12	ASSED2	0CB3
FIND	0A16	ASSED3	0CB9
FINQ	0A1C	LIIER1	0CDA
FINF	0A36	LIERN	0CE2
FINX	0A3F	LIIER2	0CE8
LISTER	0A4F	TLES1	0D06
LISING	0A52	TLES2	0D18
ACONQ	0A58	TLES3	0D2A
LIMEDE	0A69	UNIT	0D46
MEMLI2	0A70	CODEHL	0D4E
MEMLI3	0A8A	ASSOTH	0D56
KYU	0A9B	LIERR	0D83
KYV	0AA5	LIAZ	0D86
EDISX	0AAA	DEL	0D8E
EDITM1	0AB3	MOVE	0DB3
EDITM0	0AB9	MOVE1	0DD6
EDIWCQ	0ABF	AMT	0DF8
EDITEX	0AC2	ASSMAZ	0DFF
CURSOU	0ADE	MAZ	0E08
ERINDA	0AF9	PRENOS	0E12
ERAD1	0AFC	TRANSX	0E27
ERAD	0B01	TRANS	0E29
COMPRI	0B07	MVMT	0E32
ASSCOM	0B12	TEXTP	0E3D
C9	0B16	TXT1	0E40
COMPRI	0B19	NO	0E54
MEMOUT	0B3E	YES	0E57
C17	0B46	OVERA	0E5D
STRING	0B4F	OVERAL	0E5F
EDIST2	0B65	OVERAK	0E62
DISAMO	0B6B	OVER	0E69
BWORD	0B74	OVE	0E6B
BPRVE	0B77	TABWS	0E6F
BPRVE1	0B79	OLT	0E72
BTBLE	0B82	RWT	0E7A
BTBLE1	0B85	FIEND	0E84
BTBLE2	0B87	NEXLT	0E8B
ASKIN	0B91	NEXL	0E9B
ASKIN1	0B94	CROS	0E9D
ASKIN2	0BA4	ASLIDE	0EA3
PRINH1	0BAC	NWL	0EAD
PRINHL	0BB0	EDIN	0EBD
CHOUT	0BB7	NUMIN	0EC0
DISASM	0BBA	NUMLPL	0EC7
DEBDIS	0BBA	NUMBRB	0ED5
DIAID	0BC8	NUMOPT	0EE1

ASLINE	0EFA	ZFIN	1256
NUMB	0F04	ZMA	125C
NUMHEX	0F16	ZCEN	126B
ETD	0F27	OPIN	1279
ET	0F28	C11	127F
ASSNEM	0F2B	ZLOPL	1282
OTHOP	0F59	ZCIN	128E
CAR	0F6D	ZAPS	1293
UNC	0F6F	ZOTHER	129B
ASSROL	0F97	ERROR!	12BB
AENOP	0FA7	ERROR	12BE
AND	0FA8	CISIN	12D0
VLOZ	0FC1	EQU	12D5
MEOVS	0FD0	YNOXT	12D8
WIN	0FDE	YPLUS	12F7
C10	0FE1	YPL	130E
EFD	0FE7	ZAZNAM	1315
MA	0FED	TR6	132D
BLE	0FF8	TR7	133A
NX	0FF9	TR5	1349
CTL	1004	UNES	1372
EDED	1009	MOVAA	1380
CHOUT2	1012	ZAPIS	1386
NUOUT2	1020	END2	13AD
LIIER3	1032	END3	13B2
FIN1	104B	ZPSM	13B6
LISTAS	1062	BADORG	13C4
FINE	1065	TXIN	13CD
LISTAD	106B	TRANAS	13D5
READY	1074	TR2	13F9
FIANUP	108C	TR1	1405
FIA2	10A4	TR4	1434
LOUT1	10D0	TR3	1449
LOREPT	10D5	TRNS	1468
LOUT	10E3	TRRE	1478
HLOUT	10E7	TRR1	1495
DEOUT	10E8	C12	149C
ASLINO	10F0	LDIRX	14B6
INIMG	1114	LDIR	14B7
ENTSEC	112C	MOVES	14C2
ENTS1	1132	ERSOU	14CD
SAVE	1142	PROC	14D6
SAVL1	1147	C13	14DC
TOEN	1155	PRO	14E8
SAVL3	1157	UXO	14F8
SAVEAS	115E	TR8	1503
HDD	119F	IFX	1509
HDD1	11C9	P	151C
LOADAS	11D4	TR9	153D
VERIFY	11DB	NOTHA	1548
MERGE	11DD	NOHA	156F
LOAD1	11F3	ZA	1573
HDX	1213	OKAY	1594
ME01	1233	NOTH	159D
MEOV	1239	TT2	15D2
ZFNB	1242	T1	15D4
ZFNAV	1244	BOM4	15DE
ZBLE	124C	BOM3	15EF

ZDATA	15F2	TRFLG	C2BC
ZDATA1	15FE	FGFT	C2BE
BOM2	160C	PFXE	C2F0
BOM2A	1615	FLINE	C2F0
BOMXX	161A	FTISK	C2F1
BOM2B	1633	FPAGE	C2F2
C14	1652	FTAPE	C2F3
END	1659	FPRIN	C2F4
NOFILE	166D	FSTAP	C2F5
OUTH2	1673	LNOP	C2FC
OUTH1	1675	LENX	C2FE
OTH1	1678	UKAZ	C330
PR2	1683	ZACP	C332
C15	16E7	SUMA	C334
C16	16ED	LOCAND	C335
BOM1	16F3	CEND	C340
BOOM	16FA		
BRUM	1719		
S1	1721		
S	1734		
SW1	1755		
QQA	1760		
QQA1	1761		
QQA3	1776		
QQA2	177C		
OUTX	177D		
QQA9	1798		
QQA8	17A0		
QQA7	17AB		
QQA5	17AB		
QQA4	17AD		
QQA6	17B2		
WROT1	17B5		
WECC0	17C7		
WECOUT	17D3		
WECT1	17F1		
PRBUFF	1800		
KONECA	1900		
STACK	7680		
PROM0	C230		
PROM1	C232		
PROM2	C234		
PROM3	C236		
PROM4	C238		
PROM5	C23A		
PROM6	C23C		
FLAGS	C23E		
WINDOW	C270		
REGLOC	C272		
WARI	C27E		
TB21	C2B0		
WECTOR	C2B1		
TBX	C2B3		
PFX1	C2B4		
TB1	C2B5		
LOC1	C2B6		
LOC2	C2B8		
LOC3	C2BA		

## OBSAH

MANUÁL DAM-0000 .....	5
ASSEMBLER .....	7
EDIT .....	7
TRANSLATE .....	13
SAVE .....	15
VERIFY .....	15
LOAD .....	15
MERGE .....	16
MONITOR .....	16
EDIT .....	16
FIND .....	17
LIST .....	17
DISASSEMBLER .....	17
DEBUGGER .....	18
SPECIÁLNÍ FUNKCE .....	20
TISKOVÁ FUNKCE .....	21
MOŽNOSTI UŽIVATELSKÝCH ÚPRAV .....	21
VÝPIS ZDROJOVÉHO TEXTU .....	23
VÝPIS NÁVĚŠTÍ .....	67
OBSAH .....	72



Assembler-monitor pro PMD 85. DAM 0000.

Autor Karel Šuhajda. Edice elektroniky Svazarmu. Řada 5 - výpočetní technika. Odpovědný redaktor PhDr. Jindřich Jirka. Návrh obálky a grafická úprava Josef Svoboda. Technická redakce Daniela Prokopová. Vydal ÚV Svazarmu v 602. základní organizaci Svazarmu, ul. Dr. Z. Winttra 8, 160 41 Praha 6 pro potřeby vlastního aktivu. Vytiskly Tiskařské závody Praha. Náklad 2000 výtisků. Neprodejné. Praha 1987.

**Tiskařské závody, n. p., závod 6, provoz 66, Příbram IV, Anenská 348**

