

**TESLA**

**Výskumno-realizačný ústav spotrebnej elektroniky, k. ú. o.  
Bratislava**

**MIKROPOČÍTAČ SB-8035**

**PRÍRUČKA UŽÍVATEĽA**

**BRATISLAVA 1985**

Obsah:

Všeobecný popis mikropočítačového systému SB-8035	1
Monitor SB-8035	12
Popis podprogramov monitoru SB-8035 vhodných pre užívateľa	30

Prílohy:

Schéma zapojenia mikropočítača SB-8035
Rozloženie súčiastok na doske plošného spoja
Bloková schéma mikropočítača SB-8035
Rozpiska súčiastok
Výpis programov monitora SB-8035

## Všeobecný popis mikropočítačového systému SB - 8035

Popis obsahuje základné technické informácie potrebné pre prácu so systémom. Zahrňuje popis hlavných funkčných blokov, technického riešenia, možnosti komunikácie s vonkajším prostredím a adresáciu pamäťových blokov. Uvedený je aj popis konektorov a funkčných prepojení. Jadrom systému je mikropočítač MHB 8035 bez vnútornej pamäti programu. Spolu s ďalšími obvody vytvára ucelený systém, ktorý užívateľovi umožňuje prakticky sa zoznámiť s činnosťou jednočipových mikropočítačov radu 8048, ich vlastnosťami, inštrukčným súborom, overiť nasadenie jednočipových mikropočítačov a systémov na ich báze v jednoduchých riadiacich aplikáciách.

Systém SB-8035 je vhodný aj pre potreby školení mikropočítačovej techniky a programovanie v strojovom kóde mikropočítačov radu 8048. Hoci je systém uzavretý, možno ho prostredníctvom značného počtu V/V liniek určitým spôsobom rozširovať.

Zostavený je výhradne zo súčiastok vyrábaných v ČSSR a krajinách RVHP s výnimkou obvodu 8155, ktorý je však pripravený do výroby v k.p. TESLA Piešťany. Výhodou je aj jednoduché napájanie zo zdroja +5V/1A /napr. zdroj so stabilizátorom MA 7805/.

### 1. Bloková schéma a stručný popis systému SB - 8035

Mikropočítače radu 8048 sú typickými predstaviteľmi počítačov harwardského typu, ktoré majú oddelenú pamäť programu od pamäti dát. Z tohto pohľadu vyplýva aj delenie mikropočítača na jednotlivé funkčné bloky /pozri tiež blokovú schému systému SB-8035, obr. 1/ :

- blok obvodov mikropočítača a dekodovacej logiky
- blok pamäti programu
- blok pamäti dát
- blok displeja a klávesnice
- blok obvodov vstupu/výstupu

Tieto funkčné bloky komunikujú s riadiacim mikropočítačom prostredníctvom multiplexovanej adresno-dátovej zbernice a adresnej zbernice.

1.1 Blok obvodov mikropočítača 8035 a dekodovacej logiky zabezpečuje vykonávanie inštrukcií programu a správny výber jednotlivých pamätí. Spracúva príkazy a dáta z klávesnice a vysielá správy, ktoré sa zobrazujú na displeji. Blok zabezpečuje správnu synchronizáciu činností celého systému, spracúva asynchrónne vstupy, pri nábehu napájacieho napätia inicializuje systém.

1.2 Blok pamäti programu pozostáva z vlastnej pamäti programu EPROM typu K573 PD5 /ZSSR/ alebo MHB 2716 a užívateľskej pamäti programu RAM, pozostávajúcej z dvojice obvodov MHB 2114. Adresný priestor 1 KB /adresy 000 až 3FFH/ pevnej pamäti EPROM je obsadený základným riadiacim programom /monitorom/, horný KB tejto pamäti je dekodovaný do priestoru C00 až FFFH, čiže do najvyššieho Kbyte celého adresného priestoru mikropočítača 8035. Táto oblasť slúži na potreby rozšíreného monitoru, ktorý môže zabezpečovať napr. komunikáciu s vyšším systémom, alebo na odkladanie viacúčelových užívateľských programov. Monitor systému neumožňuje čítanie pamäti EPROM, pretože táto pamäť nie je technickými prostriedkami pre takýto režim sprístupnená. Užívateľská pamäť programu RAM /adresy 400 až 7FFH/ vytvára oblasť, do ktorej užívateľ systému môže vkladať v štádiu vytvárania a oľadovania svoj užívateľský program vo forme strojového kódu a programových konštánt. Táto pamäť z tohto pohľadu pracuje v dvoch režimoch :

- užívateľský / práca pod monitorovským programom /
- programový / režim vykonávania programu užívateľa /

V užívateľskom režime je logikou systému zabezpečené, že si blok pamäti RAM 2114 správa ako pamäť typu RWM, čo znamená, že možno do nej aj zapisovať aj z nej čítať obsah. Na modifikáciu obsahu pamäti slúžia funkčné a hexadecimálne tlačidlá, umožňujúce adresovanie tejto pamäti, krokovanie nahor a nadol a modifikáciu adresovanej pozície pamäti.

V programovom režime sa pamäť správa ako pevná pamäť ROM, údaje nemôžu byť do nej zapisované a užívateľovi nie je prístupná. Mikropočítač z nej len číta operačné kódy a konštanty.

Celá táto činnosť je riadená programovým čítačom /PC register/. Počas vykonávania monitorovských príkazov je najvyšších 64 byte programovej pamäti RAM rezervovaných na odkladanie registrov a vnútornej pamäti dát.

- 1.3 Blok pamäti dát je tvorený nezávislými pamäťovými oblasťami 64 byte vnútornej pamäti dát mikropočítača MHB 8035 a externej pamäti 256 byte kombinovaného obvodu 8155. Vnútoraná pamäť dát je rozčlenená na dve nezávislé banky univerzálnych registrov, 8-úrovňový zásobník /stack/ a 32 byte voľnej pamäti RAM na všeobecné použitie. Podľa rôznych podmienok daných programom môže užívateľ v najlepšom prípade využívať i plnú kapacitu vnútornej pamäti dát.

Externá dátová pamäť slúži na rozšírenie pamäťového priestoru pre odkladanie dát v prípade, že nepostačuje kapacita vnútornej pamäti dát. Externá pamäť dát je prístupná pre užívateľa v oboch režimoch a to v užívateľskom pomocou sústavy funkčných tlačidiel a v programovom jedine pomocou špeciálnych inštrukcií MOVX.

- 1.4 Blok displeja a klávesnice pozostáva z niekoľkých obvodov, ktorých jadro tvorí expandér V/V liniek určený pre obvody radu 8048 - obvod MHB 8243. Tento zabezpečuje prenos dát medzi mikropočítačom a klávesnicou s displejom pod programovým riadením monitoru. Klávesnica predstavuje pole dvoch matic 4 x 4. Jedna matica tvorí hexadecimálne číslice 0,1,...,F, druhá matica je tvorená funkčnými tlačidlami, ktorých môže byť až 16. Monitor pracuje s 8-imi tlačidlami tejto matice. Okrem týchto tlačidiel je v systéme ešte jedno tlačidlo umiestnené mimo oboch matic. Je to tlačidlo RESET pre aktiváciu vstupu RESET mikropočítača pri inicializácii. Displej pracuje v multiplexnom režime a pozostáva z 8-ich 7-segmentových displejov.

- 1.5 Blok vstupov/výstupov predstavuje pre užívateľa prostriedok styku s okolím a umožňuje prenášať riadiace signály a dáta medzi mikropočítačom a riadeným procesom, resp.

zariadením. Súčasne ich možno využiť na generovanie externých zberníc pre rozširovanie pamäťového systému, napr. zálohované pamäti CMOS na odkladanie vyvíjaných programov, programátory pamäti EPROM, sériové komunikačné linky ap. Špeciálnym prípadom je časovač obvodu 8155.

Blok možno rozdeliť na tieto časti :

- priamo ovládané V/V
- nepriamo ovládané V/V
- externý časovač
- programovo riadené špeciálne V/V

Priamo ovládané V/V kanály sú tie, ktoré užívateľ môže využívať priamo použitím príslušných inštrukcií mikropočítača 8035. Týchto je v systéme SB-8035 využiteľných 8 s vlastnosťami tzv. kvaziobojsmerného kanálu, t.j. jeden vývod môže súčasne slúžiť pre vstup aj výstup dát.

Nepriamo ovládané kanály V/V sú kanály obvodu 8155. Tie sú užívateľovi prístupné tzv. pamäťovou adresáciou, t.j. operácie s nimi užívateľ vykonáva obdobne ako čítanie alebo zápis do pamäti. Týchto liniek je 22 a sú tvorené dvomi obojsmernými portami 8-bitovými a jedným 6-bitovým riadiacim.

Obvod 8155 má vstavaný aj 14-bitový časovač so 4 režimami činnosti, čo je možné využívať aj v systéme SB-8035 na realizáciu externého čítača/časovača. Jeho časovú základňu možno odvodiť dvomi spôsobmi a to buď od systémových hodín obvodu 8035 z vývodu  $T_0$  alebo externými hodinovými impulzami v širokom frekvenčnom rozsahu. Takto má užívateľ možnosť generovať široké spektrum časových oneskorení.

Programovo ovládané V/V majú mnohostranné použitie a ich podrobnejší popis je zahrnutý v popise obvodu 8035. Mikropočítač SB-8035 dovoľuje využívať vstupy/výstupy  $\overline{INT}$ ,  $T_0$  a  $T_1$ .

## 2. Programové vybavenie systému

Programové vybavenie je vytvorené na základe samostatných modulov vo forme podprogramov, ktoré sú pospájané do výsledných monitorovských programov. Základnú časť monitoru tvoria

h programy pre komunikáciu s klávesnicou a displejom. Jednotlivé príkazy monitoru sú ako cieľové programy, ktoré sa vyvolávajú stlačením príslušného tlačidla. Kvôli úspore tlačidiel majú niektoré funkčné tlačidlá dvojitý význam.

Základný monitor systému s kapacitou 1 Kbyte poskytuje užívateľovi tieto funkcie :

- a/ vkladať operačný kód inštrukcií do užívateľskej pamäti programu s kapacitou 1 Kbyte od ľubovoľnej adresy v rozsahu bloku 400 až 7FFH,
- b/ prezerat' a menit' obsah užívateľskej pamäti programu od ľubovoľnej adresy smerom nahor a nadol,
- c/ vkladať priamo počiatočné hodnoty dát do externej pamäti dát v rozsahu 256 byte na adresách 00 až FFH,
- d/ prezerat' a menit' obsah externej dátovej pamäti smerom nahor i nadol,
- e/ vkladať počiatočné hodnoty dát do pracovných registrov a internej pamäti dát na čipe mikropočítača,
- f/ prezerat' a menit' obsahy registrov a internej dátovej pamäti po vykonaní časti programu, celého programu alebo po nábehu monitoru,
- g/ odštartovať vykonávanie programu od ľubovoľnej adresy v adresnom priestore pamäti programu bez zastavenia s návratom do monitoru tak, že zostanú zachované obsahy pracovných registrov a pamäti dát,
- h/ odštartovať a ladiť užívateľský program zaťažením ladiacich bodov a postupným krokováním a prezeraním obsahu registrov a pamäti prechádzať cez tieto ladiačie body, až kým sa program nevykonáva správne. Systém umožňuje zadať až 96 ladiacich bodov.

Poznámka :

Mikropočítačový systém SB-8035 je schopný činnosti i bez osadenia obvodom 8155, sú však obmedzené jeho možnosti, hlavne vstupy/výstupy a pamäť dát. Systém v tomto prípade nie je schopný vykonávať monitorovské funkcie uvedené v bodoch c,d. Po technickej stránke možno systém jednoducho rozšíriť v bloku V/V pomocou expandérov MHB 8243, ktorých  $\overline{CS}$  vstup je riadený programovo.

### 3. Technický popis systému

Všetky obvody mikropočítača SB-8035 sú umiestnené na jednej doske s obojstrannými plošnými spojmi rozmerov 255 x 173 mm. Doska zahrňuje aj voľné pole, umožňujúce zapojenie ďalších obvodov podľa potrieb užívateľa. Technické riešenia vyplývajú z vlastností, určenia a základných funkcií typu TEMS 80-03, PMI-80 pre mikroprocesor 8080, ale s odlišnosťami danými špecifickými črtami počítačov harwardského typu, ku ktorým rad 8048 patrí.

Jadrom systému je jednočipový mikropočítač bez vnútornej pamäti programu typu MHB 8035. Užívateľia vlastníaci obvody MHB 8048, resp. MHB 8748 môžu tieto obvody použiť bez zmeny v zapojení, teda zasunúť ich priamo do päťice pre obvod 8035, funkcia systému sa nezmení. Vnútoraná pamäť programu týchto obvodov je v systéme SB-8035 automaticky blokována.

Časovanie systému je určené použitým kryštálom. Niektoré verzie mikropočítačov SB-8035 sú osadené kryštálom s rezonančnou frekvenciou 6 MHz, niektoré s frekvenciou 4,43 MHz a iné s frekvenciou 4,194 MHz. Rezonančnou frekvenciou kryštálu je daná aj perióda systémových hodín  $T_{syst}$ , interval inkrementovania vnútorného čítača/časovača obvodu 8035 -  $T_{minint}$ , maximálna hodnota oneskorenia nastaviteľná interným čítačom/časovačom -  $T_{maxint}$  a obdobne hodnoty  $T_{minext}$  a  $T_{maxext}$  pre externý časovač obvodu 8155 za predpokladu, že synchronizácia je odvodená od výkonnej fázy FV systémových hodín z vývodu TØ. Pre tieto hodnoty a pre inštrukčný cyklus  $T_{cy}$  platí nasledujúca tabuľka :

$f_{kryst}$	$T_{syst}$	$T_{cy}$	$T_{minint}$	$T_{maxint}$	$T_{minext}$	$T_{maxext}$
6	0,5	2,5	80	20,48	1	8,196
4,43	0,677	3,386	108,352	27,738	1,354	11,092
4,194	0,715	3,577	114,449	29,299	1,43	11,715
4	0,75	3,75	120	30,72	1,5	12,288

Údaje pre  $f_{kryst}$  sú v MHz, pre  $T_{syst}$ ,  $T_{cy}$ ,  $T_{minint}$  a  $T_{minext}$  sú v  $\mu s$ , ostatné údaje v ms.



Užívateľ má k dispozícii aj prostriedok asynchrónneho vstupu / vývod  $\overline{\text{INT}}$  /. Tento vývod je pripojený na konektor K1. Riadenie programu po prijatí žiadosti o prerušenie je smerované na adresu 003 pamäti programu. Na tejto adrese je vložená inštrukcia skoku na adresu 403H, čo je pamäťové miesto v užívateľskej pamäti programu. Od adresy 403 možno vkladať obslužný program prerušenia. Obdobne je realizovaná obsluha žiadosti o prerušenie od vnútorného čítača/časovača. Užívateľ musí vkladať tento obslužný program od adresy 407H.

Na konektor K1 je vyvedený aj vývod T0 a vstup TIM IN obvodu 8155. Ak chceme použiť tento vývod vo funkcii zdroja hodinových impulzov pre časovač 8155, treba v programe použiť inštrukciu ENT0 CLK. Pamäťové obvody MHB 2716, 2114 a 8155 sú pripojené k multiplexovanej AD zbernici, ktorá je generovaná prostredníctvom kanála BUS. Obvod 8155 je pripojený priamo, obvody 2716 a 2114 vyžadujú demultiplexovanie tejto zbernice. Preto sa adresy oddeľujú pomocou záchytného registra MH 3212 na aktívnu hranu signálu ALE. Obvod MH 3212 adresuje len 8 bitov, t.j. jednu stránku. Na adresáciu jednotlivých stránok slúžia bity  $P_{20}$  až  $P_{23}$ , ktoré spolu adresujú 16 stránok, t.j. 4 Kbyte. Výber jednotlivých pamätí je riadený adresným dekodérom MH 3205, ktorý pre adresy 000 až 3FFH aktivuje pamäť EPROM MHB 2716 dolný Kbyte / súčasne je vybraná aj pamäť 8155 /, pre adresy 400 až 7FFH adresuje užívateľskú pamäť programu realizovanú obvody MHB 2114, adresy 800 až BFFH sú nevyužitú a priestor C00 až FFFH aktivuje znovu pamäť EPROM MHB 2716 - horný Kbyte. V súčinnosti s výberovými signálmi sa vhodne hradľujú riadiace signály  $\overline{\text{PSEN}}$ ,  $\overline{\text{RD}}$  a  $\overline{\text{WR}}$ . Tieto spolu s invertovaným signálom  $\overline{\text{RESET}}$  sú zosilnené budičom MH 3216. Signál  $\overline{\text{PSEN}}$  je privádzaný priamo na uvoľňovací vstup  $\overline{\text{CE}}$  a zabezpečuje len čítanie pamäti EPROM. Výberový signál pre dvojicu pamätí MHB 2114 je vytvorený logickým súčtom signálov  $\overline{\text{RD}}$ ,  $\overline{\text{WR}}$  a  $\overline{\text{PSEN}}$ . Signály  $\overline{\text{RD}}$ ,  $\overline{\text{WR}}$  zabezpečujú aktiváciu pamäti v užívateľskom režime, signál  $\overline{\text{PSEN}}$  v programovom režime. Súčtový signál je navyše hradlovaný signálom výberu užívateľskej pamäti programu /obvod MH 7410/. V užívateľskom

režime pamäť pracuje ako RWM, t.j. užívateľ ju môže i čítať i modifikovať, v programovom režime je táto pamäť len čítaná a mikropočítač s ňou narába obdobne ako s pamäťou EPROM. U obvodu 8155 možno využiť aj 22 liniek V/V a 14-bitový časovač. Ak chceme pracovať v jednom z týchto režimov, treba v užívateľskom programe vyslať na linku  $P_{26}$  log. 1 čím sa užívateľovi sprístupní register CSR a porty A, B, C a časovač obvodu 8155. Tak možno potom navoliť riadiace slovo a vysielat', resp. prijímať dáta cez V/V linky 8155.

Na doske SB-2035 sú umiestnené dva expandéry MIB 8243. S mikropočítačom komunikujú pomocou 4-bitovej zbernice, ktorá slúži na prenos príkazov i dát. Druh prenosu je určený aktívnymi hranami signálu PROG. Expandér IO 2 je využitý ako obvod interfejsu medzi mikropočítačom a klávesnicou s displejom. Expandér predstavuje vlastne prevodník 4 na 16. V/V linky expandéra sú organizované v štyroch 4-bitových portoch P4 až P7. Porty P5 a P6 sú určené na vysielanie sedemsegmentového kódu pre jednotlivé displeje, ktoré pracujú v multiplexnom režime. Signály pre časový multiplex sú generované z portu P7 vo forme 3-bitového kódu. Konverziu 3 na 8 zabezpečuje dekodér 74 145. Použitím tohto dekodéra bolo možné využiť port P7 expandéra na testovanie klávesnice. Signál časového multiplexu displejov / v jednom časovom okamihu je log. 0 len na jednom vývode dekodéra / je využitý zároveň ako testovacia vzorka pre klávesnicu. Vzorka log. 0 vyslaná na niektorý z riadkov matice sa vyhodnotí na porte P4 vtedy, ak bolo stlačené niektoré tlačidlo. V prípade nestlačeného tlačidla sa na porte P4 načíta hodnota FH, v prípade jedného stlačeného tlačidla niektorý z kódov 7H, BH, DH alebo EH. Kód tlačidla sa zistí z tohto kódu a testovacej vzorky programovými prostriedkami. Snímanie stlačenia tlačidiel je súčasne ošetrené proti záknitom a proti súčasnému stlačeniu viacerých tlačidiel - v tomto prípade sa vyhodnotí kód posledne uvoľneného tlačidla. Tranzistory KSY 63 slúžia ako budiče jednotlivých segmentov, typ KFY 18 sú budiče segmentoviek LQ 410.

Ďalší expandér IO 3 je užívateľský a je plne k dispozícii užívateľovi. V jednom časovom okamihu možno pracovať len

s jedným expandérom, pretože ich výberové signály sú navzájom invertované. V užívateľskom režime je automaticky vybraný expandér pre klávesnicu a displej, po odštartovaní programu môže užívateľ vybrať ľubovoľný z expandérov nastavením logickej úrovne bitu  $P_{27}$  /  $P_{27} = 0$  - vybraný je IO 2,  $P_{27} = 1$  - vybraný je IO 3 /. Na komunikáciu s vyšším systémom slúži sériová linka, realizovaná optočlenom WK 16412. Táto komunikácia je jednosmerná a to z vyššieho systému na systém SB-8035. Sériová informácia sa spracúva testovateľným vstupom T1.

Poznámka :

Port P2 obvodu 8035 sa využíva v systéme na niekoľko funkcií :

- a/ v režime vykonávania programu je cez  $P_{20}$  až  $P_{23}$  vysielaná adresa  $A_8$  až  $A_{11}$  z programového čítača inštrukcií / u obvodu 8048 je táto adresa prítomná na porte P2 v režime vnútornej pamäti programu až od adresy  $400H$  /,
- b/ v režime komunikácie mikropočítača MHB 8035 s expandérom MHB 8243 tvoria bity  $P_{20}$  až  $P_{23}$  interface medzi oboma obvodmi,
- c/ v režime prehliadania a modifikácie užívateľskej pamäti programu a externej pamäti dát slúži na programové generovanie požadovanej adresy, resp. na aktiváciu výberových signálov. Toto vykonáva užívateľ /alebo monitor / programovo na rozdiel od bodu a/, kde generovanie adresy prebieha vnútorne / je dané obsahom PC registra /. Detailnejšie informácie možno nájsť vo firemnej literatúre k obvodom radu 8048.

#### 4. Popis konektorov

Na doske SB-8035 sú tri pozície pre konektory FRB. Označené sú ako K1, K2 a K3. Na konektor K1 sú vyvedené linky portu P1 obvodu MHB 8035, linky kanálov A, B, C obvodu 8155 a niektoré ďalšie signály. Na konektor K2 sú vyvedené linky užívateľského expandéra. Pozícia K3 je určená pre konektor patriaci k voľnému poľu v pravom hornom rohu dosky. Jednotlivé vývody konektorov K1 a K2 sú priradené nasledovne :

#### Konektor K1 /TX 516 48 12/

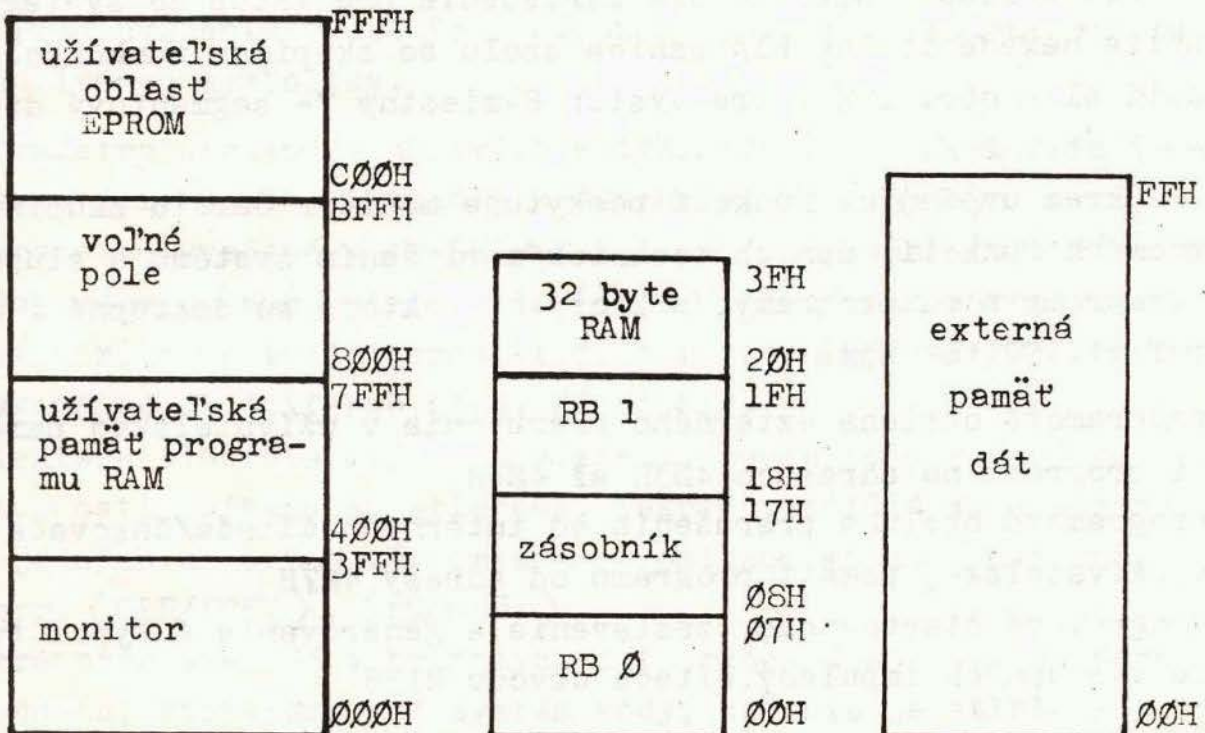
<u>Vývod</u>	<u>signál</u>	<u>Vývod</u>	<u>signál</u>
1	GND	25	PB3/8155
2	GND	26	INT/8035
3	+5V	27	PB2/8155
4	+5V	28	T1/8035
5	PC5/8155	29	PB1/8155
6	MVS+	30	-
7	PC4/8155	31	PB0/8155
8	MVS-	32	P24/8035
9	PC3/8155	33	PA7/8155
10	-	34	P17/8035
11	PC2/8155	35	PA6/8155
12	-	36	P16/8035
13	PC1/8155	37	PA5/8155
14	TIMOUT/8155	38	P15/8035
15	PC0/8155	39	PA4/8155
16	-	40	P14/8035
17	PB7/8155	41	PA3/8155
18	-	42	P13/8035
19	PB6/8155	43	PA2/8155
20	-	44	P12/8035
21	PB5/8155	45	PA1/8155
22	-	46	P11/8035
23	PB4/8155	47	PA0/8155
24	T0/8035+TIMIN/8155	48	P10/8035

Konektor K2 / TX 512 20 12 / :

<u>Vývod</u>	<u>signál</u>	<u>vývod</u>	<u>signál</u>
1	GND	2	GND
3	P50	4	P43
5	P51	6	P42
7	P52	8	P41
9	P53	10	P40
11	P60	12	P73
13	P61	14	P72
15	P62	16	P71
17	P63	18	P70
19	+5V	20	+5V

Poznámka : V prípade využívania sériovej linky prostredníctvom vstupu T1, treba na konektore K1 prepojiť vývody 22/K1 a 28/K1.

Mapa pamäti systému SB - 8035



## 1. Základné vlastnosti

Riadiaci program " Monitor Sb-8035 verzia V-04 " je uložený v pevnej pamäti programu typu EPROM MHB 2716 a zaberá 1Kbyte na adresách 000H až 3FFH. Je určený na riadenie činností a obsluhu mikropočítača Sb-8035.

Monitor zaistuje tieto základné funkcie :

- inicializáciu systému
- prezeranie a zmenu obsahu užívateľskej pamäti programu
- prezeranie a zmenu obsahu pracovných registrov, akumulátora a internej dátovej pamäti
- prezeranie a zmenu obsahu externej dátovej pamäti
- štartovanie užívateľského programu bez zastavenia / funkcia Go no break /
- štartovanie užívateľského programu so zastavením / Go with break /
- ladenie užívateľského programu zmenou obsahu pamäti programu, pracovných registrov, internej a externej dátovej pamäti po zastavení na ladiacom bode / break point /

Ako základné operátorské zariadenie pre vstup do systému je použitá hexadecimálna klávesnica spolu so skupinou funkčných tlačidiel / obr. 1 / a pre výstup 8-miestny 7-segmentový displej / obr. 2 /.

Okrem uvedených funkcií poskytuje monitor ďalšiu skupinu pomocných funkcií, daných technickým riešením systému a služobné programy a podprogramy, z ktorých niektoré sú dostupné i užívateľovi. Sú to najmä :

- programová obsluha externého prerušenia v užívateľskej pamäti programu na adresách 403H až 406H
- programová obsluha prerušenia od interného čítača/časovača v užívateľskej pamäti programu od adresy 407H
- programové štartovanie, zastavenie a generovanie 4 typov tvaru výstupných impulzov čítača obvodu 8155

- programové ovládanie 22 bitov vstupov/výstupov obvodu 8155, ktoré sú adresované ako byty pamäti
- voľne použiteľné služobné podprogramy, napr. zápis a čítanie bytu operačného kódu do pamäti programu, resp. bytu dát ako dátovej pamäti, snímanie matice tlačítok, zobrazenie správy, adresy alebo dát na displeji, ap.

## 2. Popis príkazov monitoru

Jednotlivé funkcie monitoru sa vyvolávajú stlačením zodpovedajúcej postupnosti funkčných /príkazových/ tlačidiel a potrebných hexadecimálnych tlačidiel. Každé tlačidlo klávesnice systému generuje hexadecimálny kód / pozri obr. 1 /, pre číslice sú to kódy 00 až 0FH, pre funkčné tlačidlá / pri plnom osadení klávesnice / kódy 10 až 1FH. Výnimku tvorí tlačidlo <RESET>, ktoré negeneruje kód, ale priamo ovplyvňuje vstup RESET obvodu MHB 8035.

Tlačidlo <RESET> možno stlačiť v ľubovoľnom okamihu vykonávania programu. Ostatné tlačidlá možno stlačiť len za určitého stavu, keď je ich stlačenie povolené, t.j. keď je monitor v stave testovania klávesnice. Inak systém na stlačenie tlačidla nereaguje.

Pred uvedením vlastných príkazov monitoru uvidíme vysvetlivky používanej symboliky.

1. Parametre uvedené v uhlových zátvorkách < > znamenajú stlačenie príslušného tlačidla klávesnice, napr. <Go>
2. Parametre uvedené v okrúhlych zátvorkách predstavujú číselný údaj adresu (adr) alebo dáta (data). Zadávaná adresa môže byť maximálne trojciferná, t.j. v rozsahu 000 až FFFH, dáta maximálne dvojciferné, t.j. 00 až FFH.
3. Niektoré tlačidlá majú dva významy, podľa toho v akej postupnosti príkazu sú stlačené. Takéto tlačidlá sú uvádzané v jedných uhlových zátvorkách s lomítkom medzi funkciami, napr. <CLR/PREV>, <rG/nbrk>
4. Parametre uvedené v hranatých zátvorkách predstavujú náhradnú hodnotu, ktorú priradí systém vždy, keď nie je stlačené

príslušné funkčné tlačidlo a pritom nedôjde k chybe, napr. [nb].

## 2.1 Inicializácia systému - RESET

Stlačením tlačidla <RESET> sa privedie na vývod RESET obvodu MHB 8035 signál a prebehne inicializácia systému, ktorá zahŕňa tieto činnosti :

- čítač programu PC sa nastaví na  $\emptyset\emptyset H$
- ukazovateľ zásobníka SP sa nastaví na  $\emptyset$
- vyberie sa pamäťová banka MB $\emptyset$
- vyberie sa banka registrov RB $\emptyset$
- nastaví sa port 1 a 2 do vstupného módu
- zakáže sa externé prerušenie a prerušenie od čítača/časovača
- zastaví sa čítač/časovač
- nuluje sa príznak preplnenia čítača/časovača TF
- nulujú sa príznakové bity F $\emptyset$  a F1
- zakáže sa výstup výkonnej fázy FV na vývod T $\emptyset$
- nulujú sa výstupné registre /latch/ portov A,B,C obvodu 8155 a všetky 3 porty sa nastaví do vstupného módu
- zastaví sa externý čítač, ktorý je v obvode 8155

Tlačidlo <RESET> je potrebné držať stlačené min. 50 ms. Po uvoľnení tlačidla prebehne inicializačný program monitoru a vypíše sa systémová správa

S b - 8 0 3 5

ktorá značí, že systém očakáva zadanie príkazu z klávesnice.

Poznámka : Stlačenie tlačidla <RESET> neovplyvňuje hodnotu užívateľských registrov, pamäti programu a pamäti dát, nezachováva však ich posledný stav ak bolo stlačené <RESET> počas vykonávania užívateľského programu.

## 2.2 Prezeranie a zmena obsahu pamäti programu, pamätí dát a pracovných registrov

Prezeranie a zmena obsahu pamätí a registrov je inicializovaná stlačením tlačidla <ECh>. Po stlačení tohto tlačidla



sa vypíše na displeji správa " ECh ". Táto odozva znamená, že užívateľ má možnosť čítať a meniť obsah užívateľskej pamäti programu / adresy 400 až 7BFH /, internej dátovej pamäti / adresy 00 až 3FH / a obsah externej dátovej pamäti / adresy 00 až FFH /. Presun na vyššiu adresu v danom type pamäti sa dosiahne stlačením tlačidla <NEXT>, presun na nižšiu adresu stlačením tlačidla <CLR/PREV>. Príkaz "ECh" sa ukončuje stlačením tlačidla <END> .  
V ďalších bodoch je uvedená syntax a používanie príkazov pre jednotlivé typy pamätí :

a/ Prezeranie a zmena obsahu užívateľskej pamäti programu - funkcia ECh Pr

Syntax : <ECh> <PrMEM> (adr) <NEXT> (data) <NEXT> (data) ... <END>  
resp.  
<ECh> <PrMEM> (adr) <NEXT> (data) <CLR/PREV> (data) ...

Náhradná hodnota: PrMEM -[Pr]

Odozva na displeji : EChPr adr - data ...

V prípade, že uložené dáta súhlasia s vkladnými, možno priamo stlačiť tlačidlo <NEXT>, resp. <CLR/PREV>. Stlačenie tlačidla <END> spôsobí ukončenie príkazu a výpis správy " Sb-8C35 " na displeji.

- Poznámky :
- 1/ Adresa môže byť maximálne 3-ciferná, systém akceptuje adresy v rozsahu 000 až FFFH. Dáta na adresách, kde sú uložené pamäte typu EPROM však nemožno čítať / predstavujú operačné kódy / a na displeji sa zobrazuje len dolných 8 bitov príslušnej adresy, napr. pre adresu 2EFH sú to dáta EFH. Toto neplatí ak je pamäť programu prekryvaná s externou pamäťou dát.
  - 2/ Po stlačení chybného tlačidla pri vkladaní adresy alebo dát netreba zopakovať od začiatku celý príkaz, ale stačí vložiť správne číslo. Adresa je platná až po stlačení tlačidla <NEXT> .
  - 3/ Pri vkladaní dát systém nezobrazuje na displeji hodnotu stlačeného tlačidla, ale už hodnotu, ktorú

prečítal z príslušnej adresy po zapísaní hodnoty tlačidla. Pri chybnom zápise sa preto nevypisuje výpis Error, znamenajúci chybu, ale užívateľ má možnosť priamo po zápise skontrolovať, aké dáta sú skutočne v pamäti. Výpis Error sa objaví len v prípade stlačenia nesprávneho tlačidla. Pri vkladaní prvej číslice dát na príslušnú adresu sa nastaví horná číslica priamo na nulu.

- 4/ Pri vkladaní nízkych hodnôt adries , napr. 03AH, netreba nulu na displeji písať, nakoľko sa aj tak nezobrazí. Vždy sa zobrazujú len významné nuly.
- 5/ Tlačidlá <NEXT> a <CLR/ PREV> možno používať v ľubovoľnom poradí; po ich stlačení sa zobrazí obsah najbližšej vyššej, resp. nižšej adresy.
- 6/ Tlačidlo <END> musí byť použité ako posledné tlačidlo príkazu. Môže byť však tiež použité miesto prvého <NEXT> pre zobrazenie obsahu prvej žiadanej adresy. Tlačidlo <CLR/PREV> nesmie byť stlačené pred prvým <NEXT> alebo <END> , inak pôsobí ako zrušenie príkazu.
- 7/ Ak sa miesto tlačidla <PrMEM> stlačí ľubovoľné iné tlačidlo okrem <RESET> , systém priradí náhradnú hodnotu a vypíše na displej " Pr ", čo znamená, že je opäť vybraná užívateľská pamäť programu.

Príklad: Zapište hodnoty 3AH a 7FH na adresy 401H a 402H a spätne prekontrolujte obsah:

Činnosť užívateľa	Obsah displeja
<ECh>	E C h
<PrMEM>	E C h P r
<4>	E C h P r 4
<0>	E C h P r 4 0
<1>	E C h P r 4 0 1
<NEXT>	P r 4 0 1 - X X - náhodné hodnoty
<3>	P r 4 0 1 - 0 3
<A>	P r 4 0 1 - 3 A
<NEXT>	P r 4 0 2 - X X
<7>	P r 4 0 2 - 0 7

<F>	Pr 4 0 2 - 7 F
<CLR/PREV>	Pr 4 0 1 - 3 A
<END>	S b - 8 0 3 5

Poznámka: Prakticky je obmedzená oblasť užívateľskej pamäti programu na adresy 400 až 7BFH, keďže adresy 7C0 až 7FFH využíva monitor na odkladanie užívateľských registrov a internej pamäti dát, ako i niektorých monitorovských premenných.

b/ Prezeranie a zmena obsahu internej dátovej pamäti a pracovných registrov - funkcia ECh rG

Syntax : <ECh> <rG/nbrk> (adr) <NEXT> (data) <NEXT> (data)  
<CLR/PREV> (data) . . . <END>

Náhradná hodnota: PrMEM -[Pr]

Odozva na displeji : ECh rG adr - data ...

Základné pravidlá syntaxe sú ako v prípade a/.

Po vyslaní príkazu sa postupne zobrazujú na displeji obsahy internej dátovej pamäti a užívateľských registrov. Pritom platí:

rG 0 až rG 7	sú pracovné registre banky RB0
rG 8 až rG 16H	sú náhodné hodnoty, ktoré využíva monitor a nemôže ich užívateľ meniť
rG 17H	akumulátor ACC
rG 18 až rG 1FH	sú pracovné registre banky RB1
rG 20 až rG 3FH	sú adresy ďalších bytov internej dátovej pamäti

Poznámky: 1/ Hoci užívateľ môže využívať všetky pracovné registre a akumulátor vo svojom programe, obsah registra rG 1FH /register R-7 banky RB1/ nemožno na displeji čítať ani meniť z klávesnice, keďže je využívaný monitorom.  
2/ Obsahy registrov rG 8 až rG 16H nezodpovedajú skutočným hodnotám, ktoré sú v internej RAM na týchto adresách uložené /oblasť zásobníka/. Znamená to, že z klávesnice užívateľ nemôže túto oblasť pamäti ovplyvňovať. Môže byť však plne využívaná pri vykonávaní užívateľského programu bez zastavenia. V okamihu štartovania a zasta-

venia užívateľského programu je táto oblasť využívaná monitorom.

- 3/ Akumulátor, ktorý nemá v internej dátovej pamäti vlastné miesto, sa odkladá ako register rG 17 a na tejto adrese možno meniť aj jeho obsah.
- 4/ Stavové slovo PSW nie je užívateľovi priamo prístupné. Na zistenie jeho obsahu je potrebné použiť inštrukciu MOV A, PSW.
- 5/ Na adresy internej dátovej pamäti 20 až 3FH možno ľubovoľne vkladať a čítať dáta.

Príklad: Zmeňte obsah akumulátora na hodnotu 47H a obsah registra R1 banky RB1 na 2BH.

Činnosť užívateľa	odozva na displeji
<ECh>	E C h
<rG/abrak>	E C h r G
<1>	E C h r G .1
<7>	E C h r G 1 7
<NEXT>	r G 1 7 - X X náhodné hodnoty
<4>	r G 1 7 - 0 4
<7>	r G 1 7 - 4 7
<NEXT>	r G 1 8 - X X R0 banky RB1
<NEXT>	r G 1 9 - X X R1 banky RB1
<2>	r G 1 9 - 0 2
<B>	r G 1 9 - 2 B
<END>	S b - 8 0 3 5

c/ Prezeranie a zmena obsahu externej dátovej pamäti - funkcia ECh dA

Syntax : <ECh> <dA/brk> (adr) <NEXT> (data) <NEXT> (data) ...  
<CLR/PREV> (data) ...<END>

Náhradná hodnota : PrMEM -[Pr]

Odozva na displeji : ECh dA adr - data ...

Základné pravidlá syntaxe sú ako v prípade a/.

Po vyslaní tohto príkazu je užívateľovi cez monitor sprístupnená externá dátová pamäť na adresách 00 až FFH. Užívateľ môže obdobne ako v prípade a/ prezerat' alebo menit' dáta v tejto pamäti smerom nahor pomocou tlačidla <NEXT>, smerom nadol pomocou tlačidla <CLR/PREV>. Príkaz sa ukončuje tlačidlom <END>.

Poznámka : Z adresácie pamätí vyplýva, že rovnaké dáta ako na adrese 00 až FFH sú i na adresách 100 až 3FFH a 800 až BFFH, ktoré tvoria zrkadlové polia. Treba si uvedomiť, že adresy 000 až 3FFH sa prekrývajú s adresami monitorovskej EPROM a preto nedochádza k čítaniu EPROM, ale pamäti dát.

Príklad : Prekontrolujte externú dátovú pamäť od adresy 9CH po adresu 99H a tam, kde nie sú uložené dáta 46H ich vložte. Uvažujme, že tieto dáta sú uložené na adresách 99, 9A a 9CH.

Činnosť užívateľa	Odozva na displeji
< ECh >	E C h
< dA/brk >	E C h d A
< 9 >	E C h d A 9
< C >	E C h d A 9 C
< NEXT >	d A 9 C - 4 6
< CLR/PREV >	d A 9 b - X X náhodná hodnota
< 4 >	d A 9 b - 0 4
< 6 >	d A 9 b - 4 6
< CLR/PREV >	d A 9 A - 4 6
< CLR/PREV >	d A 9 9 - 4 6
< END >	S b - 8 0 3 5

### 2.3 Štartovanie programu

System Sb-8035 umožňuje štartovanie programu vo dvoch režimoch :

a/ štartovanie bez zastavenia / Go with no break /

b/ štartovanie so zastavením na ladiacom bode / Go with break /

Pri štartovaní programu bez zastavenia má užívateľ možnosť meniť parametre len pred vlastným odštartovaním programu a zistiť výsledky / porovnať hodnoty / až po vykonaní programu.

Pri využití možností ladiacich bodov môže užívateľ meniť parametre po vykonaní určitého počtu / postupnosti / inštrukcií, môže porovnať očakávané a vypočítané hodnoty a na základe toho zmeniť i niektoré časti programu v užívateľskej pamäti programu.

Nakoľko inštrukčný súbor mikropočítačov série 8048 neobsahuje inštrukciu reštartu, ktorú by bolo možné využiť pre zachovanie obsahu registrov, bolo potrebné programovo takúto funkciu zabezpečiť. Program pre odloženie užívateľských registrov a vykonanie niektorých ďalších operácií v monitore sa nazýva RST1. Je nevyhnutné, aby užívateľský program, v ktorom chce užívateľ kontrolovať a meniť obsahy registrov a dátovej pamäti po vykonaní programu alebo jeho časti, obsahoval ako poslednú inštrukciu skok na adresu RST1, t.j. inštrukciu JMP RST1 / operačný kód 24 13 /. Jedine vtedy monitor zabezpečí všetky potrebné úkony pre uchovanie požadovaných parametrov a umožní ich meniť.

Stlačením tlačidla <RESET> pri vykonávaní užívateľského programu sa zničí obsah užívateľských pracovných registrov i niektorých bytov internej dátovej pamäti. Obsah externej dátovej pamäti zostáva zachovaný.

Druhým programom pre odkladanie užívateľských registrov a stavu programu je program RST2. Tento monitorovský program nie je užívateľovi prístupný a slúži na uchovávanie okamžitého obsahu registrov pri zastavení na ladiacom bode. Nakoľko Sb-8035 je jednoprocesorový systém, sú funkcie štartu programu bez a so zastavením realizované na úkor určitých možností systému / obmedzenia / , ktoré budú popísané v ďalšom podľa potreby.

Odstavce a/ a b/ popisujú syntax a príklady použitia oboch príkazov spolu s jednotlivými obmedzeniami.

a/ Štartovanie programu bez zastavenia - Go nb

Syntax : <Go> <rG/nbrk> (adr) <END>

Náhradná hodnota : nbrk - ["nb"]

Odozva na displeji : Go nb adr

Po zadaní tohto príkazu a stlačení tlačidla <END> sa vykoná program od zadanej adresy. Táto adresa môže byť ľubovoľná v rozsahu 000 až FFFH, t.j. možno rovnako štartovať programy v pamäti EPROM ako aj v užívateľskej pamäti programu simulovanej pamäťou RAM. V prípade, že program nie je ukončený inštrukciou JMP RST1 zhasne displej a neustále sa vykonáva užívateľský program. V prípade, že program je zakončený inštrukciou JMP RST1, displej zhasne len počas vykonávania programu a potom sa na ňom vypíše správa " Sb-8035 ", čo znamená, že všetky registre sú zachované a monitor čaká na ďalší príkaz z klávesnice. Užívateľ má možnosť prezerat' si obsahy registrov a pamätí, porovnať a modifikovať ich, prípadne program s novými hodnotami znovu odštartovať.

Poznámky : 1/ Adresy 403 až 406H sú vyhradené pre obsluhu externého prerušenia

2/ Adresy 407 a 408H sú vyhradené pre obsluhu prerušenia od čítača/časovača.

Na týchto adresách môže byť uložený program pokiaľ nie sú prerušenia využívané.

3/ Na adresy 7C0 až 7FFH nemožno vkladať užívateľský program, nakoľko tieto adresy využíva monitor na odkladanie pracovných registrov a internej dátovej pamäti.

4/ Nesprávne odštartovanie a ukončenie programu alebo skok na nesprávnu adresu môže zapríčiniť zničenie obsahu dátovej pamäti alebo aj niektorých bytov užívateľskej pamäti programu.

Príklad : Predpokladajme, že užívateľský program sa štartuje od adresy 67AH a je zakončený inštrukciou JMP RST1. Odštartujme tento program.

Činnosť užívateľa	Odozva na displeji
<Go>	G o
<rG/nbrk>	G o n b
<6>	G o n b        6
<7>	G o n b        6 7
<A>	G o n b        6 7 A
<END>	zhasne displej a po určitom čase sa vypíše správa S b - 8 0 3 5

b/ Štartovanie programu so zastavením na ladiacich bodoch -  
- funkcia Go brk

Syntax :<RESET> <Go> <dA/brk> (adr) <END> <NEXT> <NEXT> ...<END>

Náhradná hodnota : nbrk - ["nb"]

Odozva na displeji : Go brk adr ...

Pred vlastným odštartovaním programu so zastavením na ladiacich bodoch musí užívateľ zadať na vyhradené adresy užívateľskej pamäti programu adresy ladiacich bodov / body na ktorých sa má vykonávanie programu zastaviť /. Tieto adresy sa vkladajú pomocou príkazu ECH Pr pre vkladanie do užívateľskej pamäti programu takto :

Adresa prvého ladiaceho bodu sa vkladá na adresu 7BFH / horný byte / a 7BEH / dolný byte /, adresa druhého bodu obdobne na adresy 7BDH / horný byte / a 7BCH / dolný byte / a takto postupne smerom nadol maximálne až 96 ladiacich bodov, t.j. po adresu 700H. Pritom na adresu najbližšiu nižšiu za adresou posledného ladiaceho bodu treba vložiť hodnotu FFH, čo znamená, že už nižšie nie sú uložené adresy ladiacich bodov. Ak boli teda vložené adresy dvoch ladiacich bodov, treba hodnotu FFH vložiť na adresu 7BBH.

Počet ladiacich bodov treba voliť uvážene, nakoľko znižujú kapacitu voľnej pamäti pre uloženie programu.

Ladiace body nesmú adresovať tieto pamäťové miesta :

- posledný byte stránky, t.j. adresa XFFH
- druhý byte dvojbytovej inštrukcie



- kde je uložený operačný kód inštrukcie návratu z podprogramu RET alebo RETR ani miesto na adrese o jedno nižšej
- pevnú pamäť programu typu ROM, PROM, EPROM

Užívateľský program, ktorý sa štartuje so zastavením na ladiacich bodoch, by mal byť ukončený inštrukciou JMP RST1. Postup pri vlastnom ladení programu pomocou príkazu pre štartovanie so zastavením na ladiacich bodoch :

- 1/ Zadáme ladiace body podľa vyššie uvedeného popisu na adresu 7BFH a nižšie. Ako poslednú hodnotu použijeme hodnotu FFH. Každá adresa musí obsahovať dva byty !
- 2/ Stlačíme tlačidlo <RESET>, čím sa inicializuje systém a monitorovské registre potrebné pre funkciu " Go brk ".
- 3/ Odštartujeme vložený program od štartovacej adresy podľa postupnosti uvedenej v syntaxi príkazu " Go brk ", napr. od adresy 439H takto :

<Go><dA/brk> (439) <END>

- 4/ Vykoná sa program po najbližší ladiaci bod a na displeji sa vypíše správa :

P C    adr    data

čo značí, že najbližšia inštrukcia sa bude vykonávať na adrese (adr) / hodnota čítača programu PC / a okamžitý obsah akumulátora je rovný hodnote (data), t.j. ACC = (data) .

- 5/ V tomto okamihu má užívateľ tieto možnosti :

- a/ stlačiť tlačidlo <NEXT>, čím sa vykoná program po ďalší ladiaci bod a vypíše sa na displeji obdobná správa ako v bode 4/,
- b/ stlačiť tlačidlo <END>, pričom sa vypíše systémová správa " Sb-8035 " a užívateľ môže vkladať nové príkazy / nevykonáva sa už funkcia " Go brk " /,
- c/ stlačiť tlačidlo <ECH>, čím sa dostane do režimu modifikácie pamäti programu, dátovej pamäti a pracovných registrov a podľa kapitoly 2.2 môže priebežne kontrolovať výsledky, prípadne i meniť program. Prezeranie a modifi-

káciu registrov a pamäti treba ukončiť stlačením tlačidla  $\langle \text{END} \rangle$ , na čo sa opäť vypíše systémová správa " Sb-8035 ". V tomto okamihu možno opäť vybrať niektorú z možností a/,b/,c/.

- 6/ Postupným prechádzaním cez ladiace body stlačením tlačidla  $\langle \text{NEXT} \rangle$ , prípadne v kombinácii s funkciou " ECh " sa vykoná užívateľský program až po inštrukciu JMP RST1. Jej vykonanie sa prejaví na displeji tým, že miesto výpisu

P C    adr    data

sa vypíše systémová správa " Sb-8035 " a ďalšie stláčanie tlačidla  $\langle \text{NEXT} \rangle$  alebo  $\langle \text{END} \rangle$  už tento výpis neovplyvňuje.

- Poznámky : 1/ Program môže byť ukončený aj skôr, než boli vyčerpané všetky ladiace body, napr. vtedy, ak sú v programe inštrukcie skokov a ladiaci bod ukazuje na tú časť programu, v ktorej sa s danými parametrami program vôbec nevykonáva. V prípade, že nebol tento ladiaci bod dosiahnutý, nedôjde k zastaveniu ani na ďalších bodoch a program sa vykoná až dokonca bez zastavenia, čo sa indikuje výpisom správy " Sb-8035 ".
- 2/ Vzhľadom nato, že pri behu programu s ladiacimi bodmi dochádza k neustálej interakcii monitoru a užívateľského programu, nemožno ladiaci bod zadať vo vnorených podprogramoch väčšej hĺbky ako dva. Preto odporúčame užívateľovi podprogramy odladiť pred ich použitím samostatne a ladiace body používať predovšetkým v hlavnom programe.
- 3/ Pri štartovaní so zastavením na ladiacich bodoch nemožno ladiť tie časti programov, ktoré pracujú s časovačom/ čítačom, externým prerušením, prerušením od čítača/časovača ani pracujúce s externým čítačom obvodu 8155.
- 4/ Nedodržanie všetkých uvádzaných podmienok pre štartovanie programu so zastavením na ladiacich bodoch, môže zapríčiniť nedefinovateľné výsledky včítane zničenia obsahov registrov a pamäti.

Príklad : Nech v užívateľskej pamäti programu je na adresách 450 až 45AH uložený operačný kód nižšie uvedeného programu. Odštartujeme program so zastavením, pričom ladiace body sú na adresách 454 a 457H.

450	23 21	SCIT2:	MOV	A, #21H
	B8 30		MOV	R0, #30H
454	B0 43		MOV	2R0, #43H
	60		ADD	A, 2R0
457	18		INC	R0
	A0		MOV	2R0, A
459	24 1B		JMP	RST1

1/ Vložíme hodnoty ladiacich bodov na adresu 7BFH a nižšie. Kvôli zjednodušeniu budeme tlačidlá číslíc písať spolu v číslach, napr. <450> :

Činnosť užívateľa	Odozva na displeji
<ECh> <Pr> <7BF>	E C h P r - 7 B F
<NEXT>	P r 7 B F - X X - náhodná hodn.
<04>	P r 7 B F - 0 4
<CLR/PREV>	P r 7 B E - X X
<54>	P r 7 B E - 5 4
<CLR/PREV>	P r 7 B D - X X
<04>	P r 7 B D - 0 4
<CLR/PREV>	P r 7 B C - X X
<57>	P r 7 B C - 5 7
<CLR/PREV>	P r 7 B B - X X
<FF>	P r 7 B B - F F - ukonč. znak

2. Odštartujeme program od adresy 450H a krojujeme cez ladiace body :

<Go> <dA/brk> <450>	G o b r 4 5 0
<END>	P C 4 5 4 2 1 hodn. l. sčít.
<ECh> <rG/nb> <0>	E C h r G 0
<NEXT>	r G 0 3 0 hodn. v R0
<END>	S b - 8 0 3 5

<NEXT> P C 4 5 7 6 4 hodnota súčtu  
v ACC  
<NEXT> S b - 8 Ø 3 5  
<ECh> <rG/nbrk> <17> E C h r G 1 7 nastav sa na ACC  
<NEXT> r G 1 7 - 6 4 obsah ACC je  
rovný súčtu  
.  
.  
.

#### 2.4 Zrušenie príkazu - funkcia CLR

System Sb-8035 je vybavený tlačidlom <CLR/PREV>, ktoré sa používa na zrušenie príkazu pri súčasnom zachovaní obsahu registrov, pamäti a vstupov/výstupov. Po jeho stlačení sa zruší predvolená / zadaná / časť príkazu a očakáva sa zadanie nového príkazu. Toto je indikované systémovou správou

S b - 8 Ø 3 5

Tlačidlom <CLR/PREV> možno zrušiť chybný príkaz jedine vtedy, ak ešte nebolo stlačené tlačidlo <END>. Ďalej slúži tlačidlo <CLR/PREV> na návrat do monitoru po chybovom výpise Error X, kde X je kód chyby.

Príklad : Miesto príkazu Go brk chceme zadať príkaz Go nbrk so zachovaním vstupov/výstupov a obsahov registrov a pamäti.

Zadaná zlá postupnosť :

<Go> <dA/brk> (adr) - v tomto okamihu možno stlačiť tlačidlo  
<CLR/PREV> - vypíše sa správa S b - 8 Ø 3 5  
a zadáme správnu postupnosť

<Go> <rG/nbrk> (adr) <END> - na čo sa odštartuje program bez zastavenia.

#### 2.5 Chybové výpisy

Pri práci s monitorom systému Sb-8035 sa môže užívateľ dopustiť niekoľkých typov chýb, ktoré sú indikované výpisom E r r o r X na displeji, kde X je kód chyby :

- Error 1 - stlačené funkčné tlačidlo nedefinuje platný príkaz. Ako prvé po tlačidle <RESET> môže užívateľ stlačiť jedine tlačidlo <ECh> alebo <Go> .
- Error 2 - stlačené tlačidlo nie je z množiny hexadecimálnych číslíc 0 až FH alebo je zle ukončený príkaz
- Error 3 - nesprávne ukončenie príkazu ECh pri prezeraní alebo zmene kódu a dát / nebolo stlačené tlačidlo <END>/.
- Error 4 - nesprávny pokus o návrat do monitoru pri výpise chyby. Chybový výpis možno zrušiť jedine tlačidlom <END> alebo <CLR/PREV> .

### 3. Mapovanie systému

#### 3.1 Adresácia pamäti programu

000 až 3FFH	EPROM 2716 dolný Kbyte - monitor systému
400 až 7BFH	RAM 2x2114 - užívateľská pamäť programu 960 byte
7C0 až 7FFH	RAM 2x2114 - oblasť pre odkladanie internej dátovej pamäti a pracovných registrov
800 až FFFH	nevyužitá
403 až 406H	adresy vyhradené pre obsluhu externého prerušenia
407 -	adresy vyhradené pre obsluhu prerušenia od čítača/časovača

#### 3.2 Adresácia internej dátovej pamäti

00 až 07	pracovné registre R0 až R7 banky RB0
08 až 16H	oblasť stacku neprístupná cez klávesnicu užívateľovi - využívaná monitorom
17H	akumulátor ACC prístupný z klávesnice
18 až 1EH	pracovné registre R0 až R6 banky RB1
1FH	pracovný register R7 banky RB1 - neprístupný z klávesnice užívateľovi
20 až 3FH	interná RAM všeobecného použitia prístupná užívateľovi z klávesnice

Poznámka : Pri vykonávaní užívateľského programu bez zastavenia sú od okamihu odštartovania užívateľovi k dispozícii všetky pracovné registre, oblasť stacku i ostatné byty internej RAM všeobecného určenia podľa katalógového listu výrobcu obvodov MHB 8048/8035. Vyššie uvádzané obmedzenia sa vzťahujú na zadávanie dát do internej RAM a registrov z klávesnice a pri vykonávaní programu so zastavením na ladiacich bodoch.

### 3.3 Externá dátová pamäť a vstupno/výstupné porty 8155

00 až FFH externá dátová pamäť 256 byte  
 100 až 3FFH zrkadlové pole  
 800 až BFFH zrkadlové pole  
 4000H CSR register V/V obvodu 8155  
 4001H port A 8 bitov V/V  
 4002H port B 8 bitov V/V  
 4003H port C 6 bitov V/V  
 4004H dolných 8 bitov dátového registra čítača 8155  
 4005H horných 6 bitov dátového registra čítača 8155 a 2 bity pre mód čítača

Poznámka : Adresy 4000 až 4005H sú fiktívne a nie sú prístupné z klávesnice.

Spolu s 8-imi bitmi V/V priamo na čipe má užívateľ celkovo k dispozícii 30 bitov vstupu/výstupu.

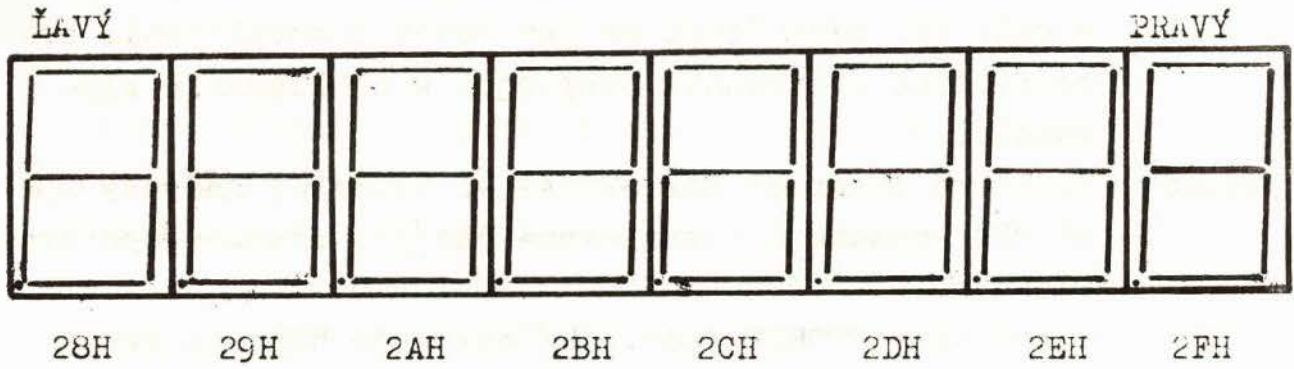
Príloha 1 : Priradenie kódov tlačidlám klávesnice :

Λ	Λ	MEM	MEM
1C	1D	1E	1F
Λ	Λ	MEM	MEM
18	19	1A	1B
Λ	Λ	MEM	MEM
14	15	16	17
Λ	Λ	MEM	MEM
10	11	12	13

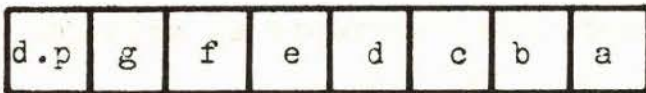
Λ	Λ	Λ	Λ
0C	0D	0E	0F
Λ	Λ	Λ	Λ
08	09	0A	0B
Λ	Λ	Λ	Λ
04	05	06	07
Λ	Λ	Λ	Λ
00	01	02	03

Obr. 1

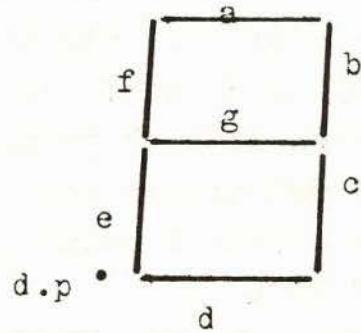
Priradenie adries internej dátovej pamäti pozíciám displeja



Obr. 2



Segmenty displeja v dátovom byte



Popis podprogramov monitoru Sb-8035 vhodných pre užívateľa

**INPKEY** - adr. 0135H Podprogram pre vstup z klávesnice a oživo-  
vovanie displeja. Využíva interný čítač  
v móde bez prerušenia na generovanie oneskorenia pre  
pravidelné oživovanie displejov a obhliadanie klá-  
vesnice.

Vstupy : Registre internej RAM SEGMAP až SEGMAP+7 /adresy 28H  
až 2FH/ obsahujú zobrazované údaje v sedemsegmentovom  
kóde.

- register KBDBUF /adr.26H/ obsahuje hodnotu FFH
- stlačenie tlačidla v matici klávesnice
- odštartovanie interného čítača/časovača

Výstupy: ACC a R3 obsahujú kód stlačeného tlačidla /00H až 1FH/  
- ak došlo k preplneniu čítača, postupne sa oživujú  
jednotlivé displeje  
- návrat z podprogramu je až po stlačení tlačidla

Volá : TOFPOL

Registre:ACC, R1, R3, KBDBUF

Poznámka: Podprogram reaguje na stlačenie tlačidla, nie až na  
jeho uvoľnenie. Snímanie klávesnice je ošetrené pro-  
ti zákmitom a stlačeniu viacerých tlačidiel

Príklad 1 : Nčítajte tlačidlo z klávesnice a jeho kód zapište  
na adresu 3FH internej RAM :

```
MOV    R0,#26H
MOV    @R0,#0FFH
STRT   T
CALL   INPKEY
MOV    R0,#3FH
MOV    @R0,A
:
:
:
```

Príklad 2 : Zobrazte na displeji SEGMAP /adr. 28H/ číslicu 8:

```
MOV    R0,#26H
MOV    @R0,#0FFH
STRT   T
MOV    R0,#28H
```



```
MOV    R0, #7FH
CALL   INPKEY
      .
      .
      .
```

V tomto prípade bude číslica 8 rozsvietená dovtedy, kým nebude stlačené ľubovoľné tlačidlo.

Poznámka : Inicializáciu registra KBDBUF na hodnotu 0FFH je potrebné robiť len pred prvým vyvolaním podprogramu INPKEY.

**TIINT** - adr. 0146H Podprogram pre oživovanie displejov.

Tento podprogram môže pracovať tak v móde čítača/časovača s prerušením alebo bez prerušenia. Slúži na pravidelné oživovanie displejov. I keď súčasťou tohto podprogramu je snímanie matice tlačidiel, nie je sám o sebe použiteľný pre snímanie /zosníme len jeden riadok/. Tento podprogram je vhodné používať v prípadoch, keď počas vykonávania užívateľského programu má byť použitý i displej. Toto je umožnené v móde s povolením prerušenia od čítača.

Vstupy : Registre SEGMAP až SEGMAP+7 /adr. 28H až 2FH/ obsahujú 7-segmentový kód zobrazovanej správy

Výstupy: Oživuje sa jeden stĺpec a jeden riadok tlačidiel. Interval prerušenia čítača je nastavený na cca 1 ms, t.j. každú ms sa nastavuje TF a v prípade povolenia prerušenia od čítača/časovača sa vykoná prerušenie.

Ničí : R0 až R7 banky R01, KEYLOC /adr.27H/, KBDBUF /adr. 26H/, NREPTS /adr. 20H/

Volá : -

Príklad : Vložte do registra SEGMAP+7 /adr.2FH/ číslicu 8 a zobrazte na displeji tak, aby bolo možné vykonávať ďalší užívateľský program :

Na adresu 407H treba vložiť :

```
CALL TIINT
RETR
```

Pred užívateľský program treba nevyhnutne vložiť tieto inštrukcie :

```
EN      TCNTI
STRT    T
MOV     R0,#SEGMAP+7    tu už začína program uží-
MOV     @R0,#7FH        vateľa
.
```

Poznámka : Ak podprogram TIINT pracuje v móde bez povolenia prerušenia od čítača, musí byť programovou sľučkou zabezpečené jeho vyvolanie minimálne raz za 1 ms, inak displej zhasne alebo bude blikať.

**CLEAR** adr. 0127H Vymazanie registrov displejov.

Tento podprogram slúži na vymazanie / naplnenie nulami / registrov 7-segmentových displejov SEGMAP až SEGMAP+7 /adr. 28H až 2FH/.

Vstupy : žiadne

Výstup : nulujú sa všetky registre displejov

Ničí : R0,R1, NEXTPL /adr.25H/

Príklad : Vymažte displej / prejaví sa zhasnutím a na adresách 28H až 2FH budú zapísané 00H /.

```
CALL    CLEAR
STRT    T
CALL    INPKEY
```

.

**RST1** adr. 011BH Reštart monitoru.

Tento program zabezpečuje po vykonaní užívateľského programu návrat do monitoru. Obsahy užívateľských registrov a internej RAM sa pritom zachovávajú / s určitými obmedzeniami , ktoré sú popísané v " Monitor Sb-8035 " /. Skok na program RST1 treba použiť ako poslednú inštrukciu užívateľského programu, v ktorom chceme prezerať alebo meniť registre a obsah internej RAM. Po vykonaní tohto pro-

gramu sa vypíše správa " Sb-8035 " a užívateľ môže vkladať príkazy z klávesnice.

**DSPACC**

adr. 0196H Zápis číslice z ACC do registra displeja  
Tento podprogram vykonáva konverziu dolných 4 bitov ACC /horné môžu byť ľubovoľné/ na 7-segmentový kód a jeho zápis do príslušného registra displeja. Pozíciu registra displeja udáva adresa 25H internej RAM označená symbolicky NEXTPL. Po vykonaní podprogramu DSPACC sa hodnota NEXTPL dekrementuje o jedničku.

Vstupy : ACC - dolné 4 bity obsahujú hexa kód číslice  
NEXTPL - adresa príslušného displeja / 1 až 8 /

Výstup : 7-segmentový kód sa zapíše do registra R6 a príslušného registra displeja  
NEXTPL = NEXTPL-1

Ničí : ACC, R1, R6, NEXTPL

Volá : -

Poznámka : Ak pred vykonaním podprogramu DSPACC ukazuje NEXTPL na prvý displej sprava, po vykonaní podprogramu sa nastaví na prvý displej zľava.

**WDISP**

- adr. 019BH Zápis 7-segmentového kódu do registra displeja.

Tento podprogram zapisuje 7-segmentový kód z ACC do niektorého z registrov displeja podľa hodnoty NEXTPL. Pre NEXTPL = 8 je to ľavý displej, pre NEXTPL = 1 je to pravý displej.

Vstupy : ACC - 7-segmentový kód znaku  
NEXTPL - pozícia displeja

Výstup : Obsah ACC sa prepíše do registra určeného NEXTPL  
NEXTPL sa zníži o 1

Ničí : ako DSPACC

Volania: -

Príklad : Zapište do registra 29H /2.displ. zľava/ kód číslice 4 a do reg. 2AH /3.disp. zľava/ kód písmena P. Ostatné displeje vyprázdňte, znaky rozsvieťte!

CALL	CLEAR	vyprázdni displeje
MOV	A,#04	číslíca 4 do ACC
MOV	R1,#NEXTPL	nastav na zápis
MOV	R1,#07	register 29H
CALL	DSPACC	zapiš 7-segm. kód číslice 4
MOV	A,#73H	zapiš 7-segm. kód znaku "P"
CALL	WDISP	do reg. 2AH
MOV	R0,#KBDBUF	adresa 26H
MOV	R0,#0FFH	inicializuj vstup z klávesnice
STRT	T	
CALL	INPKEY	rozsvieť hodnotu "4P"
	:	

**PERROR** - adr. 020BH Chybový výpis " Error X "

Program podľa obsahu dolných 4 bitov registra R2 vypíše na displej chybovú správu " Error X ", kde X je kód chyby a vráti riadenie monitoru. Výpis možno zrušiť stlačením tlačidiel <END> alebo ~~NEXT~~xx <CLR/PREV>.

Vstupy : R2 - kód chyby 0 až F

Výstup : Vypíše sa správa "Error X", kde X je kód chyby

Ničí : ACC, R1 až R3, PSW, SEGMAP až SEGMAP+7, KBDBUF

Poznámka : Tento program sa nevyvoláva inštrukciou CALL, ale inštrukciou skoku JMP. Pred skokom na program PERROR treba odštartovať čítač inštrukciou STRT T.

**UPDADR** - adr. 01D4H Zapísanie 3-ciferného čísla do registrov displeja.

Tento podprogram prevedie 3-cifernú adresu uloženú v internej RAM na adresách daných R0 a R0-1 na 3 znaky v 7-segmentovom kóde a uloží ich na prvé tri registre displejov sprava. Pritom R0 ukazuje adresu RAM, kde sú platné iba dolné 4 bity, ktoré predstavujú najvýznamnejšiu číslicu 3-ciferného čísla. Adresa R0-1 obsahuje dolných 8 bitov čísla. Nevýznamné nuly sa reprezentujú ako prázdne znaky.

Vstupy : R0 - adr. internej RAM, kde je uložený vyšší byte čísla / len 4 bity /  
R0-1 nižší byte čísla

Výstup : Prvé tri registre displejov sprava obsahujú 7-segm. kód čísla uloženého na adrese R0 a R0-1

Ničí : R0, ACC, R1, NEXTPL, SEGMAP+5 až SEGMAP+7

Volá : WDISP, DSPACC

Príklad : Zobrazte na displeji hodnotu 401H na prvých troch displejoch sprava

```
MOV    R0, #30H
MOV    @R0, #01      dolný byte uložíme na adresu 30H
INC    R0
MOV    @R0, #04      horné 4 bity na adresu 31H
CALL   UPDADR
STRT   T
MOV    R0, #KBDBUF
MOV    @R0, #0FFH
CALL   INPKEY
      :
```

**UPDAD1** - adr. 01D8H Zapísanie 3-ciferného čísla do ľubovolnej trojice displejov za sebou

Vstupy : ako u UPDADR, navyiac NEXTPL /adr.25H/ musí udávať pozíciu registra pre najvýznamnejšiu číslicu. Napr. ak má byť číslo zobrazené na prvé tri displeje zľava, treba vložiť do NEXTPL hodnotu 8.

Výstup : obdobne ako u UPDADR, len iné pozície displejov

Ničí : ako UPDADR

Volá : WDISP, DSPACC

**DSPMID** - adr. 01ECH - zápis bytu do registrov displeja  
Tento podprogram prevedie 2 ciferné číslo uložené v internej RAM na adrese danej R0 na 2 znaky v 7-segmentovom kóde a zapíše ich do dvoch susedných registrov displeja, ktorých pozícia je určená obsahom NEXTPL /adr.25H/

Vstupy : R0 - adresa internej RAM, kde je uložené číslo  
NEXTPL - pozícia registra displeja významnejšej  
číslice

Výstup : Číslo uložené na adrese danej R0 sa prevedie na  
7-segmentový kód a zapíše do registrov displeja  
adresovaných NEXTPL a NEXTPL-1

Ničí : ACC, R0 a príslušné registre SEGMAP, NEXTPL

Volá : DSPACC

Príklad : Zobrazte obsah adresy 3FH na prvé 2 pozície dis-  
plejov sprava

```
MOV    R0,#3FH
MOV    R1,#NEXTPL
MOV    DR1,#02
CALL   DSPMID
MOV    R0,#KBDBUF
MOV    DR0,#0FFH
STRT   T
CALL   INPKEY
      :
```

**INPADR** - adr. 01B0H Vkladanie 3-ciferného čísla z klávesnice  
Tento podprogram umožňuje vkladat' 3-ci-  
ferné číslo /adresu/ z klávesnice a zobrazit' ho na  
3 krajné displeje sprava. Program pracuje tak, že  
kým sú stlačené len tlačidlá predstavujúce číslice  
0 až F, posúva sa vždy číslo o jedno miesto vľavo.  
Pritom nová číslica zaujme najnižší rád a číslica  
z najvyššieho rádu sa stratí. Stlačením niektorého  
z tlačidiel <NEXT> alebo <END> sa podprogram ukončí  
a vráti sa riadenie do hlavného programu. Ak bolo  
stlačené iné funkčné tlačidlo, vypíše sa správa  
" Error 2 " a vráti sa riadenie do monitoru systému.  
Vkladané číslo /adresa/ sa zároveň zapíše do inter-  
nej RAM na adresy dané registrom R0 / dolné 2 čísli-  
ce / a R0+1 / najvýznamnejšia číslica /.

Pri vstupe do tohto podprogramu sa predpokladá, že register R3 obsahuje už prvú platnú číslicu. V inom prípade treba najskôr vyvolať podprogram INPKEY.

Vstupy : R0 - adresa internej RAM, kde sa uloží dolný byte čísla /adresy/

R3 - obsahuje prvú číslicu

Výstup : Vždy po stlačení hexa tlačidla sa zapíše do RAM na adresu danú R0 a R0+1 aktuálna hodnota čísla, pričom posledne stlačená číslica sa dopĺňa sprava a predchádzajúca najvýznamnejšia sa stráca. Aktuálne číslo sa zobrazí na prvých troch displejoch sprava. Návrat z podprogramu nastane stlačením tlačidla <NEXT> alebo <END> .

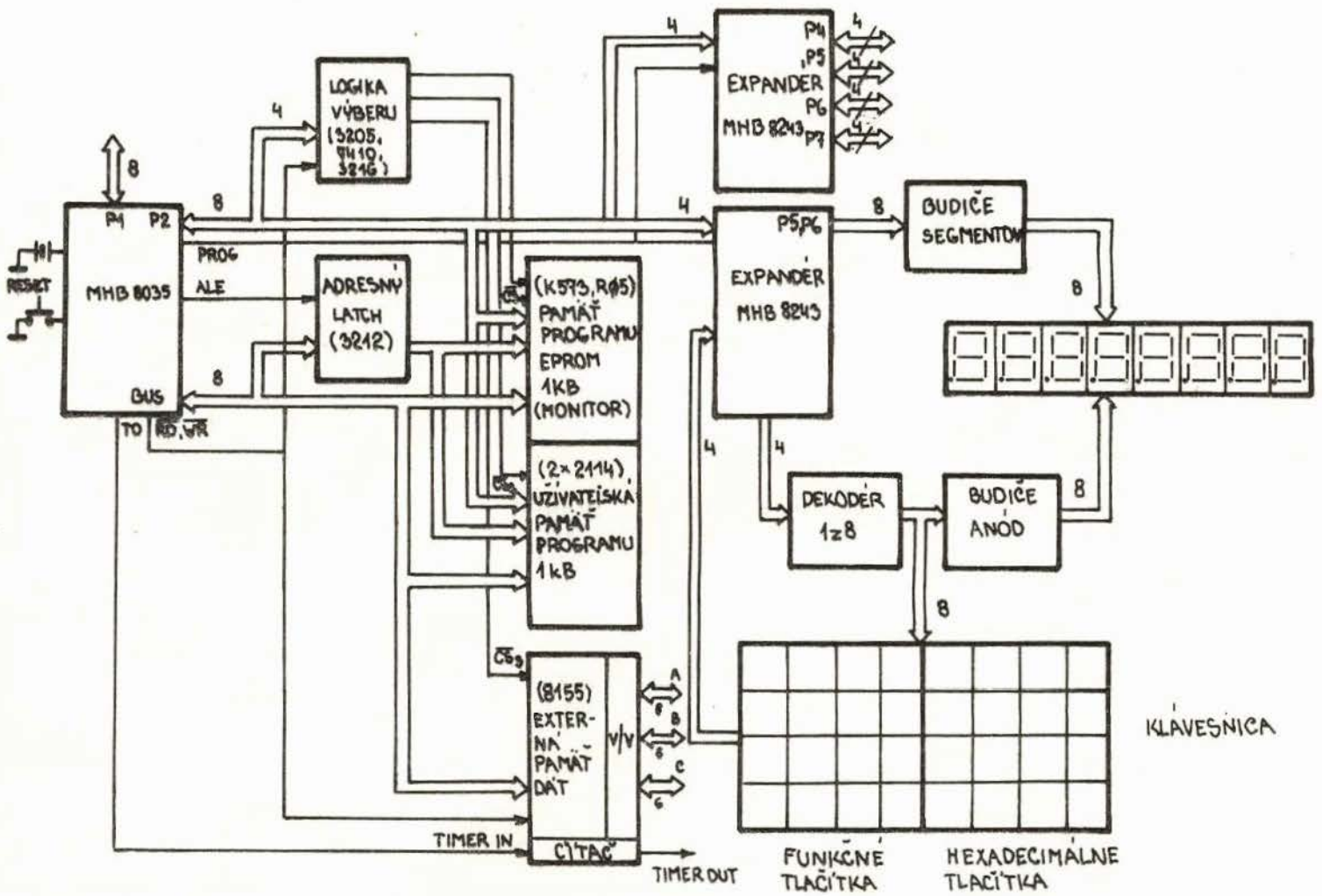
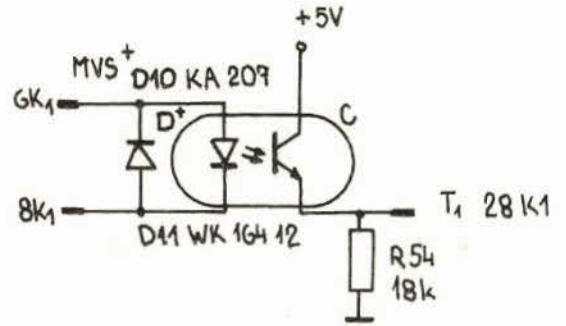
Ničí : R0 až R3, ACC

Volá : UPADR, INPKEY, DBLANK, v prípade chyby skáče na PERROR.

Príklad : Napíšte podprogram pre načítanie 3-ciferného čísla z klávesnice, zobrazenie na displeji a zapísanie hodnoty na adresy 30H a 31H internej RAM.

```
MOV    R0,#KBDBUF
MOV    @R0,#0FFH
CALL   CLEAR          nuluj displeje
STRT   T
CALL   INPKEY         načítaj prvú číslicu
MOV    R0,#30H
CALL   INPADR
RET
```

K ukončeniu podprogramu dôjde až po stlačení tlačidla <END> alebo <NEXT> .



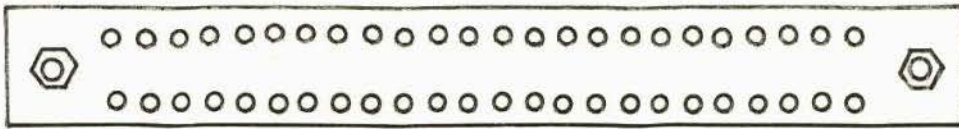
BLOKOVÁ SCHEMA SYSTÉMU SB-8035



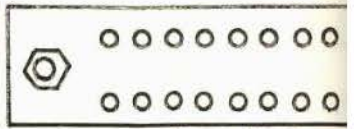


KY 132/80

D9



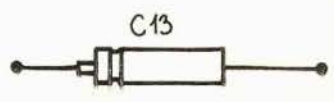
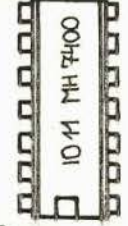
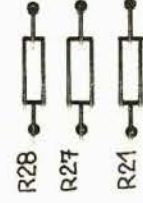
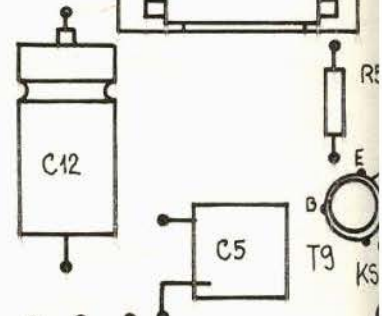
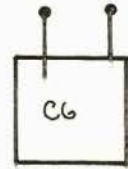
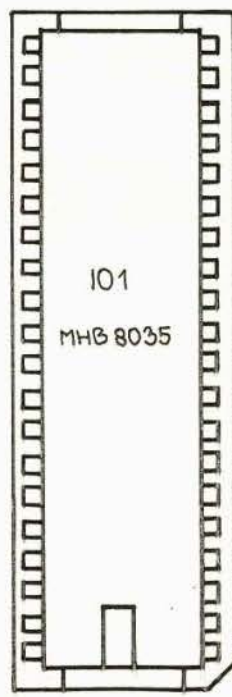
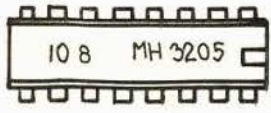
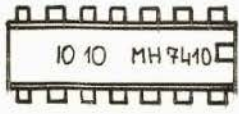
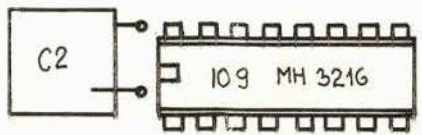
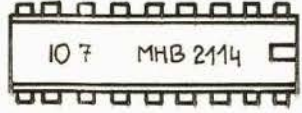
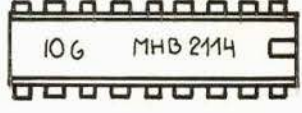
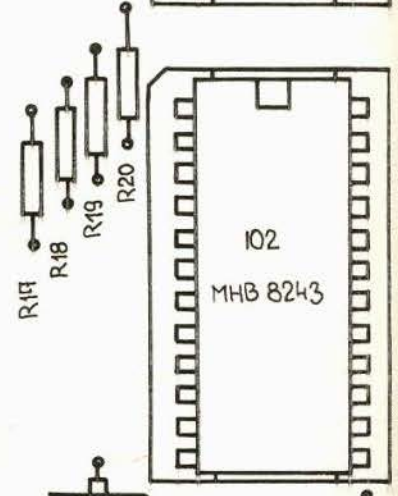
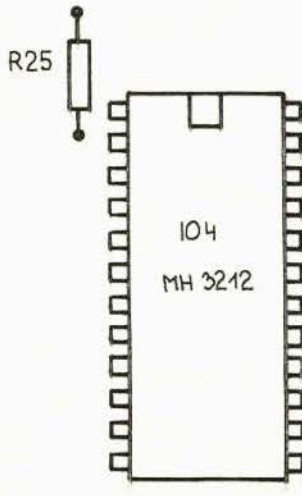
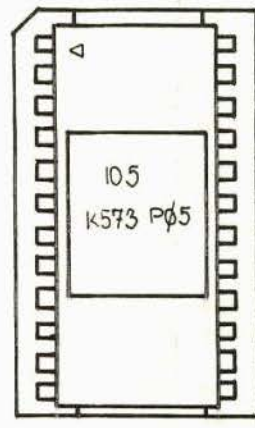
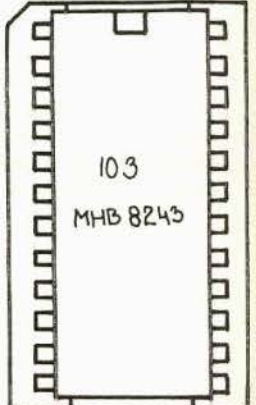
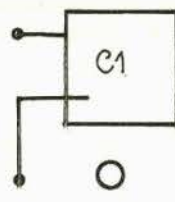
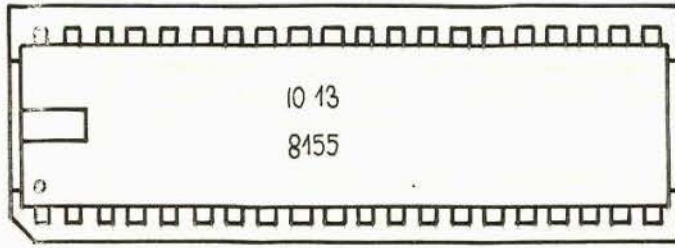
TX 516 4812

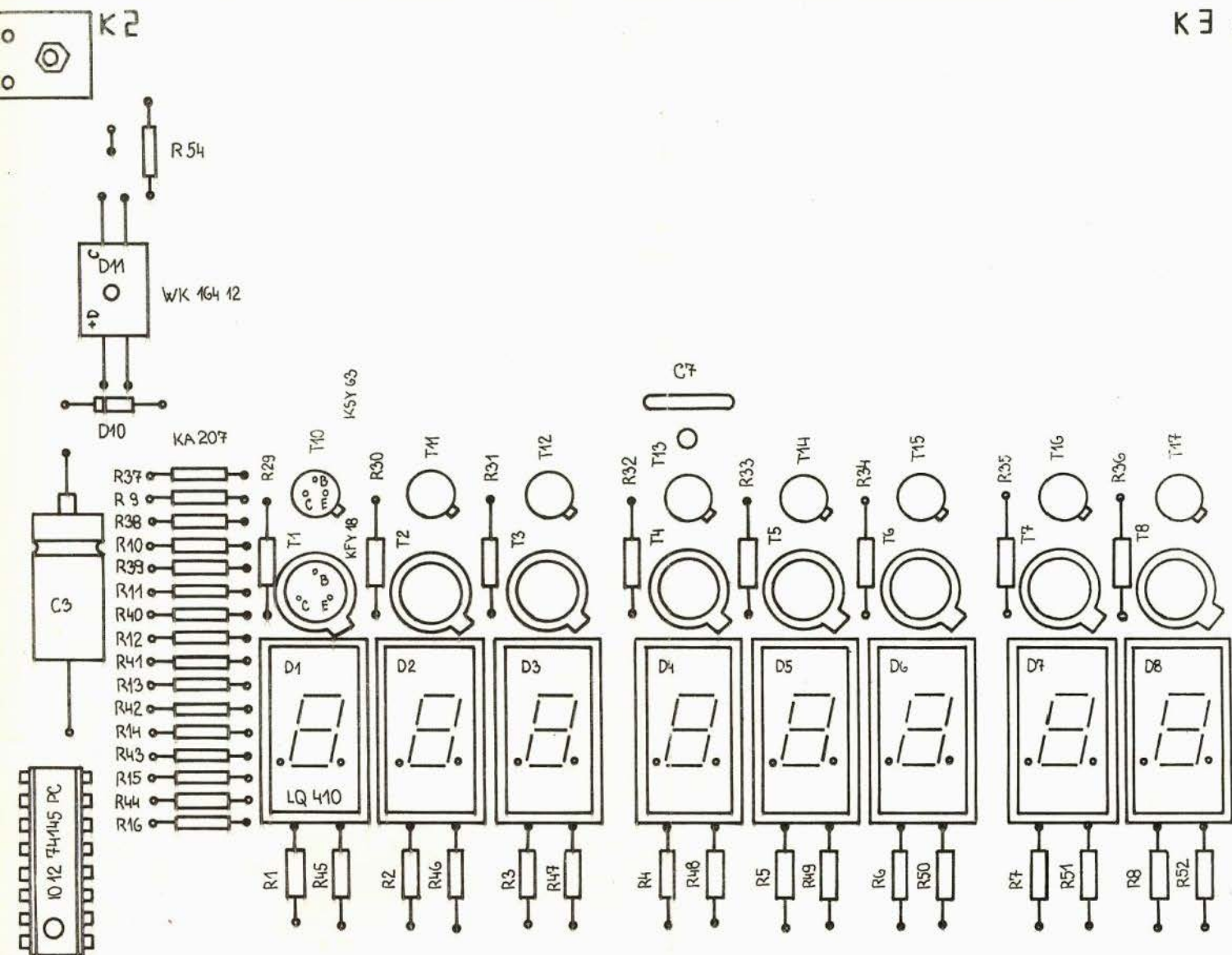


TX 512 2012

K1

+5V





GO	ECH
PM	RG NB
DM BR	CL PR
NXT	END

C	D	E	F
8	9	A	B
4	5	6	7
0	1	2	3

TESLA VRÚSE K.Ú.D. SB-8035<sup>TM</sup> 1985

LOC	OBU	LINE	SOURCE STATEMENT
		1	*TITLE(' MONITOR SB-8035 V-04 2.JANUARA 1984')
		2	;
		3	;
		4	;
		5	;
		6	;
		7	DEKLAROVANIE KONSTANT A PORTOV
		8	ADRHI EQU 07H ;ADRESA URCUJUCA ZACIATOK OBLASTI PA-
0007		9	ADRLO EQU 0BFH ;MATI PROG. RAM PRE ODKLADANIE REG
00BF		10	BRKBUF EQU 22H
0022		11	CHARNO EQU 08H ;POCET DISPLEJOV
0008		12	EMALO EQU 32H ;KONECNA
0032		13	EMAH1 EQU 33H ;ADRESA
0033		14	KEYMOD EQU 1FH
001F		15	KEYGO EQU 1EH
001E		16	KEYPM EQU 1AH
001A		17	KEYDM EQU 16H
0016		18	KEYEND EQU 13H ;KOD TLACITKA END
0013		19	KEYNXT EQU 12H ;KOD TLACITKA NEXT
0012		20	KEYCLR EQU 17H ;KOD TLACITKA CLR
0017		21	KEYLOC EQU 27H ;POZICIA TLACITKA V MATICI
0027		22	KEYREG EQU 1BH
001B		23	KBDBUF EQU 26H ;KOD TLACITKA
0026		24	LFEDM EQU LFEPM
022C		25	LSTDM EQU LSTPM
023F		26	MEMLO EQU 34H ;PARAMETER BLOKOVEHO PRENOSU
0034		27	MEMHI EQU 35H
0035		28	NREPTS EQU 20H ;POCET OPAKOVANI
0020		29	NOVALS EQU 0EH ;POCET PRENASANYCH HODNOT
000E		30	NUMCON EQU 38H ;POCET KONSTANT
0038		31	NCOLS EQU 04H ;POCET STLPCOV KTORE SU NEOTESTOVANE
0004		32	NEXTPL EQU 25H ;POZICIA DISPLEJA
0025		33	OPTION EQU 24H
0024		34	PINPUT EQU P4 ;VSTUP KLAVESNICE
000B		35	PSEGLO EQU P5 ;VYSTUP NA DISPLEJE
000C		36	PSECHI EQU P6
000D		37	PDIGIT EQU P7
000E		38	RUN EQU 7D4H ;START. ADR. KAZDEHO UZIVAT. PROG.
07D4		39	SAVE EQU 21H ;ODKLADANIE PSW
0021		40	SMAHI EQU 31H ;POCIATOCNA ADR. V KAZDOM PRIKAZE
0031		41	SMALO EQU 30H
0030		42	STRTMP EQU 23H
0023		43	SEGMAP EQU 28H ;POCIATOCNA ADR. REG. DISPLEJA
0028		44	USRINT EQU 403H ;ADR. UZIVATELSKEHO EXT. PRERUSENIA
0403		45	USRTIM EQU 407H ;ADR. UZIVAT. PRERUSENIA OD CASOVACA
0407		46	ZERO EQU 00H
0000		47	;
		48	;
		49	DEKLAROVANIE REGISTROV
		50	;
		51	REG. BANKA 0
		52	;
0002		53	LDATA SET R2
0003		54	KEY SET R3
0004		55	ITMP SET R4

LOC	OBJ	LINE	SOURCE STATEMENT
0005		55	BCODE SET R5
0006		56	DSPTMP SET R6
0007		57	TYPE SET R7
		58 ;	
		59 ;	REG. BANKA 1
		60 ;	
0002		61	ROTPAT SET R2
0003		62	ROTCNT SET R3
0004		63	LASTKY SET R4
0005		64	CURDIG SET R5
0006		65	KEYFLG SET R6
0007		66	ASAVE SET R7
		67	*EJECT

LOC OBJ LINE SOURCE STATEMENT

```

68 ;
69 ; KODOVANIE TLACITOK
70 ;
71 ; +---+---+---+---+ +---+---+---+---+
72 ; |1C|1D|1E|1F| |0C|0D|0E|0F|
73 ; +---+---+---+---+ +---+---+---+---+
74 ; |18|19|1A|1B| |08|09|0A|0B|
75 ; +---+---+---+---+ +---+---+---+---+
76 ; |14|15|16|17| |04|05|06|07|
77 ; +---+---+---+---+ +---+---+---+---+
78 ; |10|11|12|13| |00|01|02|03|
79 ; +---+---+---+---+ +---+---+---+---+
    
```

```

80 ;
81 ;
82 ;
83 ; POPIS TLACITOK
84 ;
    
```

85 ;	+---+---+---+---+				+---+---+---+---+			
86 ;								
87 ;	X	X	GO	EXAM/CHA	C	D	E	F
88 ;								
89 ;	+---+---+---+---+				+---+---+---+---+			
90 ;				REGISTER				
91 ;	X	X	PROG. MEM	-----	8	9	A	B
92 ;				NO BRK				
93 ;	+---+---+---+---+				+---+---+---+---+			
94 ;			DATA MEM					
95 ;	X	X	-----	CLR/PREV	4	5	6	7
96 ;			WITH BRK					
97 ;	+---+---+---+---+				+---+---+---+---+			
98 ;								
99 ;	X	X	NEXT/	END/	0	1	2	3
100 ;								
101 ;	+---+---+---+---+				+---+---+---+---+			
102 ;								

```

103 ; X=NEOBSADENE TLACITKO
104 *EJECT
    
```

LOC	OBJ	LINE	SOURCE STATEMENT
105			;
106			ULOZENIE KONSTANT V PAMATI RAM
107			;
108			;ADRESA
109			001A DB 00 ROTPAT (R2)
110			001B DB 00 ROTCNT (R3)
111			001C DB 00 LASTKY (R4)
112			001D DB 08 CURDIG (R5)
113			001E DB 00 KEYFLG (R6)
114			001F DB 00 ASAVE (R7)
115			0020 DB 04 NREPTS
116			0021 DB 08 SAVE
117			0022 DB FF BRKBUF
118			0023 DB 00 STRTMP
119			0024 DB 00 OPTION
120			0025 DB 08 NEXTPL
121			0026 DB FF KBDBUF
122			0027 DB 00 KEYLOC
123			0028 DB XX SEGMAP
124			0029 DB XX SEGMAP+1
125			002A DB XX +2
126			002B DB XX +3
127			002C DB XX +4
128			002D DB XX +5
129			002E DB XX +6
130			002F DB XX +7
131			0030 DB XX SMALO
132			0031 DB XX SMAHI
133			0032 DB XX EMALO
134			0033 DB XX EMAHI
135			0034 DB XX MEMLO
136			0035 DB XX MEMHI
137			*EJECT

REGISTRE 7-SEGMENTOVYCH  
 DISPLEJOV

ADRESA ZOBRAZENA  
 NA DISPLEJI

LOC	OBJ	LINE	SOURCE STATEMENT
138	/		OBSADENIE PORTOV 8035
139	/		
140	/		!P27!P26!P25!P24 ! P23!P22!P21!P20!
141	/		+-----+MONITOR EPROM 2716
142	/		! 0   0   X   X ! 0   0   X   X !1.KBYTE
143	/		+-----+PAMET PROGRAMU 2114
144	/		! 0   0   X   X ! 0   1   X   X !2.KBYTE
145	/		+-----+EXTERNA PAMET DAT 8155
146	/		! 0   0   X   X ! 1   0   X   X !256*8 BIT 3.KBYTE
147	/		+-----+UZIVATELSKA OBLAST EPROM
148	/		! 0   0   X   X ! 1   1   X   X !2716 4.KBYTE PAM.PROG.
149	/		+-----+PAMETOVA ADRESACIA
150	/		! 0   1   X   X ! 1   0   X   X !PORTOV ABC A TIMERU
151	/		+-----+OBVODU 8155
152	/		! ! ! ! !
153	/		! ! ! INEVYUZITE !MENIACE SA BITY A8 A A9 ADRESY
154	/		! ! ! INEVYUZITE
155	/		! !PREPINANIE IO/MN
156	/		!SPINANIE CSN OBVODU 8243
157	/		*EJECT

1 0 0-4 8-1 0 0 0  
 1 0 1-5 9-1 0 0 1  
 1 1 0-6 A-1 0 1 0  
 1 1 1-7 B-1 0 1 1



LOC	OBJ	LINE	SOURCE STATEMENT
		158	*****
		159	*****
		160	;INIT :PROGRAM PRE INICIALIZACIU SYSTEMU,OBVODOV KLAVESNICE
		161	;----- A DISPLEJA.DEFINUJE EXT. PRERUSENIE NA ADRESU 0403H
		162	; A PRERUSENIE OD CASOVACA NA ADR. 0407H.VKLADA POCIA-
		163	; TOCNE HODNOTY DO REGISTROV A NIEKTORYCH BYTE INT. RAM.
		164	;
0000	C5	165	INIT: SEL R00
0001	0409	166	JMP INIT1
0003	8403	167	JMP USRINT
0005	6499	168	GOBRXX: JMP GOBRKX
0007	8407	169	JMP USRTIM
0009	9A33	170	INIT1: ANL P2,#33H
000B	27	171	CLR A
000C	3D	172	MOVD PSEGLO,A
000D	3E	173	MOVD PSEGL1,A
000E	B81A	174	MOV R0,#1AH
0010	B90E	175	MOV R1,#LOW NOVALS
0012	BA00	176	MOV R2,#LOW INVALS
0014	FA	177	INITLP: MOV A,R2
0015	E3	178	MOV3 A,@A
0016	A0	179	MOV @R0,A
0017	10	180	INC @R0
0018	1A	181	INC R2
0019	E914	182	DJNZ R1,INITLP
001B	55	183	STRT T
		184	*****
		185	;MAIN :HLAVNY MONITOROVSKY PROGRAM.ZABEZPECUJE VYPIS SPRAV NA
		186	;----- DISPLEJ.TESTUJE KODY STLACENYCH TLACITOK A ROZLISUJE
		187	; PODLA NICH RIADIACI PRIKAZ.NASTAVUJE SLOVA BC0DE,TYPE,
		188	; OPTION,NUMCON POTREBNE PRE IDENTIFIKACIU PRIKAZU A JE-
		189	; HO PARAMETROV.JEHO VYSTUPOM JE ODOVZDANE RIADENIE PRO-
		190	; GRAMU EXAMIN ALEBO GO.
		191	;
		192	; VYPIS SPRAVU 'SB-8035'
		193	;
001C	2301	194	MAIN: MOV A,#01H
001E	3400	195	CALL OUTUTL
0020	3435	196	MAIN4: CALL INPKEY
		197	;
		198	; AK BOLO STLACENE <END>,CAKAJ DALSIE TLACITKO
		199	;
0022	FB	200	MAIN2: MOV A,KEY
0023	D313	201	XRL A,#KEYEND
0025	C61C	202	JZ MAIN
0027	FB	203	MOV A,KEY
		204	;
		205	; POKRACOVANIE PRE FUNKCIU GOBRK
		206	;
0028	D312	207	XRL A,#KEYNXT
002A	9635	208	JNZ MAIN3
002C	B822	209	MOV R0,#BRKBUF
002E	F0	210	MOV A,@R0
		211	;
		212	; AK ESTE NEBOL ODSTARTOVANY PROGRAM - SKOK DO MONITORA

LOC	OBJ	LINE	SOURCE STATEMENT
		213 ;	INAK VYKONAJ PROGRAM PO DALSI BREAK
		214 ;	
002F	D3FF	215	XRL A,#OFFH
0031	C61C	216	JZ MAIN
0033	0405	217	JMP G0BRXX
0035	BC2C	218 MAIN3:	MOV ITMP,#LOW CTAB
		219 ;	
		220 ;	INICIALIZUJ IDENTIFIKATORY PRIKAZU
		221 ;	
0037	BD00	222	MOV BCODE,#ZERO
0039	BF00	223	MOV TYPE,#ZERO
		224 ;	
		225 ;	ZISTI PODLA STLACENEHO FUNKCNEHO TLACITKA TYP PRIKAZU,
		226 ;	PO STLACENI CHYBNEHO TLACITKA VYPIS CHYBU
		227 ;	
003B	FC	228 FINDOP:	MOV A,ITMP
003C	E3	229	MOVP3 A,@A
003D	B289	230	JBS MERROR
003F	DB	231	XRL A,KEY
0040	C649	232	JZ MAINA
0042	FC	233	MOV A,ITMP
0043	0302	234	ADD A,#02H
0045	AC	235	MOV ITMP,A
0046	1D	236	INC BCODE
0047	043B	237	JMP FINDOP
		238 ;	
		239 ;	NAPIS PRISLUSNU SPRAVU DO REG. DISPLEJA
		240 ;	
0049	FD	241 MAINA:	MOV A,BCODE
004A	0334	242	ADD A,#LOW STRCOM
004C	3402	243	CALL OUTCLR
004E	1C	244	INC ITMP
004F	FC	245	MOV A,ITMP
0050	E3	246	MOVP3 A,@A
0051	B924	247	MOV R1,#OPTION
0053	A1	248	MOV @R1,A
0054	B906	249	MOV R1,#06H
		250 ;	
		251 ;	NULUJ VSTUPNE PARAMETRE PRIKAZU
		252 ;	
0056	B830	253	MOV R0,#SMALO
0058	B000	254 MAINB:	MOV @R0,#00H
005A	18	255	INC R0
005B	E958	256	DJNZ R1,MAINB
005D	3435	257	CALL INPKEY
005F	B924	258	MOV R1,#OPTION
0061	F1	259	MOV A,@R1
0062	AC	260	MOV ITMP,A
0063	1C	261	INC ITMP
0064	FC	262 MAINC1:	MOV A,ITMP
		263 ;	
		264 ;	TESTUJ MODIFIKATOR PRIKAZU
		265 ;	
0065	E3	266	MOVP3 A,@A
0066	97	267	CLR C

LOC	OBJ	LINE	SOURCE STATEMENT
0067	F7	268	RLC A
0068	77	269	RR A
0069	DB	270	XRL A,KEY
		271 ;	
		272 ;	AK SUHLASI VYPIS SPRAVU MODIFIKATORA
		273 ;	
006A	C674	274	JZ MAIND
		275 ;	
		276 ;	AK JE KONIEC TABULKY MODIF.-BER NAHR. HODNOTU
		277 ;	
006C	F672	278	JC MAIND1
006E	1F	279	INC TYPE
006F	1C	280	INC ITMP
0070	0464	281	JMP MAINC1
		282 ;	
		283 ;	NAHRADNA HODNOTA MODIF.=0
		284 ;	
0072	BF00	285	MAIND1: MOV TYPE,#ZERO
0074	B924	286	MAIND: MOV R1,#OPTION
		287 ;	
		288 ;	ADRESA SPRAVY PRE DANY TYPE,VYPIS SPRAVY
		289 ;	
0076	F1	290	MOV A,@R1
0077	E3	291	MOVP3 A,BA
0078	6F	292	ADD A,TYPE
0079	3404	293	CALL OUTMSG
007B	3435	294	CALL INPKEY
		295 ;	
		296 ;	ZAPISUJ VKLADANE PARAMETRE POSTUPNE OD
		297 ;	ADRESY SMALO PO MEMHI
		298 ;	
007D	BB30	299	MAINB0: MOV R0,#SMALO
		300 ;	
		301 ;	CITAJ ADRESU A ZOBRAZ V ADRESNOM POLI DISPLEJOV
		302 ;	AK BOLO STLACENE FUNKCNE TLACITKO -
		303 ;	SKOK NA OBSLUHU PRIKAZU
		304 ;	
007F	34B0	305	CALL INPADR
		306 ;	*****
		307 ;	CMDINT:ROZHODOVACI PROGRAM,KTORY URUCJE ADRESU OBSLUZNEHO
		308 ;	-----PROGRAMU PODLA DRUHU PRIKAZU.
		309 ;	
0081	FD	310	CMDINT: MOV A,BCODE
		311 ;	
		312 ;	ZMENA OBSAHU PAMATI A REGISTROV
		313 ;	
0082	C68D	314	JZ EXAMIN
0084	FF	315	MOV A,TYPE
		316 ;	
		317 ;	START PROGRAMU BEZ LADIACICH BODOV
		318 ;	
0085	C6DF	319	JZ GONBRK
		320 ;	
		321 ;	START PROGRAMU SO ZASTAVENIM V LADIACOM BODE
		322 ;	

LOC	OBJ	LINE	SOURCE STATEMENT
0087	6476	323	JMP GOBRK
		324	;*****
		325	;MERROR:VYPIS SPRAVY 'ERROR-1'--CHYBNE PRIKAZOVE TLACITKO
		326	;-----
		327	;
0089	BA01	328	MERROR: MOV LDATA,#01H
008B	440B	329	JMP PERROR
		330	;*****
		331	;EXAMIN:RIADIACI PROGRAM PRE MODIFIKACIU OBSAHU PAMATI A
		332	;----- REGISTROV.MA TRI MODIFIKATORY:
		333	; TYPE=00 =>PR-MODIFIKACIA OBSAHU PAMATI
		334	; TYPE=01 =>DA-MODIFIKACIA OBSAHU DAT
		335	; TYPE=02 =>RC-MODIFIKACIA OBSAHU REGISTROV A INTERNEJ
		336	; PAMATI DAT
		337	; STLACENIM TLACITKA NEXT SA PRESUVA NA DALSIU VYSSIU
		338	; ADRESU.
		339	; STLACENIM TLACITKA CLR/PREV SA PRESUVA NA ADRESU 0 1
		340	; NIZSIU.
		341	; STLACENIM TLACITKA END SA PRIKAZ UKONCUJE A VRATI
		342	; SA RIADENIE DO PROGRAMU MAIN
		343	;
008D	85	344	EXAMIN: CLR F0
		345	;
		346	; URCI TYP PAMATI
		347	;
008E	FF	348	EXAM0: MOV A,TYPE
008F	D302	349	XRL A,#02H
0091	969C	350	JNZ EXAMR
0093	B831	351	MOV R0,#SMAHI
0095	B000	352	MOV @R0,#00H
0097	C8	353	DEC R0
0098	F0	354	MOV A,@R0
0099	533F	355	ANL A,#3FH
009B	A0	356	MOV @R0,A
009C	FF	357	EXAMR: MOV A,TYPE
009D	0336	358	ADD A,#LOW STRMEM
009F	3402	359	CALL OUTCLR
00A1	B831	360	MOV R0,#SMALO+1
		361	;
		362	; ZOBRAZ ADRESU PAMATI
		363	;
00A3	34D8	364	CALL UPDAD1
00A5	2340	365	MOV A,#40H
00A7	349B	366	CALL WDISP
		367	;
		368	; PRENES OBSAH ADRESY PAMATI DO R2
		369	;
00A9	5425	370	CALL LFETCH
00AB	FA	371	MOV A,LDATA
00AC	47	372	SWAP A
00AD	3496	373	CALL DSPACC
00AF	FA	374	MOV A,LDATA
		375	;
		376	; ZAPIS DATA DO DAT POLA DISPLEJA A ZOBRAZ
		377	;

LOC	OBJ	LINE	SOURCE STATEMENT
00B0	3496	378	CALL DSPACC
00B2	3435	379	CALL INPKEY
00B4	FB	380	MOV A,KEY
		381 ;	
		382 ;	AK BOLO STLACENE FUNKCNE TLACITKO,
		383 ;	TESTUJ,CI JE PRIPUSTNE
		384 ;	
00B5	92C5	385	JB4 EXAM1
00B7	FA	386	MOV A,LDATA
00B8	47	387	SWAP A
00B9	53F0	388	ANL A,#0F0H
00BB	B6BF	389	JF0 EXAM5
00BD	27	390	CLR A
00BE	95	391	CPL F0
00BF	6B	392 EXAM5:	ADD A,KEY
00C0	AA	393	MOV LDATA,A
		394 ;	
		395 ;	NOVY UDAJ VLOZ NA POVODNU ADRESU
		396 ;	
00C1	5438	397	CALL LSTORE
00C3	048E	398	JMP EXAM0
		399 ;	
		400 ;	AK BOLO STLACENE TLACITKO <END>-KONIEC PRIKAZU
		401 ;	EXAMINE,INAC MODIFIKUJ DALSIE DATA
		402 ;	
00C5	D313	403 EXAM1:	XRL A,#KEYEND
00C7	C61C	404	JZ MAIN
00C9	FB	405 EXAM2:	MOV A,KEY
		406 ;	
		407 ;	AK BOLO STLACENE <NEXT>,ZVYS ADRESU
		408 ;	
00CA	D312	409	XRL A,#KEYNXT
00CC	96D2	410	JNZ EXAM3
00CE	5400	411	CALL INCSMA
00D0	048D	412	JMP EXAMIN
00D2	FB	413 EXAM3:	MOV A,KEY
		414 ;	
		415 ;	AK BOLO STLACENE CLR/PREV,ZNIZ ADRESU
		416 ;	
00D3	D317	417	XRL A,#KEYCLR
00D5	96DB	418	JNZ EXAM4
00D7	34F4	419	CALL DECSMA
00D9	048D	420	JMP EXAMIN
		421 ;	
		422 ;	AK BOLO STLACENE INE TLACITKO,VYPIS CHYBU
		423 ;	'ERROR 3'
		424 ;	
00DB	BA03	425 EXAM4:	MOV LDATA,#03H
00DD	440B	426	JMP PERROR
		427 ;	
		428 ;	OBSLUHA FUNKCIE GO NOBREAK
		429 ;	
00DF	545E	430 GONBRK:	CALL STATUS
00E1	65	431	STOP TCNT
00E2	E4D4	432	JMP RUN

LOC	OBJ	LINE	SOURCE STATEMENT
		433	*****
		434	;BRKMOV:PODPROGRAM PRE PRESUN 2 BYTOV OPERACNEHO KODU Z ADRESY
		435	;----- BREAK A BREAK+1 NA ADRESU BUFF A BUFF+1 A ZAPIS IN-
		436	; STRUKCIE JMP RST2 NA ADRESU BREAK A BREAK+1. ADRESA
		437	; BREAK MUSI BYT ULOZENA V EMA.
		438	;
0044		439	JMPCD EQU 44H
07D0		440	BUFF EQU 7D0H
		441	;
		442	; VLOZ DO SMA HODNOTU BUFF, T. J. 7D0H
		443	;
00E4	B831	444	BRKMOV: MOV R0, #SMAHI
00E6	B007	445	MOV @R0, #HIGH BUFF
00E8	C8	446	DEC R0
00E9	B0D0	447	MOV @R0, #LOW BUFF
		448	;
		449	; PRENES 1 BYTE OPERACNEHO KODU DO BUFF A OP. KOD
		450	; INSTRUKCIE JMP RST2 NA ADRESU BREAK
		451	;
00EB	54ED	452	CALL BUFSTR
00ED	BA44	453	MOV R2, #JMPCD
00EF	74B6	454	CALL BRBAC1
		455	;
		456	; PRESUN NA ADRESY BUFF+1 A BREAK+1
		457	;
00F1	B830	458	MOV R0, #SMALO
00F3	10	459	INC @R0
00F4	B832	460	MOV R0, #EMALO
00F6	10	461	INC @R0
		462	;
		463	; PRENES 2. BYTE OP. KODU DO BUFF+1
		464	;
00F7	54ED	465	CALL BUFSTR
		466	;
		467	; PRENES 2. BYTE OP. KODU INSTRUKCIE JMP RST2
		468	; NA ADRESU BREAK+1
		469	;
00F9	BAC5	470	MOV R2, #LOW RST2
00FB	64B6	471	JMP BRBAC1
		472	*EJECT

LOC	OBJ	LINE	SOURCE STATEMENT
0100		473	ORG 100H
		474	*****
		475	OUTUTL:PODPROGRAM ZAPISUJE DO REGISTRA DISPLEJOV SPRAVU
		476	----- 'SB-8035' ALEBO 'ERROR X' PODLA HODNOTY V ACC
		477	;
0100	0331	478	OUTUTL: ADD A,#LOW STRUTL
		479	*****
		480	OUTCLR:PODPROGRAM ZAPLNI REGISTRE DISPLEJA 00H(MEDZERY) A
		481	----- ZAPISE DO NICH SPRAVU Z ADRESY,NA KTORU UKAZUJE OBSAH
		482	ACC NA STRANKE 3.
0102	3427	483	OUTCLR: CALL CLEAR
		484	*****
		485	OUTMSG:PODPROGRAM KOPIRUJE RETAZEC Z PAMETI ROM DO REGISTRA
		486	----- DISPLEJOV PODLA OBSAHU ACC.ZACIATOK RETAZCA,KTORÝ SA
		487	KOPIRUJE JE URCENÝ OBSAHOM ACC,KONIEC RETAZCA JE URCE-
		488	NY NASTAVENIM BITU D7 DO 1 U KOPIROVANEHO ZNAKU.POD-
		489	PROGRAM VZDY ZACINA ZOBRAZOVAT OD LAVEHO DISPLEJA PO-
		490	KIAL UZIVATEL PRED JEHO VOLANIM NEDEFINUJE V PARAMETRI
		491	NEXTPL INY REGISTER DISPLEJA AKO PRVY.
		492	;
		493	VSTUPY :ACC-OBSAHUJE ADRESU 1.ZNAKU RETAZCA ULOZENEHO
		494	NA STRANKE P3.
		495	VYSTUPY:DO REGISTROV DISPLEJA NA ADRESU SEGMAP AZ SEG-
		496	MAP+7 SA PREPISE SPRAVA Z ROM.
		497	NICI :REG. ACC A R1
		498	VOLANIA:WDISP
		499	;
0104	E3	500	OUTMSG: MOVP3 A,@A
0105	B923	501	MOV R1,#STRTMP
0107	A1	502	MOV @R1,A
0108	B923	503	PRNT2: MOV R1,#STRTMP
010A	F1	504	MOV A,@R1
010B	E3	505	MOVP3 A,@A
010C	F217	506	JB7 PRNT1
010E	349B	507	CALL WDISP
0110	B923	508	MOV R1,#STRTMP
0112	F1	509	MOV A,@R1
0113	17	510	INC A
0114	A1	511	MOV @R1,A
0115	2408	512	JMP PRNT2
0117	537F	513	PRNT1: ANL A,#07FH
0119	249B	514	JMP WDISP
		515	*****
		516	RST1 :PROGRAM PRE RESTART MONITORA.ZABEZPECUJE PREPIS UZI-
		517	----- VATELSKYCH REGISTROV A INTERNEJ RAM DO PROGRAMOVEJ
		518	RAM OD ADRESY 07C0H DO 07FFH S NAVRATOM DO HLAVNEHO
		519	MONITOROVSKEHO PROGRAMU CEZ PROGRAM INIT.NA TENTO
		520	PROGRAM TERBA VZDY VYKONAT SKOK,AK CHCE UZIVATEL ZA-
		521	CHOVAT OBSAH REGISTROV A INTERNEJ RAM.
		522	;
011B	545E	523	RST1: CALL STATUS
011D	B822	524	MOV R0,#BRKBUF
011F	F0	525	MOV A,@R0
0120	37	526	CPL A
0121	C625	527	JZ RST1A

LOC	OBJ	LINE	SOURCE STATEMENT
0123	74A3	528	CALL BRBACK
0125	0400	529	RST1A: JMP INIT
		530	;*****
		531	;CLEAR:PODPROGRAM PRE VYPRAZDNIENIE REGISTROV DISPLEJA (NAPL-
		532	; NENIE HODNOTAMI 00H).ADRESA PRVEHO REGISTRA,DO KTOREHO
		533	; SA ZAPISUJE MEDZERA,URCUJE NEXTPL(V OBRATENOM PORADI),
		534	; T.J. PRE PRVY REGISTER ZLAVA NEXTPL=8,PRE PRVY ZPRAVA
		535	; NEXTPL=1
		536	;
		537	; VSTUPY: NEXTPL OBSAHUJE ADRESU PRVEHO NULOVANEHO
		538	; REGISTRA (V OPACNOM SMERE)
		539	; VYSTUP: NULUJU SA VSETKY REGISTRE OD ZADANEHO SME-
		540	; ROM VLAVO.NEXTPL SA NASTAVI NA 8.
		541	; NICI:R0,R1,SEGMAP AZ SEGMAP+7,NEXTPL
		542	; VOLANIA:ZIADNE
		543	;
0127	B82F	544	CLEAR: MOV R0,#SEGMAP+7
0129	B908	545	MOV R1,#CHARNO
012B	B000	546	DBLANK: MOV @R0,#00H
012D	C8	547	DEC R0
012E	E92B	548	DJNZ R1,DBLANK
0130	B925	549	MOV R1,#NEXTPL
0132	B108	550	MOV @R1,#CHARNO
0134	83	551	RET
		552	;*****
		553	;INPKEY:HLAVNY PROGRAM PRE REFRES DISPLEJA A SNIMANIE KLAVES-
		554	;-----NICE,KTORY VYUZIVA INTERNY CASOVAC NA GENEROVANIE
		555	; ONESKORENIA PRE POSTUPNE BUDENIE JEDNOTLIVYCH SEGMENTOV
		556	; VSETKYCH 8-ICH DISPLEJOV.
		557	;
		558	; VSTUPY:STLACENE TLACITKO V MATICI
		559	; VYSTUP:ACC A R3 OBSAHUJU KOD STLACENEHO TLACITKA.OZI-
		560	; VI SA JEDEN KRAT CELA OSMICA DISPLEJOV AK
		561	; DOSLO K PRETECENIU CITACA.
		562	; NICI:ACC,R1,R3,KBDBUF
		563	; VOLANIA:TOFPOL
		564	;
0135	3443	565	INPKEY: CALL TOFPOL
		566	;
		567	; AK NEBOLO STLACENE TLACITKO,CAKAJ V SLUCKE
		568	;
0137	B926	569	MOV R1,#KBDBUF
0139	F1	570	MOV A,@R1
013A	F235	571	JB7 INPKEY
013C	27	572	CLR A
		573	;
		574	; AK ANO,ODPOJ VYSTUPY EXPANDERA NA KLAVESNICU
		575	;
013D	3E	576	MOVD PSEghi,A
013E	3D	577	MOVD PSEGLO,A
013F	37	578	CPL A
		579	;
		580	; PREPIS KOD TLACITKA DO ACC A R3
		581	;
0140	21	582	XCH A,@R1



LOC	OBJ	LINE	SOURCE STATEMENT
0141	AB	583	MOV KEY,A
0142	83	584	RET
		585 ;	
		586 ;	VOLANIE REFRESOVACIEHO PROGRAMU V PRIPADE,ZE DOSLO
		587 ;	K UPLYNUTIU VYMEDZENEO CASU SVIETIVOSTI DISPLEJOV
		588 ;	
0143	1646	589	TOFPOL: JTF TIINT
0145	83	590	RET
		591 ;	*****
		592 ;	TIINT: PROGRAM PRE OBSLUHU DISPLEJA A KLAVESNICE. JE VOLANY PE-
		593 ;	----- RIADICKY VZDY, KED TREBA ROZSVIETIT DISPLEJE ALEBO SNI-
		594 ;	MAT MATICU TLACITOK. TENTO PROGRAM VYKONAVA DYNAMICKY
		595 ;	REFRES DISPLEJOV CYKLICKY Priblizne RAZ ZA 1MS 1 DISP.
		596 ;	
		597 ;	VSTUPY :ZIADNE
		598 ;	VYSTUP :REFRESUJE SA JEDEN DISPLEJ, SNIMA SA JEDEN RIA-
		599 ;	DOK TLACIDIEL (CELKOVE JE 8 RIADKOV PO 4 STL-
		600 ;	PCE I KED ZDANLIVO JE TO NAOPAK)
		601 ;	NICI :R0 AZ R6 V BANKE RB1,ASAVE,KEYLOC,NREPTS,
		602 ;	KBDBUF,RDELAY
		603 ;	VOLANIA:ZIADNE
		604 ;	
0146	D5	605	TIINT: SEL RB1
0147	AF	606	MOV ASAVE,A
0148	23F8	607	MOV A,#0F8H
014A	62	608	MOV T,A
014B	27	609	CLR A
014C	3E	610	MOVD PSEGI,A
014D	3D	611	MOVD PSEGLO,A
014E	FD	612	MOV A,CURDIG
014F	07	613	DEC A
0150	3F	614	MOVD PDIGIT,A
0151	0C	615	MOVD A,PINPUT
0152	AA	616	MOV ROTPAT,A
0153	FD	617	MOV A,CURDIG
0154	07	618	DEC A
0155	0328	619	ADD A,#SEGMAP
0157	A8	620	MOV R0,A
0158	F0	621	MOV A,@R0
0159	3D	622	MOVD PSEGLO,A
015A	47	623	SWAP A
		624 ;	
		625 ;	ZOBRAZUJE SA DALSI ZNAK,ZAROVEN SO SNIMANIM
		626 ;	RIADKU KLAVESNICE JE REFRESOVANY KOD SEGMENTOV
		627 ;	DISPLEJOV,TESTUJE SA,CI BOLO STLACENE TLACITKO
		628 ;	
015B	3E	629	MOVD PSEGI,A
015C	BB04	630	MOV ROTCNT,#NCOLS
015E	FA	631	NXTLOC: MOV A,ROTPAT
015F	67	632	RRC A
0160	AA	633	MOV ROTPAT,A
0161	F67F	634	JC SCAN5
		635 ;	
		636 ;	BOLO DETEKOVANE STLACENE TLACITKO V BEZNOM STLPICI.
		637 ;	JEHO KOD JE V REGISTRU KEYLOC.KONTROLUJE SA CI AJ

LOC	OBJ	LINE	SOURCE STATEMENT
		638 ;	PRI DAL SOM CITANI BOLO STLACENE TO ISTE TLACIDLO
		639 ;	
0163	BE01	640	MOV KEYFLG,#01H
0165	B927	641	MOV R1,#KEYLOC
0167	F1	642	MOV A,@R1
0168	2C	643	XCH A, LASTKY
0169	DC	644	XRL A, LASTKY
		645 ;	
		646 ;	BOLO ZISTENE STLACENIE INEHO TLACITKA V TOMTO
		647 ;	CYKLE NEZ V PREDCHADZAJUCOM->TESTUJ ZNOVA
		648 ;	DALSIE STLACENIE
		649 ;	
016A	C672	650	JZ SCAN3
016C	B920	651	MOV R1,#NREPTS
016E	B106	652	MOV @R1,#06H
		653 ;	
		654 ;	BOLO DETEKOVANE TO ISTE TLACITKO AKO V PRED-
		655 ;	CHADZAJUCOM CYKLE .AK UZ BOLO NACITANE 5 ALEBO
		656 ;	6 X,ZAPIS HO DO REGISTRA LASTKY A KBDBUF,INAC
		657 ;	TESTUJ ZNOVA
		658 ;	
0170	247F	659	JMP SCAN5
0172	B920	660	SCAN3: MOV R1,#NREPTS
0174	F1	661	MOV A,@R1
0175	C67F	662	JZ SCAN5
0177	07	663	DEC A
0178	A1	664	MOV @R1,A
0179	967F	665	JNZ SCAN5
017B	FC	666	MOV A, LASTKY
017C	B926	667	MOV R1,#KBDBUF
017E	A1	668	MOV @R1,A
		669 ;	
		670 ;	KEYLOC VYTVARA KOD TLACITKA <=> JEHO POZICII
		671 ;	
017F	B927	672	SCAN5: MOV R1,#KEYLOC
0181	11	673	INC @R1
		674 ;	
		675 ;	POSTUPNY PRECHOD CEZ VSETKY RIADKY A STLPCE
		676 ;	
0182	EB5E	677	DJNZ ROTCNT,NXTLOC
0184	ED93	678	DJNZ CURDIG,TIRET1
0186	BD08	679	MOV CURDIG,#CHARNO
		680 ;	
		681 ;	SPATNA INICIALIZACIA REGISTROV KEYLOC,KEYFLG,
		682 ;	LASTKY PRE DALSIE VOLANIE PROGRAMU TIINT.
		683 ;	
0188	B927	684	MOV R1,#KEYLOC
018A	B100	685	MOV @R1,#00H
018C	FE	686	MOV A,KEYFLG
018D	9691	687	JNZ SCAN8
018F	BCFF	688	MOV LASTKY,#0FFH
0191	BE00	689	SCAN8: MOV KEYFLG,#00H
		690 ;	
		691 ;	OBNOV ACC
		692 ;	

LOC	OBJ	LINE	SOURCE STATEMENT
0193	FF	693	TIRET1: MOV A,ASAVE
0194	C5	694	SEL R00
0195	83	695	RET
		696	*****
		697	;DSPACC:ZOBRAZUJE HEXAZNAK,KTORY JE V DOLNOM NIBBLE DO BEZ-
		698	;----- NEHO REGISTRA DISPLEJOV.ZNAK SA PREVEDIE NAJSKOR DO
		699	; 7 SEGMENTOVEHO KODU.POZICIU REGISTRA,DO KTOREHO SA
		700	; ZNAK ZAPISE URCLUJE NEXTPL,TENTO SA VZDY PO ZAPISE
		701	; DEKREMENTUJE O 1,T.J. ZAPISUJE SA POSTUPNE DO DIS-
		702	; PLEJOV ZLAVA DOPRAVA.
		703	;
		704	; VSTUPY :ACC OBSAHUJE V BITOCH 0-3 ZNAK 0-FH,KTORY SA
		705	; MA ZOBRAZIT V SEDEMSEGMENTOVOM KODE.
		706	; NEXTPL OBSAHUJE POZICIU DISPLEJA(V OBRATENOM
		707	; PORADI)
		708	; VYSTUPY:ZNAK V ACC SA PREVEDIE DO 7 SEGM. KODU A ZA-
		709	; PISE SA DO PRISLUSNEHO REGISTRA
		710	; NEXTPL SA ZNIZI O 1
		711	; NICI :ACC,R1,R6,SEGMAP AZ SEGMAP+7,NEXTPL
		712	; VOLANIA:ZIADNE
		713	; POZN. :AK SA DOSIAHNE DISPLEJ PRVY ZPRAVA NASTAVI SA
		714	; NEXTPL NA DISPLEJ PRVY ZLAVA.
		715	;
0196	530F	716	DSPACC: ANL A,#0FH
0198	031C	717	ADD A,#LOW DGPATS
019A	E3	718	MOV3 A,@A
		719	*****
		720	;WDISP :ZAPISUJE DO ZVOLENEHO REGISTRA DISPLEJA OBSAH ACC
		721	;----- (7 SEGMENTOVY KOD ZNAKU).
		722	;
		723	; VSTUPY :ACC-7 SEGMENTOVY KOD ZNAKU
		724	; NEXTPL-POZICIA REGISTRA DISPLEJOV,KDE SA ZNAK
		725	; ZAPISE
		726	; VYSTUPY:OBSAH ACC SA ZAPISE DO PRISLUSNEHO REG.DISPLEJA
		727	; NEXTPL SA ZNIZI O 1
		728	; OSTATNE AKO U DSPACC
		729	;
019B	AE	730	WDISP: MOV DSPTMP,A
019C	B925	731	MOV R1,#NEXTPL
019E	F1	732	MOV A,@R1
019F	37	733	CPL A
01A0	17	734	INC A
01A1	0330	735	ADD A,#SEGMAP+8
01A3	A9	736	MOV R1,A
01A4	FE	737	MOV A,DSPTMP
01A5	A1	738	MOV @R1,A
01A6	B925	739	MOV R1,#NEXTPL
01A8	F1	740	MOV A,@R1
01A9	07	741	DEC A
01AA	A1	742	MOV @R1,A
01AB	96AF	743	JNZ WDISP1
01AD	B108	744	MOV @R1,#CHARNO
01AF	83	745	WDISP1: RET
		746	*****
		747	;INPADR:VKLADA UDAJE DO DVOJBYTOVEHO REGISTRA ADRESOVANEHO RE-

LOC	OBJ	LINE	SOURCE STATEMENT
		748	;----- GISTROM R0. OCAKAVA VSTUP CISLICOVYCH KLAVES AZ KYM NIE
		749	; JE STLACENE NEXT ALEBO END. PO ICH STLACENI SA UDAJE
		750	; PREPISU DEFINITIVNE DO PRISLUSNEHO REGISTRA A NOVA ADR.
		751	; SA OBJAVI NA DISPLEJI V ADRESOVOM POLI. AK JE POCET KON-
		752	; STANT VYCERPANY, T. J. NUMCON=0, PROGRAM SA NEVYKONA.
		753	;
		754	; VSTUPY ;R0 OBSAHUJE ADRESU INTERNEJ RAM, KAM SA ULOZI
		755	; DOLNY BYTE ADRESY(DAT)
		756	; VYSTUPY: PO VLOZENI PRISLUSNEJ ADRESY A STLACENI NEXT,
		757	; RESP. END SA VLOZI ADRESA DO PRISLUSNEHO DVOJ-
		758	; BYTOVEHO REGISTRA ADRESOVANEHO R0 A ZOBRAZI NA
		759	; NA DISPLEJI V POZICIACH ADRESY(PRVE 3 ZPRAVA)
		760	; NICI ;R0,R1,ACC,R2,R3
		761	; VOLANIA: UPDADR, INPKEY, DBLANK
		762	; POZN. ; V PRIPADE, ZE AKO ODDELOVACI ZNAK NIE JE POU-
		763	; ZITE TLACITKO NEXT ALEBO END, JE VYPISANA SPR-
		764	; VA O CHYBE 'ERROR 2'
		765	;
01B0	FB	766	INPADR: MOV A,KEY
01B1	92BF	767	JB4 ELSIF1
01B3	20	768	XCH A,@R0
01B4	47	769	SWAP A
01B5	20	770	XCH A,@R0
01B6	30	771	XCHD A,@R0
01B7	18	772	INC R0
01B8	30	773	XCHD A,@R0
01B9	34D4	774	CALL UPDADR
01BB	3435	775	CALL INPKEY
01BD	24B0	776	JMP INPADR
		777	;
		778	;
01BF	FB	779	ELSIF1: MOV A,KEY
01C0	D312	780	XRL A,#KEYNXT
01C2	C6CD	781	JZ ELSIF2
01C4	FB	782	MOV A,KEY
01C5	D313	783	XRL A,#KEYEND
01C7	C6CD	784	JZ ELSIF2
01C9	BA02	785	MOV LDATA,#02H
01CB	440B	786	JMP PERROR
01CD	B82F	787	ELSIF2: MOV R0,#SEGMAP+7
01CF	B903	788	MOV R1,#03H
01D1	342B	789	CALL DBLANK
01D3	83	790	RET
		791	;*****
		792	;UPDADR: PREPISE OBSAH 3 CISLIC Z REGISTRA V INTERNEJ RAM,
		793	;----- KTORY JE ADRESOVANY REGISTROM R0-1 A R0 NA POSLED-
		794	; NE 3 REGISTRE DISPLEJA (ZPRAVA), TAK, ZE UDAJ ZAPI-
		795	; SANY V REGISTRI @R0 V BITOCH 0-3 SA PREPISE NA REGISTER
		796	; DISPLEJA 3-TI ZPRAVA, HORNÝ NIBBLE REGISTRA ADRESOVANE-
		797	; HO R0 SA PREPISE NA POZICIU DRUHU ZPRAVA A DOLNY NIBBLE
		798	; NA POZICIU UPLNE VPRAVO - NEVYZNAMNE NULY SA NEZOBRAZU-
		799	; JU (ZOBRAZUJU SA AKO MEDZERY)
		800	;
		801	; VSTUPY ;R0 - OBSAHUJE ADRESU VYSSIEHO REGISTRA ADR.,
		802	; V KTOREJ JE PLATNY LEN DOLNY NIBBLE

LOC	OBJ	LINE	SOURCE STATEMENT
		803 ;	VYSTUP ;HODNOTY @R0 A @R0-1 SA POSTUPNE PREVEDU NA
		804 ;	7 SEGMENTOVY KOD A ZOBRAZIA NA 3 PRAVE DIS-
		805 ;	PLEJE V PORADI ZLAVA DO PRAVA
		806 ;	NICI :R0,ACC,R1,NEXTPL
		807 ;	VOLANIA:WDISP,DSPACC
		808 ;	
01D4	B925	809	UPDADR: MOV R1,#NEXTPL
01D6	B103	810	MOV @R1,#03H
01D8	F0	811	UPDAD1: MOV A,@R0
01D9	C8	812	DEC R0
01DA	530F	813	ANL A,#0FH
01DC	96EA	814	JNZ DSPHI
01DE	349B	815	CALL WDISP
01E0	F0	816	MOV A,@R0
01E1	47	817	SWAP A
01E2	530F	818	ANL A,#0FH
01E4	96EE	819	JNZ DSPM1
01E6	349B	820	CALL WDISP
01E8	24F0	821	JMP DSPLO
01EA	349B	822	DSPHI: CALL DSPACC
01EC	F0	823	DSPMID: MOV A,@R0
01ED	47	824	SWAP A
01EE	349B	825	DSPM1: CALL DSPACC
01F0	F0	826	DSPLO: MOV A,@R0
01F1	349B	827	CALL DSPACC
01F3	83	828	RET
		829 ;	*****
		830 ;	DECSMA:DEKREMENTUJE OBSAH ADRESY SMAHI+SMALO AKO 16 BIT. SLOVA
		831 ;	----- 0 1,OSTATNE AKO U INCSMA,INCW
		832 ;	
01F4	B930	833	DECSMA: MOV R1,#SMALO
		834 ;	*****
		835 ;	DECW :DEKREMENTUJE HODNOTU @R1 A @R1+1 AKO 16 BIT. SLOVA
		836 ;	-----
		837 ;	
01F6	F1	838	DECW: MOV A,@R1
01F7	07	839	DEC A
01F8	21	840	XCH A,@R1
01F9	96FF	841	JNZ DECSM1
01FB	19	842	INC R1
01FC	F1	843	MOV A,@R1
01FD	07	844	DEC A
01FE	31	845	XCHD A,@R1
01FF	83	846	DECSM1: RET
		847	*EJECT

LOC	OBJ	LINE	SOURCE STATEMENT
0200		848	ORG 200H
		849	*****
		850	;INCSMA:INKREMENTUJE OBSAH ADRESY SMALO+SMAHI AKO 16 BITO-
		851	; VEHO REGISTRA 0 1
		852	;
0200	B930	853	INCSMA: MOV R1,#SMALO
		854	*****
		855	;INCW ;INKREMENTUJE OBSAH INTERNEJ RAM ADRESOVANEJ R1 A R1+1
		856	; 0 1
		857	;
		858	; VSTUPY ;R1 - UKAZUJE NA DOLNY BYTE INKR. SLOVA
		859	; VYSTUPY:HODNOTA @R1 A @R1+1 JE 0 1 VYSSIA
		860	; NICI ;R1,ACC,SMALO,SMAHI
		861	;
0202	11	862	INCW: INC @R1
0203	F1	863	MOV A,@R1
0204	9608	864	JNZ INCW1
0206	19	865	INC R1
0207	11	866	INC @R1
0208	83	867	INCW1: RET
		868	;
		869	;
		870	;
0209	BA04	871	ERROR: MOV LDATA,#04H
		872	*****
		873	;PERROR:VYPIS CHYBY, KU KTOREJ DOSLO PRI STLACENI TLACITKA KLA-
		874	;------ VESNICE
		875	; VSTUPY ;R2 - OBSAHUJE KOD CHYBY (0-F)
		876	; VYSTUP ;NA DISPLEJ SA VYPISE ERROR X, KDE X JE KOD
		877	; CHYBY
		878	; POZN. ;CHYBA SA NEVYPISE AK BOLO STLACENE TLAC. CLR
		879	; NICI ;ACC,R1,R2,PSW,R3,KBDBUF
		880	; VOLANIA:OUTUTL,DSPACC,INPKEY
		881	;
020B	27	882	PERROR: CLR A
020C	D7	883	MOV PSW,A
020D	FB	884	MOV A,KEY
020E	D317	885	XRL A,#KEYCLR
0210	C623	886	JZ ERROR2
0212	27	887	CLR A
0213	3400	888	CALL OUTUTL
0215	FA	889	MOV A,LDATA
0216	3496	890	CALL DSPACC
0218	B926	891	MOV R1,#KBDBUF
021A	B1FF	892	MOV @R1,#0FFH
021C	3435	893	CALL INPKEY
021E	FB	894	MOV A,KEY
021F	D313	895	XRL A,#KEYEND
0221	9609	896	JNZ RERROR
0223	041C	897	ERROR2: JMP MAIN
		898	*****
		899	;LFETCH:PROGRAM PRE PRENOS 1 BYTU PROGRAMOVEJ PAMATI RAM DO
		900	;------ ACC A R2
		901	; VSTUPY ;SMAHI-SMALO OBSAHUJU ADRESY PAMATI PROGRAMU
		902	; VYSTUPY:DO R2 SA PRENESIE BYTE DAT ZO SMAHI-SMALO

LOC	OBJ	LINE	SOURCE STATEMENT
		903 ;	Z PAMATI PROGRAMU
		904 ;	NICI :ACC,R0,R2,SMAHI,SMALO
		905 ;	
0225	FF	906	LFETCH: MOV A,TYPE
0226	0329	907	ADD A,#LOW LFETBL
0228	B3	908	JMPP @A
0229	2C	909	LFETBL: DB LOW LFEPM
022A	2C	910	DB LOW LFEDM
022B	4B	911	DB LOW LFEREG
022C	B831	912	LFEPM: MOV R0,#SMAHI
022E	F0	913	MOV A,@R0
022F	C8	914	DEC R0
0230	3A	915	LFEPM1: OUTL P2,A
0231	F0	916	MOV A,@R0
0232	A8	917	MOV R0,A
0233	80	918	MOVX A,@R0
0234	9AF3	919	ANL P2,#0F3H
0236	AA	920	MOV LDATA,A
0237	83	921	RET
		922	*****
		923	;LSTORE: PROGRAM PRE ZAPIS 1 BYTU DAT DG PAMATI PROGRAMU RAM
		924	;
		925	; VSTUPY : SMAHI-SMALO OBSAHUUJ ADRESU PAMATI PROGRAMU
		926	; R2 OBSAHUJE BYTE DAT PRE ZAPIS
		927	; VYSTUPY: SMAHI-SMALO OBSAHUUJ BYTE DAT Z R2
		928	; NICI :ACC,R0,R2,SMAHI,SMALO
		929	;
0238	FF	930	LSTORE: MOV A,TYPE
0239	033C	931	ADD A,#LOW LSTBL
023B	B3	932	JMPP @A
023C	3F	933	LSTBL: DB LOW LSTPM
023D	3F	934	DB LOW LSTDM
023E	4F	935	DB LOW LSTREG
023F	B831	936	LSTPM: MOV R0,#SMAHI
0241	F0	937	MOV A,@R0
0242	C8	938	DEC R0
0243	3A	939	LSTPM1: OUTL P2,A
0244	F0	940	MOV A,@R0
0245	A8	941	MOV R0,A
0246	FA	942	MOV A,LDATA
0247	90	943	MOVX @R0,A
0248	9AF3	944	ANL P2,#0F3H
024A	83	945	RET
		946	*****
		947	;LFEREG: PROGRAM PRE ZMENU OBSAHU REGISTROV A INTERNEJ RAM.
		948	;----- PRACUJE OBDOBNE AKO LFEPM1. POUZIVA NA ODKLADANIE RE-
		949	; GISTROV PAMAT PROGRAMU RAM OD ADRESY 7C0H PO 7FFH.
		950	;
024B	5453	951	LFEREG: CALL LFLST
024D	4430	952	JMP LFEPM1
024F	5453	953	LSTREG: CALL LFLST
0251	4443	954	JMP LSTPM1
0253	B930	955	LFLST: MOV R1,#SMALO
0255	F1	956	MOV A,@R1
0256	03C0	957	ADD A,#ADRLO+1

LOC	OBJ	LINE	SOURCE STATEMENT
0258	B832	958	MOV R0,#EMALO
025A	A0	959	MOV @R0,A
025B	2307	960	MOV A,#ADRHI
025D	83	961	RET
		962	*****
		963	STATUS:PODPROGRAM KTORY SA VYUZIVA PRI KAZDOM STARTE UZIVA-
		964	----- TELSKEHO PROGRAMU A RESTARTE MONITORU.ZABEZPECUJE
		965	VZAJOMNU VYMENU UZIVATELOM DEFINOVANYCH PRACOVNYCH
		966	REGISTROV A INTERNEJ RAM ZA MONITOROVSKIE REGISTRE A
		967	INTERNU RAM. PROGRAM PRENESIE VSETKY UZIVATELSKE RE-
		968	GISTRE S VYNI MKOU REG. R7 V RB1. ACC SA ODLOZI AKO
		969	R17. PROGRAM DALEJ ODKLADA PODLA ADRESY V SMAHI+SMALO
		970	(STARTOVACIA ADRESA UZIVATELSKEHO PROGRAMU) NA ADRESU
		971	07D4H - OPERACNY KOD INSTRUKCIE SEL MB0, RESP. SEL MB1,
		972	NA ADRESU 07D5H - OPERACNY KOD ZODPOVEDAJUCEJ INSTRUK-
		973	CIE JMP A NA ADRESU 07D6H DOLNYCH 8 BITOV ADRESY SKOKU
		974	;
025E	D5	975	STATUS: SEL RB1
025F	2F	976	XCH A,R7
0260	28	977	XCH A,R0
		978	;
		979	ODLOZ R0 NA ADRESU 13H
		980	;
0261	B813	981	MOV R0,#13H
0263	A0	982	MOV @R0,A
		983	;
		984	ODLOZ ACC NA ADRESU 17H
		985	;
0264	B817	986	MOV R0,#17H
0266	2F	987	XCH A,R7
0267	A0	988	MOV @R0,A
0268	B830	989	MOV R0,#SMALO
026A	F0	990	MOV A,@R0
		991	;
		992	VYPOCITAJ CI BUDE PROGRAM STARTOVANY V BANKE
		993	MB0 ALEBO MB1
		994	;
026B	AF	995	MOV R7,A
026C	18	996	INC R0
026D	F0	997	MOV A,@R0
		998	;
		999	PODLA TOHO VLOZ KOD 0E5H ALEBO 0F5H NA ADRESU
		1000	7D4H,T.J.14H V INTERNEJ RAM
		1001	;
026E	B814	1002	MOV R0,#14H
0270	B0F5	1003	MOV @R0,#0F5H
0272	7276	1004	JB3 BANKA1
0274	B0E5	1005	MOV @R0,#0E5H
0276	18	1006	BANKA1: INC R0
0277	47	1007	SWAP A
		1008	;
		1009	VYPOCITAJ OPERACNY KOD PRISLUSNEJ INSTRUKCIE JMP
		1010	A ULOZ NA 7D5H(15H)
		1011	;
0278	97	1012	CLR C



LOC	OBJ	LINE	SOURCE STATEMENT
0279	F7	1013	RLC A
027A	4304	1014	ORL A,#04H
027C	A0	1015	MOV @R0,A
		1016 ;	
		1017 ;	
		1018 ;	DOLNYCH 8 BITOV STARTOVACEJ ADRESY PROGRAMU VLOZ
		1019 ;	NA 7D6H(16)
027D	18	1020	INC R0
027E	FF	1021	MOV A,R7
027F	A0	1022	MOV @R0,A
0280	B8C0	1023	MOV R0,#0C0H
		1024 ;	
		1025 ;	PRENES RB0 NA ADRESY 7C0H A 7C7H
		1026 ;	
0282	54AE	1027	STAT1: CALL STATXY
0284	D3C7	1028	XRL A,#0C7H
0286	9682	1029	JNZ STAT1
0288	B813	1030	MOV R0,#13H
		1031 ;	
		1032 ;	PREPNI NA BANKU RB0 A PRIPRAV NA PRENOS BANKU
		1033 ;	RB1,ACC A START
		1034 ;	
028A	F0	1035	MOV A,@R0
		1036 ;	
		1037 ;	ADRESA PROGRAMU
		1038 ;	
028B	28	1039	XCH A,R0
028C	C5	1040	SEL RB0
028D	28	1041	XCH A,R0
028E	A0	1042	MOV @R0,A
028F	C8	1043	DEC R0
0290	FF	1044	MOV A,R7
0291	A0	1045	MOV @R0,A
0292	B8D4	1046	MOV R0,#0D4H
		1047 ;	
		1048 ;	PRENOS NA ADRESY 7D4 AZ 7FFH
		1049 ;	
0294	54AE	1050	STAT2: CALL STATXY
0296	D3FF	1051	XRL A,#0FFH
		1052 ;	
		1053 ;	RB1 A OSTATNE ADRESY RAM
		1054 ;	
0298	9694	1055	JNZ STAT2
029A	B812	1056	REFR: MOV R0,#12H
029C	F0	1057	MOV A,@R0
029D	AF	1058	MOV R7,A
029E	B817	1059	MOV R0,#17H
02A0	F0	1060	MOV A,@R0
02A1	B813	1061	MOV R0,#13H
		1062 ;	
		1063 ;	OBNOV HODNOTY REGISTROV R0,R7 A ACC
		1064 ;	
02A3	20	1065	XCH A,@R0
02A4	28	1066	XCH A,R0
02A5	D5	1067	SEL RB1

LUC	OBJ	LINE	SOURCE STATEMENT
02A6	28	1068	XCH A,R0
02A7	2F	1069	XCH A,R7
02A8	F0	1070	MOV A,@R0
02A9	2F	1071	XCH A,R7
02AA	28	1072	XCH A,R0
02AB	2F	1073	XCH A,R7
02AC	C5	1074	SEL R0
02AD	93	1075	RETR
		1076	*****
		1077	;STATXY;POMOCNY PODPROGRAM PRE VYMENU 1 BYTU INTERNEJ RAM A
		1078	;----- PAMATI PROGRAMU RAM
		1079	;
02AE	2307	1080	STATXY; MOV A,#ADRH1
02B0	3A	1081	OUTL P2,A
02B1	80	1082	MOVX A,@R0
02B2	2F	1083	XCH A,R7
		1084	;
		1085	; NACITAJ DO ACC OBSAH UZIV. BYTU RAM-ODLOZ DO R7
		1086	;
02B3	F8	1087	MOV A,R0
		1088	;
		1089	; NACITAJ DO ACC BYTE Z INTERNEJ RAM
		1090	;
02B4	533F	1091	ANL A,#3FH
02B6	A8	1092	MOV R0,A
02B7	F0	1093	MOV A,@R0
		1094	;
		1095	; ODLOZ DO R7
		1096	;
02B8	2F	1097	XCH A,R7
		1098	;
		1099	; BYTE Z UZIVATELSKEJ RAM -> INT.RAM
		1100	;
02B9	A0	1101	MOV @R0,A
02BA	F8	1102	MOV A,R0
		1103	;
		1104	; PREPNI NA UZIV. RAM
		1105	;
02BB	43C0	1106	ORL A,#0C0H
02BD	A8	1107	MOV R0,A
		1108	;
		1109	; VLOZ DO ACC BYTE Z INT. RAM A ZAPIS DO UZIV. RAM
		1110	;
02BE	2F	1111	XCH A,R7
02BF	90	1112	MOVX @R0,A
02C0	F8	1113	MOV A,R0
02C1	18	1114	INC R0
02C2	9AF3	1115	ANL P2,#0F3H
02C4	83	1116	RET
		1117	*****
		1118	;RST2 ;PROGRAM PRE NAVRAT Z BREAKU A OBNOVENIE UZIVATELSKYCH
		1119	;----- REGISTRŮV,S MOZNOSTOU DALSIEHO VYKONAVANIA BREAKOV A
		1120	; SO ZOBRAZENIM OKAMZITEJ HODNOTY PC A ACC NA DISPLEJI
		1121	;
		1122	; OBNOV UZIVATELSKE REGISTRE

LOC	OBJ	LINE	SOURCE STATEMENT
		1123 ;	
02C5	545E	1124 RST2:	CALL STATUS
02C7	B821	1125	MOV R0,#SAVE
02C9	C7	1126	MOV A,PSW
02CA	A0	1127	MOV @R0,A
		1128 ;	
		1129 ;	DBNOV DP.KOD NA ADRESE BREAK
		1130 ;	
02CB	74A3	1131	CALL BRBACK
		1132 ;	
		1133 ;	ODSTARTUJ TIMER PRE REFRES DISPLEJOV
		1134 ;	
02CD	55	1135	STRT T
		1136 ;	
		1137 ;	ULOZ SPRAVU 'PC' DO REG. DISPLEJA
		1138 ;	
02CE	2302	1139	MOV A,#02H
02D0	3400	1140	CALL OUTUTL
		1141 ;	
		1142 ;	VLOZ ADRESU BREAK DO EMA A ZAPIS DO DALSIICH 3 TROCH ARE-
		1143 ;	GISTROV DISPLEJA
		1144 ;	
02D2	7451	1145	CALL BRKADR
02D4	B833	1146	MOV R0,#EMAH1
02D6	34D8	1147	CALL UPDAD1
		1148 ;	
		1149 ;	NASTAV DO SMA ADRESU ODLOZENIA ACC V PAMATI 2114(7D7H)
		1150 ;	
02D8	B830	1151	MOV R0,#SMALD
02DA	B0D7	1152	MOV @R0,#LOW ACCADR
07D7		1153 ACCADR	EQU 07D7H
02DC	18	1154	INC R0
02DD	B007	1155	MOV @R0,#HIGH ACCADR
		1156 ;	
		1157 ;	PRENES OBSAH UZIV. ACC DO ACC
		1158 ;	
02DF	542C	1159	CALL LFPEM
		1160 ;	
		1161 ;	ZOBRAZ DO REGISTROV DISPLEJA GKAMZITU HODNOTU UZI-
		1162 ;	VATELSKEHO ACC NA ADRESE BREAKU,PO KTOTY SA VYKONAL
		1163 ;	PROGRAM
		1164 ;	
02E1	47	1165	SWAP A
02E2	B925	1166	MOV R1,#NEXTPL
02E4	B102	1167	MOV @R1,#02H
02E6	3496	1168	CALL DSPACC
02E8	FA	1169	MOV A,R2
02E9	3496	1170	CALL DSPACC
		1171 ;	
		1172 ;	ZOBRAZ PC XXX YY NA DISPLEJ A VRAT SA DO MONITORA
		1173 ;	
02EB	0420	1174	JMP MAIN4
		1175 ;	*****
		1176 ;	BUFSTR.PODPROGRAM PRE PRESUN BYTU Z ADRESY ULOZENEJ V EMA
		1177 ;	----- NA ADRESU ULOZENU V SMA

LOC	OBJ	LINE	SOURCE STATEMENT
		1178	;
02ED	B833	1179	BUFSTR: MOV R0,#EMAH1
02EF	542E	1180	CALL LFEPM+2
02F1	543F	1181	CALL LSTPM
02F3	83	1182	RET
		1183	*****
		1184	;SMAMOV:PODPROGRAM PRE ODLOZENIE UDAJOV V SMA DO MEM
		1185	;-----
		1186	;
02F4	B830	1187	SMAMOV: MOV R0,#SMALO
02F6	B934	1188	MOV R1,#MEMLO
02F8	F0	1189	SMAMV1: MOV A,@R0
02F9	A1	1190	MOV @R1,A
02FA	18	1191	INC R0
02FB	19	1192	INC R1
02FC	F0	1193	MOV A,@R0
02FD	A1	1194	MOV @R1,A
02FE	83	1195	RET
		1196	*EJECT

LOC	OBJ	LINE	SOURCE STATEMENT
0300		1197	ORG 300H
0300	00	1198	INVALS: DB 00H ;ROTPAT 1AH
0301	00	1199	DB 00H ;ROTCNT 1BH
0302	00	1200	DB 00H ;LASTKY 1CH
0303	08	1201	DB 08H ;CURDIG 1DH
0304	00	1202	DB 00H ;KEYFIG 1EH
0305	00	1203	DB 00H ;(REG7) 1FH
0306	04	1204	DB 04H ;NREPTS 20H
0307	08	1205	DB 08H ;SAVE 21H
0308	FF	1206	DB 0FFH ;BRKBUF 22H
0309	00	1207	DB 00H ;STRTMP 23H
030A	00	1208	DB 00H ;OPTION 24H
030B	08	1209	DB 08H ;NEXTPL 25H
030C	FF	1210	DB 0FFH ;KBDBUF 26H
030D	00	1211	DB 00H ;KEYLOC 27H
		1212	;*****
		1213	; CHYBOVY VYPIS 'ERROR'
		1214	;
030E	79	1215	ERROR: DB 79H ;E
030F	50	1216	DB 50H ;R
0310	50	1217	DB 50H ;R
0311	5C	1218	DB 5CH ;O
0312	D0	1219	DB 0D0H ;R
		1220	;*****
		1221	; INICIALIZACNA SPRAVA 'SB-8035'
		1222	;
0313	6D	1223	SBMSG: DB 6DH ;S
0314	7C	1224	DB 7CH ;B
0315	40	1225	DB 40H ;-
0316	7F	1226	DB 7FH ;8
0317	3F	1227	DB 3FH ;0
0318	4F	1228	DB 4FH ;3
0319	ED	1229	DB 0EDH ;5
		1230	;*****
		1231	; VYPIS 'PC'
		1232	;
031A	73	1233	PCMSG: DB 73H ;P
031B	B9	1234	DB 0B9H ;C
		1235	;*****
		1236	; TABULKA 7-SEGMENTOVYCH KODOV PRE HEXADECIMALNE CISLICE
		1237	;
031C	3F	1238	DGPATS: DB 3FH ;0
031D	06	1239	DB 06H ;1
031E	5B	1240	DB 5BH ;2
031F	4F	1241	DB 4FH ;3
0320	66	1242	DB 66H ;4
0321	6D	1243	DB 6DH ;5
0322	7D	1244	DB 7DH ;6
0323	07	1245	DB 07H ;7
0324	7F	1246	DB 7FH ;8
0325	67	1247	DB 67H ;9
0326	77	1248	DB 77H ;A
0327	7C	1249	DB 7CH ;B
0328	39	1250	DB 39H ;C
0329	5E	1251	DB 5EH ;D

LOC	OBJ	LINE	SOURCE STATEMENT
032A	79	1252	DB 79H ;E
032B	71	1253	DB 71H ;F
		1254	*****
		1255	; TABULKA POVOLENYCH PRIKAZOV
		1256	;
032C	1F	1257	CTAB: DB KEYMOD ;EXAMIN-KEYMOD
032D	4A	1258	DB LOW OPTAB1 ;OPTAB1
032E	1E	1259	DB KEYGO ;KEYGO
032F	4E	1260	DB LOW OPTAB2 ;OPTAB2
0330	FF	1261	DB OFFH ;UKONC ZNAK
		1262	;
		1263	; ADRESA SPRAVY ERROR,SB-8035,PC
		1264	;
0331	0E	1265	STRUTL: DB LOW ERROR ;ERROR
0332	13	1266	DB LOW SBMSG ;SB-8035
0333	1A	1267	DB LOW PCMSG ;PC
		1268	;
		1269	; MODIFIKATORY PRIKAZOV
		1270	;
0334	3B	1271	STRCOM: DB LOW DMOD ;DMOD (EXAM)
0335	3E	1272	DB LOW DGO ;DGO (GO)
		1273	;
		1274	; MODIFIKATORY PAMATI
		1275	;
0336	40	1276	STRMEM: DB LOW DPRMEM ;DPRMEM
0337	42	1277	DB LOW DDAMEM ;DDAMEM
0338	44	1278	DB LOW DRM ;DRM
		1279	;
		1280	; MODIFIKATORY STARTOVANIA PROGRAMU
		1281	;
0339	46	1282	STRGOC: DB LOW DNOBRK ;DNOBRK
033A	48	1283	DB LOW DWBRK ;DWBRK
		1284	;
		1285	; VYPIS 'ECH'
		1286	;
033B	79	1287	DMOD: DB 79H ;E FUNKCIA ECH
033C	39	1288	DB 39H ;C
033D	F4	1289	DB 0F4H ;H
		1290	;
		1291	; VYPIS 'GO'
		1292	;
033E	3D	1293	DGO: DB 3DH ;G
033F	DC	1294	DB 0DCH ;D
		1295	;
		1296	; VYPIS 'PR'
		1297	;
0340	73	1298	DPRMEM: DB 73H ;P
0341	D0	1299	DB 0D0H ;R
		1300	;
		1301	; VYPIS 'DA'
		1302	;
0342	5E	1303	DDAMEM: DB 5EH ;D
0343	F7	1304	DB 0F7H ;A
		1305	;
		1306	; VYPIS 'RG'

LOC	OBJ	LINE	SOURCE STATEMENT
		1307 ;	
0344	50	1308	DRM: DB 50H ;R
0345	BD	1309	DB 0BDH ;G
		1310 ;	
		1311 ;	VYPIS 'NB'
		1312 ;	
0346	54	1313	DNOBRK: DB 54H ;N
0347	FC	1314	DB 0FCH ;B
		1315 ;	
		1316 ;	VYPIS 'BR'
		1317 ;	
0348	7C	1318	DWBRK: DB 7CH ;B
0349	D0	1319	DB 0D0H ;R
		1320 ;	
		1321 ;	TABULKA KODOV TLACITOK PRIKAZU EXAMIN
		1322 ;	
034A	36	1323	OPTAB1: DB LOW STRMEM ;LOW STRMEM
034B	1A	1324	DB LOW KEYPM ;KEYPM
034C	16	1325	DB LOW KEYDM ;KEYDM
034D	9B	1326	DB 9BH ;KEYREG
		1327 ;	
		1328 ;	TABULKA KODOV TLACITOK PRIKAZU GO
		1329 ;	
034E	39	1330	OPTAB2: DB LOW STRGOC ;LOW STRGOC
034F	1B	1331	DB 1BH ;NOBRK
0350	96	1332	DB 96H ;WBRK
		1333 ;	*****
		1334 ;	BRKADR: PROGRAM PRE PRENESENIE ADRESY BREAKU DO REGISTRA EMA.
		1335 ;	----- BRKBUF UKAZUJE HLBKU BREAKU-MAX. 96 KANALOV. BREAKB EQU 07BEH
		1336 ;	
		1337 ;	VLOZ HORNU ODKLADACIU ADRESU BREAKOV DO SMA
		1338 ;	
07BE		1339	BREAKB EQU 07BEH
0351	B831	1340	BRKADR: MOV R0,#SMAHI
0353	B007	1341	MOV @R0,#HIGH BREAKB
0355	C8	1342	DEC R0
0356	B0BE	1343	MOV @R0,#LOW BREAKB
0358	B922	1344	MOV R1,#BRKBUF
		1345 ;	
		1346 ;	VYPOCITAJ SKUTOCNU ADRESU PODLA CISLA (PORADOVA)
		1347 ;	BREAKU, T.J. PRI KAZDOM DALSOM JE O 2 NIZSIA
		1348 ;	
035A	F1	1349	MOV A,@R1
035B	97	1350	CLR C
035C	F7	1351	RLC A
035D	37	1352	CPL A
035E	17	1353	INC A
035F	60	1354	ADD A,@R0
0360	A0	1355	MOV @R0,A
		1356 ;	
		1357 ;	V R2 A ACC JE LSB BREAKU
		1358 ;	
0361	B932	1359	MOV R1,#EMALO
0363	542C	1360	CALL LFEPM
		1361 ;	

LOC	OBJ	LINE	SOURCE STATEMENT
		1362 ;	ODLOZ HO DO EMALO
		1363 ;	
0365	A1	1364	MOV @R1,A
		1365 ;	
		1366 ;	PRESUN NA MSB BREAKU
		1367 ;	
0366	19	1368	INC R1
0367	B830	1369	MOV R0,#SMALO
0369	10	1370	INC @R0
		1371 ;	
		1372 ;	PRENES MSB BREAKU DO R2 A ACC
		1373 ;	
036A	542C	1374	CALL LFEPM
036C	A1	1375	MOV @R1,A
036D	97	1376	CLR C
		1377 ;	
		1378 ;	TESTUJ CI BOL POSLEDNY BREAK,AK NIE NULUJ CY
		1379 ;	AK AND CY=1
		1380 ;	
036E	D3FF	1381	XRL A,#OFFH
0370	9673	1382	JNZ BRKAD1
0372	A7	1383	CPL C
0373	83	1384	BRKAD1: RET
0374	FF	1385	DB OFFH
0375	FF	1386	DB OFFH
		1387 ;	*****
		1388 ;	GOBRK :RIADIACI PROGRAM PRE STARTOVANIE UZIVATELSKEHO PRO-
		1389 ;	----- GRAMU S BREAKMI
		1390 ;	
		1391 ;	ODLOZ START. ADRESU PROGRAMU DO MEM
		1392 ;	
		1393 ;	
0376	B821	1394	GOBRK: MOV R0,#SAVE
0378	C7	1395	MOV A,PSW
0379	A0	1396	MOV @R0,A
037A	54F4	1397	CALL SMAMOV
037C	B822	1398	MOV R0,#BRKBUF
		1399 ;	
		1400 ;	NASTAV HODNOTU VYKONANYCH BREAKOV NA 0 A PO KAZ-
		1401 ;	DOM NAVRATE ZVYS 0 1
		1402 ;	
037E	B0FF	1403	MOV @R0,#OFFH
0380	10	1404	GOBRKZ: INC @R0
		1405 ;	
		1406 ;	PRESUN ADRESY BREAKU DO EMA AK UZ NIE JE POZADOVANY
		1407 ;	BREAK,POTOM KONEC OPCODE -> BUFF,JMPRST2 -> BREAK
		1408 ;	
0381	7451	1409	CALL BRKADR
0383	F687	1410	JC GOBRKY
0385	14E4	1411	CALL BRKMOV
		1412 ;	
		1413 ;	NAVRAT VYKONAVACEJ ADRESY DO SMA
		1414 ;	
0387	B930	1415	GOBRKY: MOV R1,#SMALO
0389	B834	1416	MOV R0,#MEMLO



LOC	OBJ	LINE	SOURCE STATEMENT
038B	54F8	1417	CALL SMAMV1
		1418 ;	
		1419 ;	ODLOZ STAVOVE SLOVO UZIVATELSKEHO PROGRAMU
		1420 ;	
038D	B821	1421	MOV R0,#SAVE
038F	C7	1422	MOV A,PSW
0390	47	1423	SWAP A
0391	20	1424	XCH A,@R0
0392	47	1425	SWAP A
0393	30	1426	XCHD A,@R0
0394	20	1427	XCH A,@R0
0395	47	1428	SWAP A
0396	D7	1429	MOV PSW,A
		1430 ;	
		1431 ;	VYKONAJ PROGRAM PO NASLEDUJUCI BREAK
		1432 ;	
0397	04DF	1433	JMP GONBRK
		1434 ;	
		1435 ;	TU POKRACUJE PROGRAM PO STLACENI TLACITKA 'NEXT'
		1436 ;	
0399	7451	1437	GOBRKX: CALL BRKADR
039B	B832	1438	MOV R0,#EMALO
		1439 ;	
		1440 ;	INKREMENTUJ POCITADLO BREAKOV A ODLOZ P'OKRAC. ADRESU
		1441 ;	DO MEM A VYKONAJ PROGRAM PO DALSI BREAK
		1442 ;	
039D	54F6	1443	CALL SMAMOV+2
039F	B822	1444	MOV R0,#BRKBUF
03A1	6480	1445	JMP GOBRKZ
		1446 ;	*****
		1447 ;	BRBACK:PODPROGRAM PRE NAVRAT OPERACNEHO KODU NA POVODNU ADRESU
		1448 ;	----- BREAK A BREAK+1
		1449 ;	
		1450 ;	VLOZ DO EMA BREAK
		1451 ;	
03A3	7451	1452	BRBACK: CALL BRKADR
		1453 ;	
		1454 ;	VLOZ DO SMA ADRESU BUFF
		1455 ;	
03A5	B831	1456	MOV R0,#SMAHI
03A7	B007	1457	MOV @R0,#HIGH BUFF
03A9	C8	1458	DEC R0
03AA	B0D0	1459	MOV @R0,#LOW BUFF
		1460 ;	
		1461 ;	ODPAMATAJ 1.BYTE OP. KODU Z BUFF NA BREAK
		1462 ;	
03AC	74B4	1463	CALL BRBAC2
		1464 ;	
		1465 ;	PRESUN NA ADRESU BUFF+1 A BREAK+1
		1466 ;	
03AE	B830	1467	MOV R0,#SMALO
03B0	10	1468	INC @R0
03B1	B832	1469	MOV R0,#EMALO
03B3	10	1470	INC @R0
		1471 ;	

LOC	OBJ	LINE	SOURCE STATEMENT
		1472 ;	PRENOS 2. BYTE INSTRUKCIE Z BUFF+1 NA BREAK+1
		1473 ;	
03B4	542C	1474	BRBAC2: CALL LFPEM
03B6	B933	1475	BRBAC1: MOV R0,#EMAH1
03B8	5441	1476	CALL LSTPM+2
03BA	83	1477	RET
		1478	END

USER SYMBOLS

ACCADR 07D7	ADRHI 0007	ADRLD 00BF	ASAVE 0007	BANKA1 0276	BCODE 0005	BRBAC1 03B6	BRBAC2 03B4
BRBACK 03A3	BREAKB 07BE	BRKAD1 0373	BRKADR 0351	BRKBUF 0022	BRKMOV 00E4	BUFF 07D0	BUFSTR 02ED
CHARNO 0008	CLEAR 0127	CMDINT 0081	CTAB 032C	CURDIG 0005	DBLANK 012B	DDAMEM 0342	DECSM1 01FF
DECSMA 01F4	DECW 01F6	DGO 033E	DGPATS 031C	DMOD 033E	DNOBRK 0346	DPRMEM 0340	DRM 0344
DSPACC 0196	DSPHI 01EA	DSPLD 01F0	DSPM1 01EE	DSPMID 01EC	DSPTMP 0006	DWBRK 0348	ELSIF1 01BF
ELSIF2 01CD	EMAH1 0033	EMALD 0032	ERROR 030E	ERROR2 0223	EXAM0 008E	EXAM1 00C5	EXAM2 00C9
EXAM3 00D2	EXAM4 00DB	EXAM5 00BF	EXAMIN 008D	EXAMR 009C	FINDOP 003B	GOBRK 0376	GOBRKX 0399
GOBRKY 0387	GOBRKZ 0380	GOBRXX 0005	GONBRK 00DF	INCSMA 0200	INCW 0202	INCW1 0208	INIT 0000
INIT1 0009	INITLP 0014	INPADR 01B0	INPKEY 0135	INVALS 0300	ITMP 0004	JMPCD 0044	KBDBUF 0026
KEY 0003	KEYCLR 0017	KEYDM 0016	KEYEND 0013	KEYFLG 0006	KEYGO 001E	KEYLOC 0027	KEYMOD 001F
KEYNXT 0012	KEYPM 001A	KEYREG 001B	LASTKY 0004	LDATA 0002	LFEDM 022C	LFPEM 022C	LFPEM1 0230
LFEREG 024B	LFETBL 0229	LFETCH 0225	LFLST 0253	LSTBL 023C	LSTDN 023F	LSTORE 0238	LSTPM 023F
LSTPM1 0243	LSTREG 024F	MAIN 001C	MAIN2 0022	MAIN3 0035	MAIN4 0020	MAINA 0049	MAINB 0058
MAINB0 007D	MAINC1 0064	MAIND 0074	MAIND1 0072	MEMHI 0035	MEMLD 0034	MERROR 0089	NCOLS 0004
NEXTPL 0025	NOVALS 000E	NREPTS 0020	NUMCON 0038	NXTLOC 015E	OPTAB1 034A	OPTAB2 034E	OPTION 0024
OUTCLR 0102	OUTMSG 0104	OUTUTL 0100	PCMSG 031A	PDIGIT 000E	PERROR 020B	PINPUT 000B	PRNT1 0117
PRNT2 0108	PSEGLI 000D	PSEGLD 000C	REFR 029A	RERROR 0209	ROTCNT 0003	ROTPAT 0002	RST1 011B
RST1A 0125	RST2 02C5	RUN 07D4	SAVE 0021	SBMSG 0313	SCAN3 0172	SCAN5 017F	SCAN8 0191
SEGMAP 0028	SMAHI 0031	SMALD 0030	SMAMOV 02F4	SMAMV1 02F8	STAT1 0282	STAT2 0294	STATUS 025E
STATXY 02AE	STRCOM 0334	STRGOC 0339	STRMEM 0336	STRTMP 0023	STRUTL 0331	TIINT 0146	TIRET1 0193
TOFPOL 0143	TYPE 0007	UPDAD1 01D8	UPDADR 01D4	USRINT 0403	USRTIM 0407	WDISP 019B	WDISP1 01AF
ZERO 0000							

ASSEMBLY COMPLETE, NO ERRORS

ACCADR	1152	1153#	1155						
ADRHI	8#	960	1080						
ADRLO	9#	957							
ASAVE	66#	606	693						
BANKA1	1004	1006#							
BCODE	55#	222	236	241	310				
BRBAC1	454	471	1475#						
BRBAC2	1463	1474#							
BRBACK	528	1131	1452#						
BREAKB	1339#	1341	1343						
BRKAD1	1382	1384#							
BRKADR	1145	1340#	1409	1437	1452				
BRKBUF	10#	209	524	1344	1398	1444			
BRKMOV	444#	1411							
BUFF	440#	445	447	1457	1459				
BUFSTR	452	465	1179#						
CHARNO	11#	545	550	679	744				
CLEAR	483	544#							
CMDINT	310#								
CTAB	218	1257#							
CURDIG	64#	612	617	678	679				
DBLANK	546#	548	789						
DDAMEM	1277	1303#							
DECSM1	841	846#							
DECSMA	419	833#							
DECW	838#								
DGO	1272	1293#							
DGPATS	717	1238#							
DMOD	1271	1287#							
DNOBRK	1282	1313#							
DPRMEM	1276	1298#							
DRM	1278	1308#							
DSPACC	373	378	716#	822	825	827	890	1168	1170
DSPHI	814	822#							
DSPLO	821	826#							
DSPM1	819	825#							
DSPMID	823#								
DSPTMP	56#	730	737						
DWBRK	1283	1318#							
ELSIF1	767	779#							
ELSIF2	781	784	787#						
EMAHI	13#	1146	1179	1475					
EMALO	12#	460	958	1359	1438	1469			
ERROR	1215#	1265							
ERROR2	886	897#							
EXAM0	348#	398							
EXAM1	385	403#							
EXAM2	405#								
EXAM3	410	413#							
EXAM4	418	425#							
EXAM5	389	392#							
EXAMIN	314	344#	412	420					
EXAMR	350	357#							
FINDOP	228#	237							
GOBRK	323	1394#							
GOBRKX	168	1437#							
GOBRKY	1410	1415#							





Rozpiska súčiastok

Integrované obvody :

IO 1 MHB 8035 /MHB 8048, MHB 8748/  
IO 2 MHB 8243  
IO 3 MHB 8243  
IO 4 MH 3212  
IO 5 MHB 2716 /K573 P05/  
IO 6 MHB 2114  
IO 7 MHB 2114  
IO 8 MH 3205 /UCY 74S405/  
IO 9 MH 3216  
IO 10 MH 7410  
IO 11 MH 7400  
IO 12 74145 PC  
IO 13 MHB 8155

Tranzistory :

T1 až T8 KFY 16 /KFY 18, BC 313/  
T9 KSY 82  
T10 až T17 KSY 63 /KSY 62B, KSY 21/

Diódy a sedemsegmentové zobrazovače :

D1 až D8 LQ 410 /LQ 440, LQ 470/  
D9 KSY 130/80  
D10 KA 207  
D11 WK 16412 /WK 16413/

Odpory :

R1 až R28 2k7  
R29 až R36 22R  
R37 až R44 2k2 všetky rada TR 191  
R45 až R52 100R  
R53 1k5  
R54 18k

Kondenzátory :

C1 až C7	TK 782	68k
C8,C9	TK 754	22j
C10 až C12	TF 009	100M
C13	TE 988	1M

Kryštál :

XTAL 1 (KD2/13) 6 MHz /4,43 MHz, 4,194 MHz, 4 MHz/

Pätice pre IO :

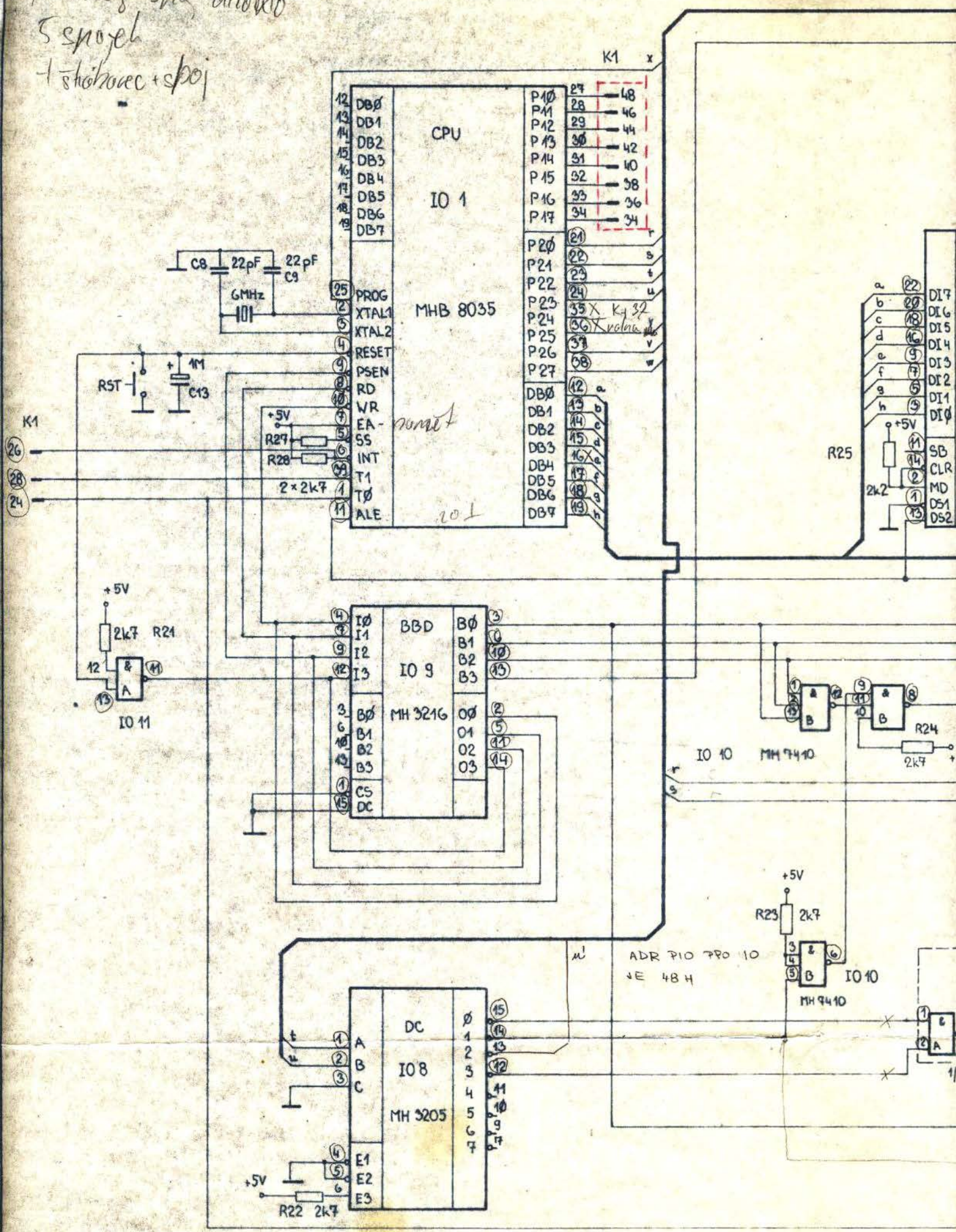
2 x 7	TX 782 2141	8 ks
2 x 12	TX 782 5251	3 ks
2 x 20	TX 787 5401	2 ks

Tlačidlá 0 až 1F, RST	3FK 573 00	25 ks
Hmatníky	3FA 401 15	25 ks

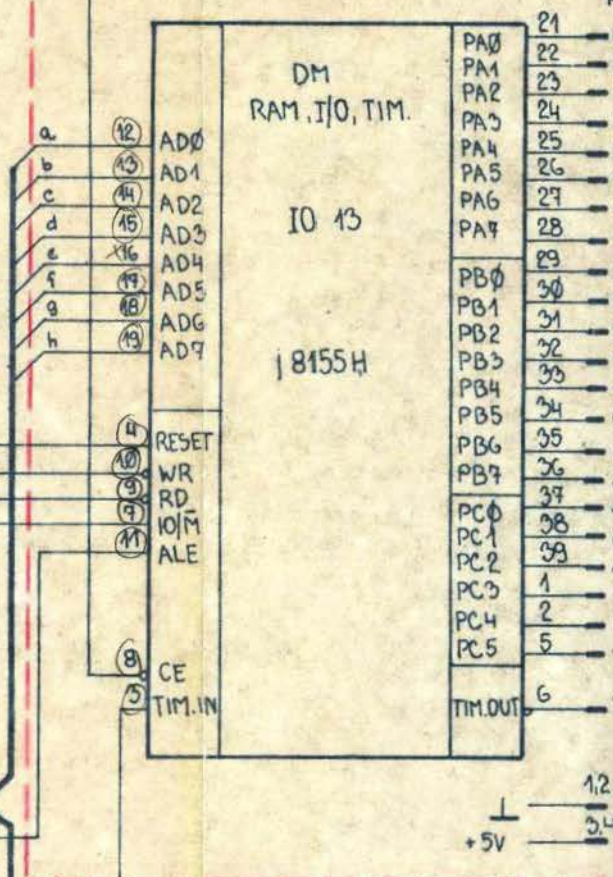
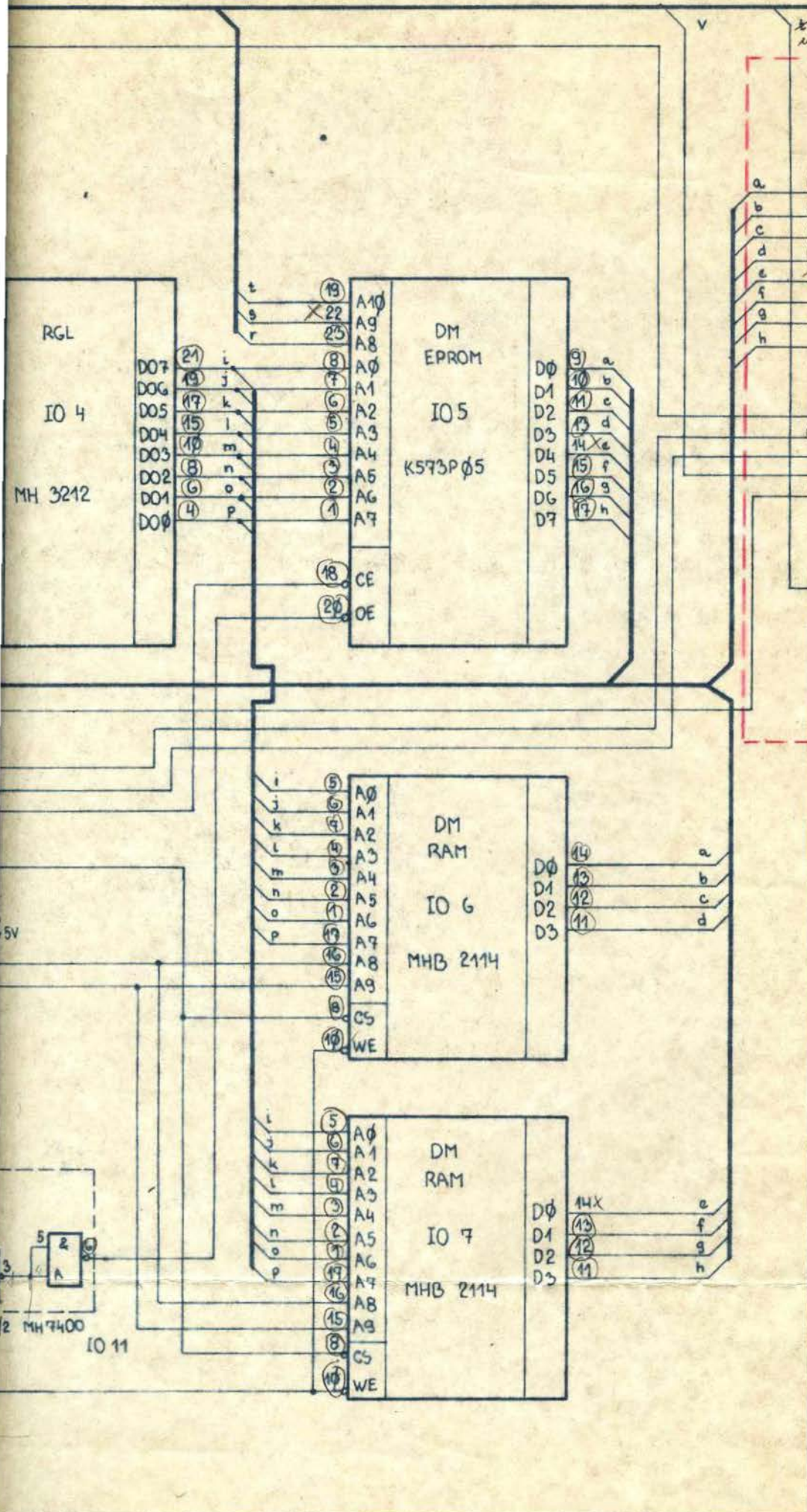
Konektory :

K1 48-pólový	TX 516 48 12
K2 20-pólový	TX 512 20 12

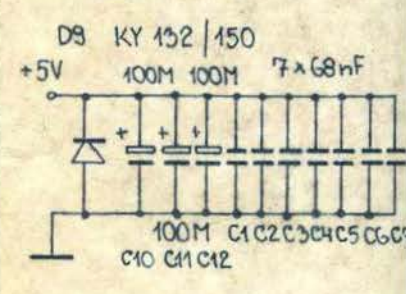
převodný spa - dílko  
 5 spoj  
 + šokovací spoj

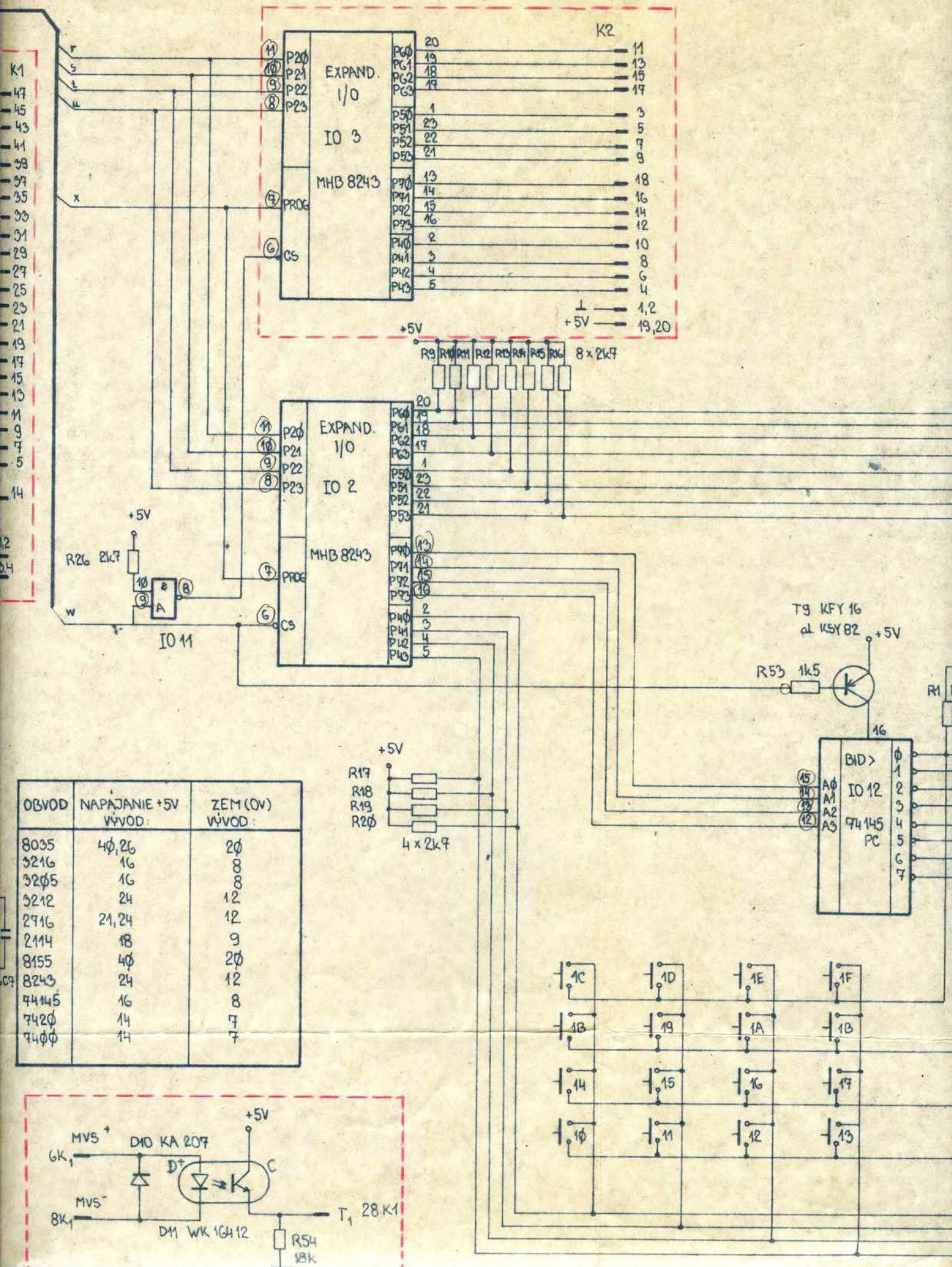




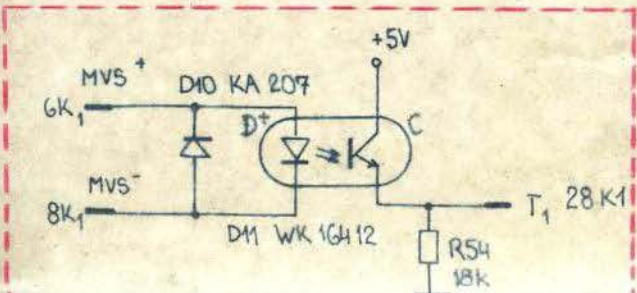


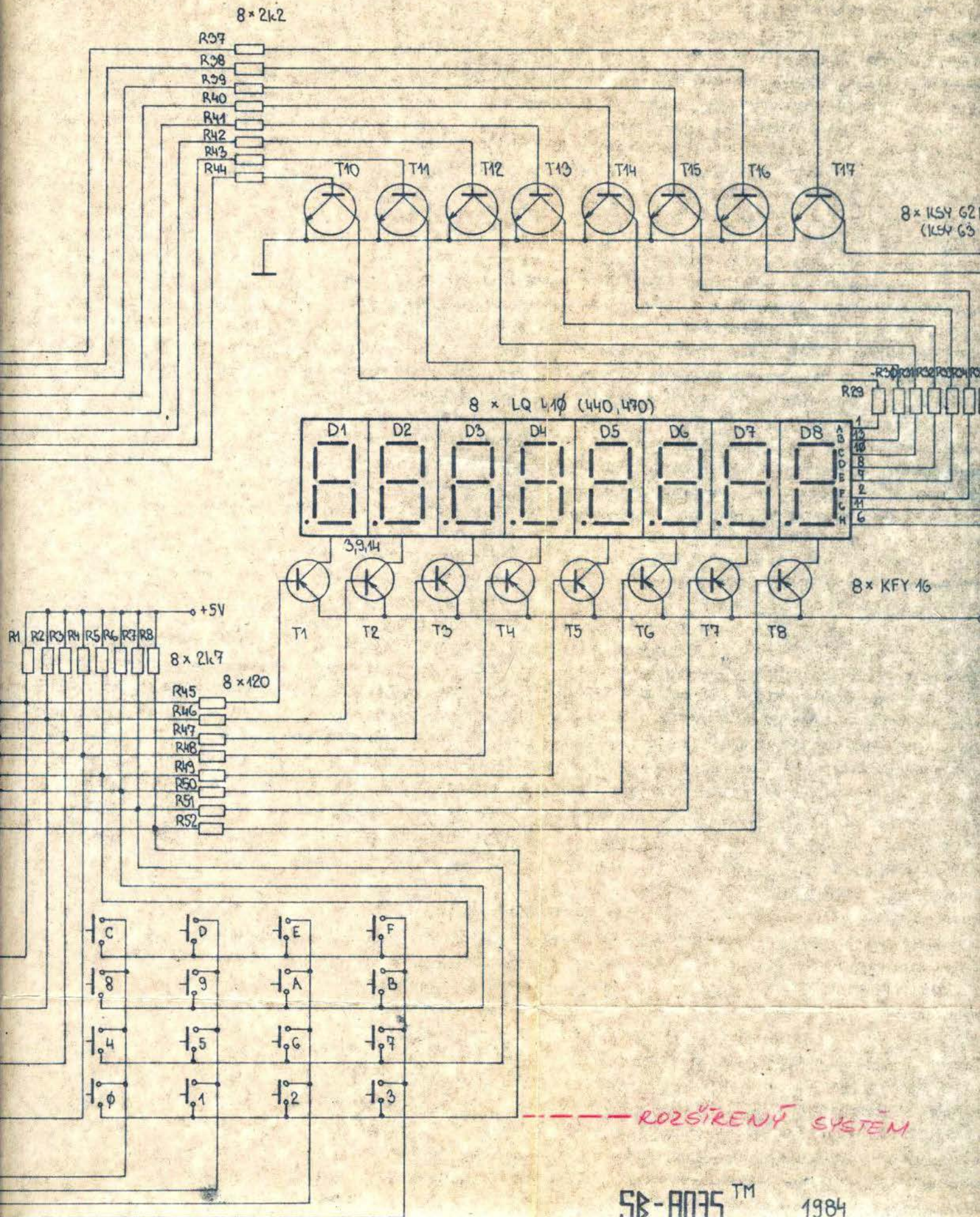
FILTRÁCIA A BLOKOVANIE ROZVODU +5V

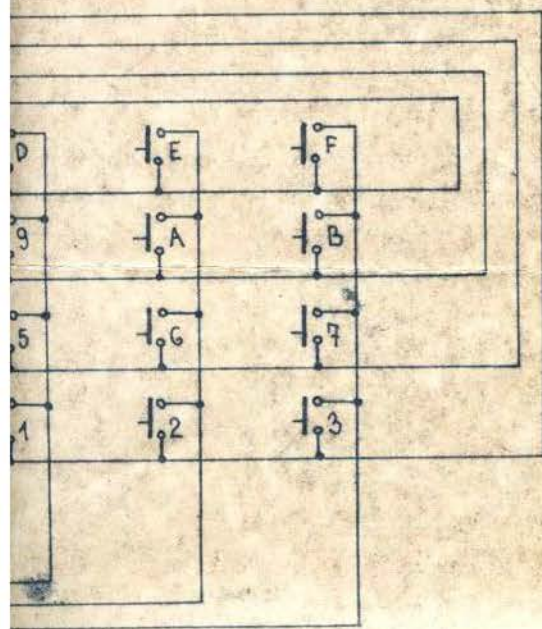
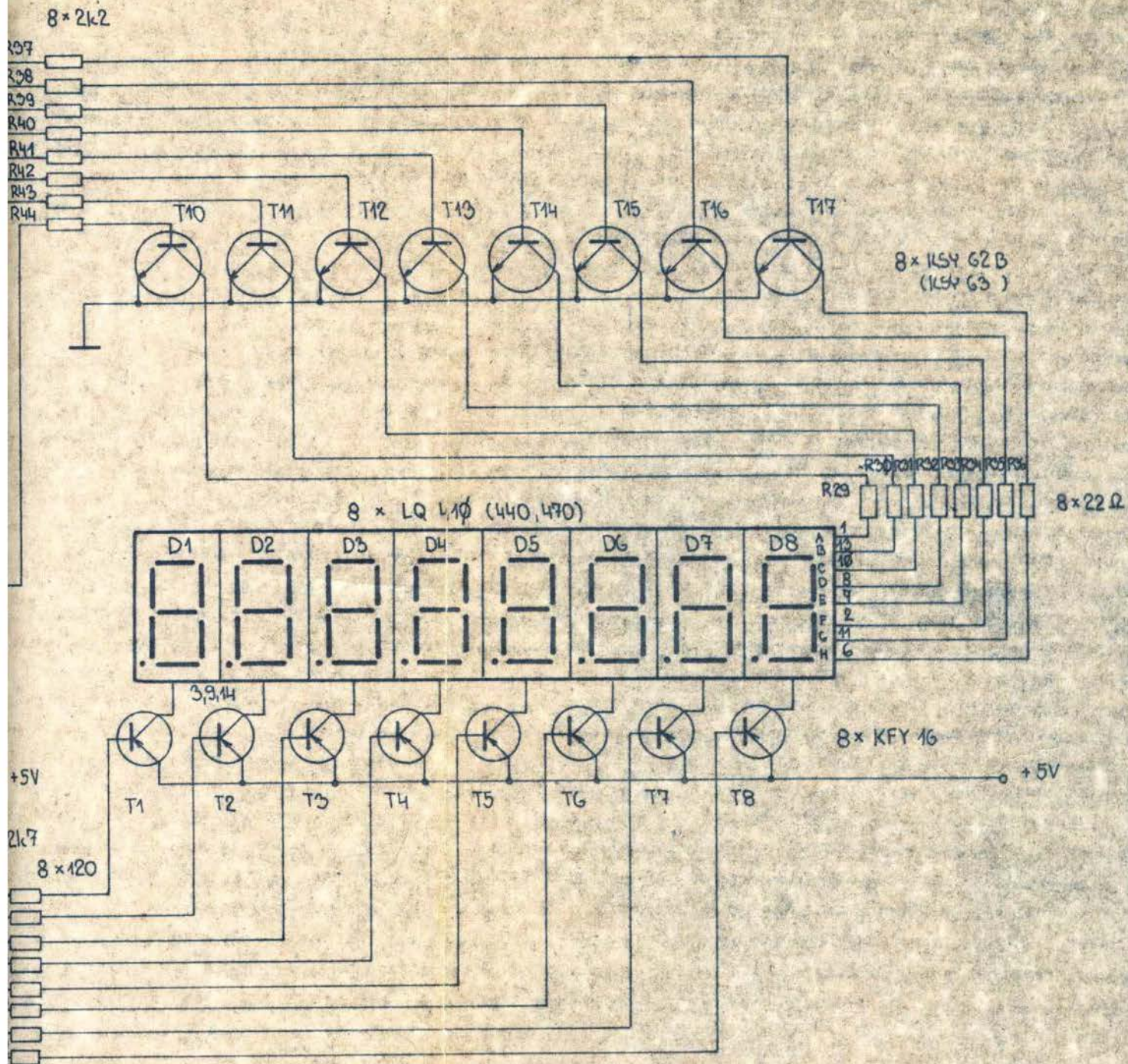




OBVOD	NAPAJANIE +5V VÝVOD:	ZEM (OV) VÝVOD:
8035	4φ, 26	2φ
3216	16	8
32φ5	16	8
3212	24	12
2716	21, 24	12
2114	18	9
8155	4φ	2φ
8243	24	12
74145	16	8
742φ	14	7
74φφ	14	7

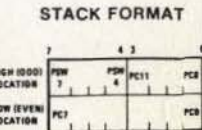
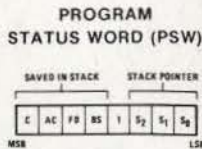
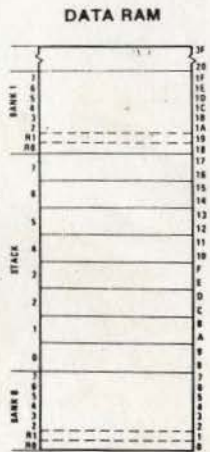
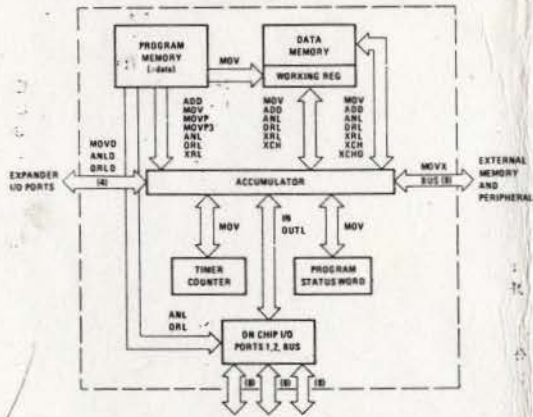






--- ROZŠÍŘENÝ SYSTÉM

MCS-48™ DATA TRANSFER INSTRUCTIONS



ACCUMULATOR

• ADD A,R <sub>r</sub>	6•
• ADD A,@R0	60
• ADD A,#data	03
• ADDC A,R <sub>r</sub>	7•
• ADDC A,@R0	70
• ADDC A,#data	13
ANL A,R <sub>r</sub>	5•
ANL A,@R0	50
ANL A,#data	53
ORL A,R <sub>r</sub>	4•
ORL A,@R0	40
ORL A,#data	43
XRL A,R <sub>r</sub>	D•
XRL A,@R0	D0
XRL A,#data	D3
INC A	17
DEC A	07
CLR A	27
CPL A	37
RL A	E7
• RLC A	F7
RR A	77
• RRC A	67
• DA A	57
SWAP A	47

DATA MOVES

MOV A,R <sub>r</sub>	F•
MOV A,@R0	F0
MOV R <sub>r</sub> ,A	F1
MOV A,#data	23
MOV R <sub>r</sub> ,A	A•
MOV @R0,A	A0
MOV R <sub>r</sub> ,A	A1
MOV R <sub>r</sub> ,#data	B•
MOV @R0,#data	B0
MOV R <sub>r</sub> ,#data	B1
XCH A,R <sub>r</sub>	2•

• = Carry Flag Affected

DATA MOVES (Cont.)

XCH A,@R0	20
R1	21
XCHD A,@R0	30
R1	31
MOV A,PSW	C7
• MOV PSW,A	D7
MOVX A,@R0	80
R1	81
MOVX @R0,A	90
R1,A	91
MOVDP3 A,@A	E3
MOVDP A,@A	A3

REGISTER

INC R <sub>r</sub>	1•
DEC R <sub>r</sub>	C•
INC @R0	10
R1	11
DJNZ R <sub>r</sub> ,addr	E•

FLAGS

• CLR C	97
• CPL C	A7
CLR F0	85
CPL F0	95
CLR F1	A5
CPL F1	B5

BRANCH

JMP addr	14
JMPP @A	B3
DJNZ R <sub>r</sub> ,addr	E•
JC addr	F6
JNC addr	E6
JZ addr	C6
JNZ addr	96
JT0 addr	36
JNT0 addr	26
JT1 addr	56
JNT1 addr	46
JF0 addr	B6
JF1 addr	76
JTF addr	16

: = See Table 2

Table 1. Register: Accumulator.

R <sub>r</sub>	MOV A,R	MOV R,A	XCH A,R	MOV R,#DATA	INC R	DEC R	DJNZ R	ADD A,R	ADDC A,R	ANL A,R	ORL A,R	XRL A,R
R0	F8	A8	28	B8	18	C8	8	68	78	58	48	D8
R1	F9	A9	29	B9	19	C9	9	69	79	59	49	D9
R2	FA	AA	2A	BA	1A	CA	A	6A	7A	5A	4A	DA
R3	FB	AB	2B	BB	1B	CB	B	6B	7B	5B	4B	DB
R4	FC	AC	2C	BC	1C	CC	C	6C	7C	5C	4C	DC
R5	FD	AD	2D	BD	1D	CD	D	6D	7D	5D	4D	DD
R6	FE	AE	2E	BE	1E	CE	E	6E	7E	5E	4E	DE
R7	FF	AF	2F	BF	1F	CF	F	6F	7F	5F	4F	DF

BRANCH (Cont.)

JNI addr	86
JB0 addr	12
JB1 addr	32
JB2 addr	52
JB3 addr	72
JB4 addr	92
JB5 addr	B2
JB6 addr	D2
JB7 addr	F2

TIMER

MOV A,T	42
MOV T,A	62
STRT T	55
STRT CNT	45
STOP TCNT	65
EN TCNTI	25
DIS TCNTI	35

CONTROL

EN I	05
DIS I	15
SEL RB0	C5
SEL RB1	D5
SEL MB0	E5
SEL MB1	F5
ENT0 CLK	75

SUBROUTINE

CALL addr	14
RET	83
RETR	93

• = Carry Flag Affected

NO OP

NOP	00
-----	----

INPUT/OUTPUT:

IN A,P1	09
OUTL P1,A	39
ANL P1,#data	99
ORL P1,#data	89
IN A,P2	0A
OUT L P2,A	3A
ANL P2,#data	9A
ORL P2,#data	8A

INS A,BUS	08
OUTL BUS,A	02
ANL BUS,#data	98
ORL BUS,#data	88
MOVD A,Pp	0:
MOVD Pp,A	3:
ANLD Pp,A	9:
ORLD Pp,A	8:

ASSEMBLER OPERATORS

(,)	
*, /, MOD, SHL, SHR	
+, -	
EQ, LT, LE, GT, GE, NE	
NOT	
AND	
OR, XOR	
HIGH, LOW	

: = See Table 2

Table 2. Input/Output.

Port	IN	OUT	AND	OR
BUS	08	02	98	88
P1	09	39	99	89
P2	0A	3A	9A	8A
P4	0C	3C	9C	8C
P5	0D	3D	9D	8D
P6	0E	3E	9E	8E
P7	0F	3F	9F	8F

Table 3. Branch.

Page	JMP	CALL
0	04	14
1	24	34
2	44	54
3	64	74
4	84	94
5	A4	B4
6	C4	D4
7	E4	F4

Page = 256 bytes

ASSEMBLER DIRECTIVES

ORG	Origin	MACRO	Define Macro
END	End program	ENDM	End Macro
		LOCAL	Define Local Symbol
DB	Define Byte		Block
DS	Define Storage	REPT	Define Repeat
DW	Define Word		Indefinite Repeat
		IRP	Indefinite Repeat
		IRPC	Indefinite Repeat
IF	Conditional Assembly		Character
ELSE	Assembly		Alternate Macro
ENDIF	End Conditional	EXITM	Exit
EQU	Set Symbol Value		

MASKA: 0-ANL #0  
1-OR: #1



Table 1. Register: Accumulator.

R <sub>r</sub>	MOV A,R	MOV R,A	XCH A,R	MOV R, #DATA	INC R	DEC R	DJNZ R	ADD A,R	ADDC A,R	ANL A,R	ORL A,R	XRL A,R
R0	F8	A8	28	B8	18	C8	E8	68	78	58	48	D8
R1	F9	A9	29	B9	19	C9	E9	69	79	59	49	D9
R2	FA	AA	2A	BA	1A	CA	EA	6A	7A	5A	4A	DA
R3	FB	AB	2B	BB	1B	CB	EB	6B	7B	5B	4B	DB
R4	FC	AC	2C	BC	1C	CC	EC	6C	7C	5C	4C	DC
R5	FD	AD	2D	BD	1D	CD	ED	6D	7D	5D	4D	DD
R6	FE	AE	2E	BE	1E	CE	EE	6E	7E	5E	4E	DE
R7	FF	AF	2F	BF	1F	CF	EF	6F	7F	5F	4F	DF

Table 2. Input/Output.

Port	IN	OUT	AND	OR
BUS	08	02	98	88
P1	09	39	99	89
P2	0A	3A	9A	8A
P4	0C	3C	9C	8C
P5	0D	3D	9D	8D
P6	0E	3E	9E	8E
P7	0F	3F	9F	8F

Table 3. Branch.

Page	JMP	CALL
0	04	14
1	24	34
2	44	54
3	64	74
4	84	94
5	A4	B4
6	C4	D4
7	E4	F4

Page = 256 bytes

ASSEMBLER DIRECTIVES

ORG	Origin	MACRO	Define Macro
END	End program	ENDM	End Macro
DB	Define Byte	LOCAL	Define Local Symbol
DS	Define Storage	REPT	Define Repeat Block
DW	Define Word	IRP	Indefinite Repeat
IF	Conditional Assembly	IRPC	Indefinite Repeat Character
ELSE		EXITM	Alternate Macro Exit
ENDIF	End Conditional		
EQU	Set Symbol Value		
Set			

MASKA: 0 - ANL #0  
1 - ORL #1

NO OP

NOP 00

INPUT/OUTPUT:

IN A,P1 09

OUTL P1,A 39

ANL P1,#data 99

ORL P1, #data 89

IN A, P2 0A

OUT L P2, A 3A

ANL P2, #data 9A

ORL P2,#data 8A

INS A, BUS 08

OUTL BUS, A 02

ANL BUS, #data 98

ORL BUS, #data 88

MOVD A,Pp 0:

MOVD Pp,A 3:

ANLD Pp,A 9:

ORLD Pp,A 8:

ASSEMBLER OPERATORS

(,)  
\*, /, MOD, SHL, SHR  
+, -

EQ, LT, LE, GT, GE, NE

NOT

AND

OR, XOR

HIGH, LOW

BRANCH (Cont.)

JNI addr 86

JB0 addr 12

JB1 addr 32

JB2 addr 52

JB3 addr 72

JB4 addr 92

JB5 addr B2

JB6 addr D2

JB7 addr F2

TIMER

MOV A,T 42

MOV T,A 62

STRT T 55

STRT CNT 45

STOP TCNT 65

EN TCNTI 25

DIS TCNTI 35

CONTROL

EN I 05

DIS I 15

SEL RB0 C5

SEL RB1 D5

SEL MB0 E5

SEL MB1 F5

ENT0 CLK 75

SUBROUTINE

CALL addr 14

RET 83

RETR 93

• = Carry Flag Affected

: = See Table 2