

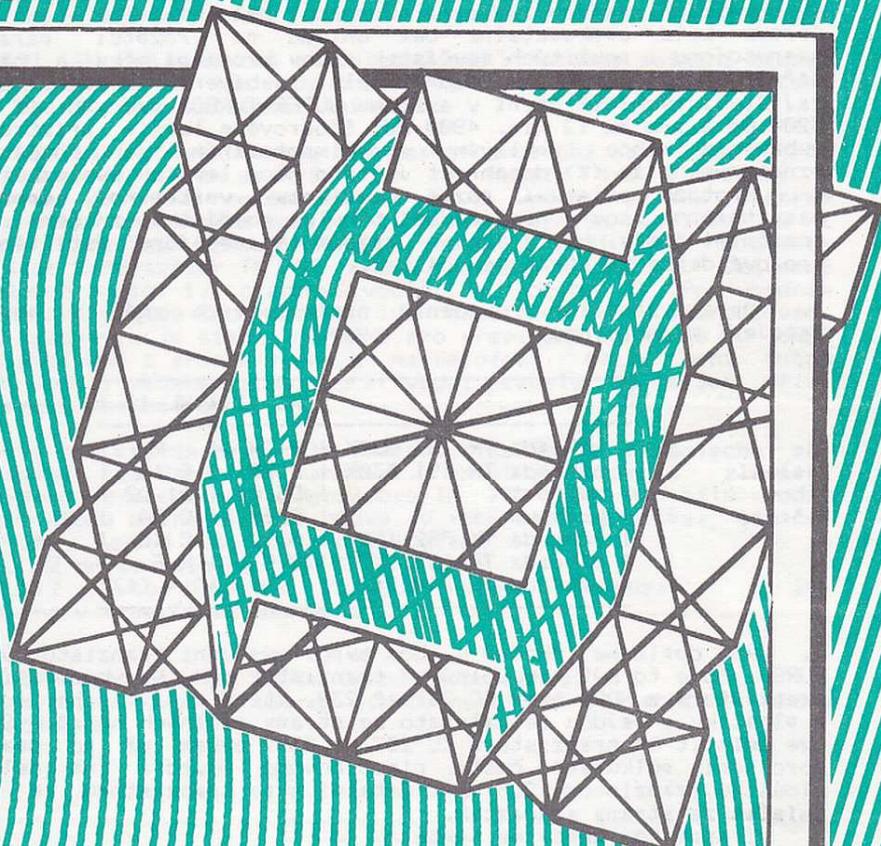


602

ATARI

3

89



Neboj se magnetofonu !

Pro šťastné majitele originálních kazetových magnetofonů ATARI typu XC 12 a XL 12, kteří se chtějí vlastními silami připojit k daval vyznavačů systému TURBO 2000, resp. SUPER TURBO, přinášíme stručný popis řešení jejich hlavního problému - úpravy magnetofonu.

Popisované řešení je výsledkem vlastních pokusů i úprav již existujících zapojení. Hlavní pozornost byla věnována tomu, aby úpravy byly opakovatelné bez ohledu na rozdíl parametrů magnetofonu i použitých součástek a aby kromě páječky a běžného nářadí nebylo zapotřebí žádné další vybavení. Uvedené úpravy zajišťují rychlosť čtení v systému SUPER TURBO pro XC 12 min. 5200 Bd a pro XL 12 min. 4900 Bd. Štourové s lepším přístrojovým vybavením mohou trpělivým experimentováním se součástkami označenými níže (*) dosáhnout ještě o něco lepších výsledků.

Protože existují různé varianty vnitřního uspořádání magnetofonů, jsou v následující tabulce součástky označeny svými pozičními čísly, která jsou v každém magnetofonu vytiskena na spojové desce ze strany součástek.

1. Úpravu zahájíme provedením následujících změn v původním zapojení magnetofonu :

	XC 12	XL 12
nahraď TR 191 330K R 24 R		
nahraď TR 191 330K R 38 R 18		
vymíout D3,D4 D1,D2		
vymíout C 6 C 24		
nahraď TK 782 150n C 12 C 13		
nahraď TR 191 2K7 R 6 R 6		
nahraď TK 744 2n2 - IC 4 (*)		

2. Nyní doplníme (shodně u obou typů) výstupní tranzistor úpravy TURBO. Může to být libovolný nf tranzistor NPN. Nejvhodnější jsou tranzistory z NDR, typy SC 237 až 239, které mají ploché pouzdro z plastiku a vejdou se načeště ze strany plošných spojů. Stejně lze umístit čs. tranzistory KC 237 až 239, ovšem až po opatrém sbroušení půlkulaté části plastikového pouzdra (na celkovou tloušťku tranzistoru 2,5 mm). Ostatní typy tranzistorů je nutno umístit ze strany součástek.

Tranzistor zapojíme takto :

- kolektor : spojit s kolektorem původního výstupního tranzistoru magnetofonu, tj. připojit jej na bílý vodič kabelu k počítači.
- báze : přes odpor TR 191 56K (*) připojit na vývod 8

int. obvodu magnetofonu.

- emitor : krátkým izolovaným vodičem připojíme na plošný spoj do místa, kam je přiveden oranžový vodič kabelu k počítači. Dále emitor propojíme přes diodu (katodu=červený proužek na emitor !) s vývodem 12 int. obvodu magnetofonu.

3. U obou typů magnetofonu propojíme vývody 9 a 10 int. obvodu přes odpor TR 191 68K. Možno použít i odpor vyjmutý z magnetofonu na začátku úpravy, označený barevnými proužky modrá - šedá - oranžová - červená.

4. U typu XL 12 ještě doplníme dva odpory :

- TR 191 27K mezi zem a bílý přívodní vodič ze zvukové stopy hlavy magnetofonu (označen w, je v rohu spojové desky v blízkosti C 9).

- TR 191 12K (*) mezi zem a společný bod přepínače snímání-záznam s kondenzátorem C 5 (je tedy připojován paralelně k datové stopě hlavy magnetofonu ve funkci snímání).

5. Poslední část úpravy je opět společná pro oba typy magnetofonu a týká se zástrčky na kabelu pro připojení k počítači. Zástrčku rozšroubojeme a z vnější strany opatrně "vypichneme" kontakty z pozice 11 (s oranžovým vodičem) a z pozice 7 (volný vedle hnědého vodiče). Mezi oba kontakty připojíme kondenzátor TK 782 47n a zasuňeme je zpět do vzájemně zaměněných pozic, tj. oranžový vodič do pozice 7. Po vhodném vytvarování vývodů kondenzátoru lze zástrčku opět zašroubovat. Tímto zapojením je signál COMMAND pro přepínání STANDARD - TURBO veden společně s příposlechem z magnetofonu, takže není nutný vnější vodič, vkusně zdobící přívodní kabely některých dříve upravených magnetofonů.

6. Pro ty, kterým by vadilo snížení hlasitosti příposlechu při přehrávání standardních nahrávek, existuje možnost zlepšení připojením keramického kondenzátoru TK 754 10p na bílý vodič zvukové stopy magnetofonové hlavy (v rohu spojové desky, označen w). Druhý vývod kondenzátoru připojíme :

- v XC 12 do společného bodu R 19 a R 22

- v XL 12 místo bývalého "živého" konce kondenzátoru C 24, vyjmutého v prvé části úpravy .

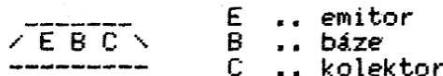
Ještě poznámky na závěr (experti prominou) :

- integrovaný obvod má vývody číslovány takto (při pohledu shora na součástky) :

14	8
- - - - -	-
D	1
- - - - -	-
1	7

4

- tranzistor v plastikovém pouzdro má vývody uspořádané takto (při pohledu zespodu na vývody):



Tranzistory v kovovém pouzdro mají u emitoru na pouzdro výstupek, pořadí vývodů je shodné.

- páječka se připojuje vidlicí do běžné síťové zásuvky, pájí se druhým koncem takto vytvořené soustavy (pozor, pálí to !).

- potřebné součástky jsou k dostání (prý skoro pořád) v odborných prodejnách podniků TESLA ELTOS a Domácí potřeby dle tohoto seznamu :

odpory	TR 191	2K7	1 ks
		12K	1 ks (pouze XL)
		27K	1 ks (pouze XL)
		56K	1 ks
		68K	1 ks
		330K	2 ks
kondenzátory	TK 744	2n2	1 ks (pouze XL)
	TK 782	47n	1 ks
		150n	1 ks
	TK 754	10p	1 ks (pouze pro náročné
dioda KA 261 až KA 265			1 ks
tranzistor (viz text)			1 ks

Přeji všem hodně štěstí !!

Honza Vyskoč, ATARI Klub Plzeň

KONEKTOR SBĚRNICE ATARI 800 XL

K počítačům ATARI 800 XL a 130 XE se poměrně velmi snadno připojují vnější přídavná zařízení. Mají jeden osmibitový vstupní výstupní port, 4 analogové digitální převodníky, sériový vstup-výstup. Dále jsou vybaveny konektorem pro připojení cartridge a konektorem paralelní systémové sběrnice. To vše dává amatérským stavitelem přídavných zařízení velké možnosti. Téměř všichni však svá přídavná zařízení připojují na joystickové porty. Máme-li takových zařízení více, musíme je stále přepojovat. A přitom existuje velmi jednoduchý způsob, jak si rozšířit počet portů na celkem 4 osmibitové připojit na sběrnici obvod MHB 8255A. Způsob byl dostatečně popsán ve zpravodaji č. 4 pražského ATARI klubu. Pro ty, kteří si to chtějí zkoušit přinášíme na straně 14 schema konektoru sběrnice ATARI 800 XL.

U Z I V A T E L S K E P R O G R A M Y

ENGLISH TUTOR

Program byl vytvořen za účelem procvičování anglických slovíček, idiomů a mluvnických jevů. Protože tento program je určen pro práci s kazetopáskovou pamětí a počítačem, je nutno řešit otázku omezené vnitřní paměti. Problém je řešen členěním výuky do lekcí, které jsou uloženy na kazetě odděleně od programu. Před vlastním zahájením procvičování je nutné do počítače nahrát interpret Basicu TBASIC DOS, potom program "Analictina" a nakonec data příslušné lekce.

Vlastní práci s programem zahájíme příkazem RUN. Po několika sekundách se na obrazovce objeví základní menu:

Select function:

Save date	1
Load date	2
Write date	3
Correct date	4
Exercise	5
Activity	6
End	7

Choice:

Z voleb menu je zřejmé, že program může pracovat ve dvou základních režimech:

1. vytváření nové lekce a její uložení na kazetu
2. Přečtení hotové lekce z kazety a její procvičování.

První režim zahájíme vytvořením tabulek otázek, odpovědí a výslovnosti. Docílíme toho volbou funkce 3. Na obrazovce se objeví zpráva: "Write number of lesson". Po zadání čísla lekce a jeho odeslání příkazem RETURN se vypíše zpráva "Write a question". Po odeslání otázky RETURNem se zobrazí zpráva "Write an answer". Po odeslání odpovědi požádá počítač o zapsání výslovnosti zprávou "Write a pronunciation". Tento cyklus se opakuje. Při zadávání otázky je možno zadat dvě speciální hodnoty, které se nepřenesou do tabulky, ale ovládají určité pomocné činnosti. Jsou to "*", která signalizuje konec zadávání dat a návrat do základního menu, a "+", které zvýší číslo vypisované za oznamem "Seavence". Jeho význam bude vysvětlen později.

Vlastní vytváření lekce probíhá tak, že jako otázku zadáváme

úkol, který chceme procvičovat. Jako odpověď zadáme vzorový tvar odpovědi a jako výslovnost popis správné výslovnosti obtížných slov. Dole na obrazovce se nám objevují následující zprávy: "délka polí C=.. A=.. V=..", které udávají délku jednotlivých tabulek a počet záznamů. Každý záznam může být dlouhý maximálně 100 znaků. Celková délka tabulek nemusí překročit u otázek a odpovědí 5000 znaků, u výslovnosti 3000 znaků. V jedné lekci smí být maximálně 200 záznamů. Další omezení spočívá v tom, že poslední záznam není využit při procvičování a je proto nutné jej napsat dvakrát!

Po návratu do menu musíme zapsat činnosti, které se s jednotlivými skupinami otázek (stejné číslo sekvence) mají provádět. Toho dosáhneme volbou funkce 6. Na obrazovce se vypíše "Entry Activity or Correct". Po zadání "E" nás počítač požádá o zápis jednotlivých činností. Volba "C" nám umožnuje upravovat dříve zadané činnosti. Zadávání i opravy ukončíme volbou "*".

Protože obvykle nezapišeme celý soubor otázek a odpovědí bezchybně, je možné záznamy v hotové tabulce opravovat. Režim oprav zvolíme volbou "4". K chybnemu záznamu se dostaneme opětovným stlačováním klávesy RETURN. Zde místo RETURN odmáčkneme "C" a budeme dotázání, které záznam chceme přepisovat (otázka, odpověď, výslovnost). Po zapsání a odeslání příslušného čísla, počítač požádá o nové znění záznamu. Chvilku počkáme. Nyní jsme dotázání, zda chceme ukončit opravy (E), nebo znova prohlížet záznamy (RETURN).

Poslední činností při vytváření lekce je její zápis na kazetu. Provedeme to volbou funkce 1.
(Poznámka: 1. Pokud chceme program ukončit bez uložení vytvořené lekce, vypíše se při volbě "7" upozornění a program se vrátí do menu. Práce se ukončí až při opakování volbě "7". 2. Protože otázky jsou psány většinou česky a výslovnost vyžaduje speciální fonetické znaky, jsou tyto znaky nadefinovány místo graf. symbolů.)

Druhým režimem práce programu je procvičování dříve vytvořených záznamů. Po odstartování programu nahrajeme do paměti volbou "2" lekci, kterou chceme procvičovat. Po nahrání dat do paměti počítače se na několik sekund objeví na obrazovce číslo nahrané lekce. Vlastní procvičování zahájíme volbou funkce "5". Počítač na obrazovku označí činnost a vypíše otázku. Čeká na vstup odpovědi. Po jejím zapsání a odeslání, provede její porovnání se vzorovou odpovědí. Při porovnávání počítač toleruje rozdíly mezi malými a velkými písmeny a v počtu oddělovacích mezer. Po vyhodnocení odpovědi pokračuje buď vypsáním OK, nebo v případě chyby správné odpovědi s výslovností. Po odmáčknutí pokračuje v zadávání dalších otázek. Na spodní řádce obrazovky se vypisují zprávy o výsledku procvičování a dotaz, jestli chceme v

procvičování pokračovat. Algoritmus zadávání otázek je vytvořen tak, že otázky, které byly chybně zodpovězeny se zadávají s větší frekvencí než otázky zodpovězené správně.

Program je zabezpečen proti takové posloupnosti akcí, která by způsobila chybu. Nevydává však o tom na obrazovce žádné zprávy. Protože zdánlivá nečinnost programu může vyvolat rozpaky uživatele, uvedu její nejčastější příčiny:

- pokus uložit data, aniž by byly zadány činnosti
- pokus procvičovat lekci, aniž byly zadány činnosti
- pokus o zápis prázdných tabulek
- pokus opravovat prázdnou tabulku.

Ing. Pavel Tomek

ANTIC MAKER 2

ANTIC MAKER 2 je program pro tvorbu programů v jazyku symbolických adres grafického procesoru ANTIC, tj. program pro tvorbu DISPLAY LISTŮ. Umožňuje vytvořený DL uložit na výstupní zařízení pro pozdější použití.

Po nahrání a spuštění programu má uživatel k dispozici čtyři volby v základním menu:

1. ASSEMBLER
2. DISASSEMBLER
3. HELP
4. CASARSOFT INFO

Volbou "3" si můžeme vyvolat základní informace o použití a obsluze programu ANTIMAKER 2.

Volba "4" nám zobrazí informace o autorovi programu.

Volba: 1. ASSEMBLER

Tento volbou přejde program do režimu assembleru. Zobrazí se otázka "ORG ?". Počítač se ptá na počáteční adresu DL. Po jejím zadání se vykreslí pracovní část obrazovky, informační řádek a program očekává zadání instrukce DL. Ihned po zadání instrukce je v pravé části obrazovky vypsán její kód a aktualizován informační řádek. Je očekávána další instrukce. Psaní instrukcí DL ukončíme pseudoinstrukcí "END". Program položí bezpečnostní otázku, zda opravdu chceme ukončit psaní instrukcí a po kladné odpovědi přejde do dalšího menu.

1. Save display list
2. GO 1.st menu
3. Control list DL
4. End of work

Pomocí volby "1" uložíme vytvořený DL na magnetické medium pro pozdější použití.

Volbou "2" se vrátíme do základního menu.

Volba "3" spustí kontrolní výpis vytvořeného display listu na obrazovku.

Pro ukončení práce s programem ANTIC MAKER 2 a návrat do BASICu zvolíme "4".

Instrukce assembleru

- G xx - xx je kód grafického modu shodný s Basicem. Při překladu je automaticky převeden do kódu ANTIC. Mezi "G" a "xx" je povinná mezera.

- A xx - xx je kód grafického modu přímo v kódu ANTIC. "xx" musí být vždy dvojciferné (např. "14", nebo "02"). Mezi "G" a "xx" je opět povinná mezera.

- A'xx,h,v,l,d - xx je ANTIC-kód grafického modu,
 - h je FHS, tj. jemný horizontální scroll,
 - v je FVS, tj. jemný vertikální scroll,
 - l je LMS, určení videoram,
 - d je DLI, tj. nastavení přerušení DL.

- BL x - (Blank line) - vytvoří x prázdných (mrtvých) řádků. "x" může nabývat hodnot z intervalu <1,8>. Mezi "BL" a "x" je povinná mezera.

- JMP adr - nepodmíněný skok na adresu "adr". Mezi "JMP" a "adr" je povinná mezera. Adresa se zadává decimálně.

- JVB adr - skok na adresu "adr" podmíněný synchronizačním pulsem obrazovky. Pro adresu platí stejná pravidla, jako u instrukce "JMP".

- END - tato instrukce se nepřekládá. Slouží jako pseudoinstrukce pro přechod do druhé nabídky.

Volba: 2. DISASSEMBLER

Také v tomto režimu může uživatel volit z nabídky pěti možností:

1. Load and disassembly display list
2. Disass. standard DL
3. Disass. DL in memory
4. Disass. work DL
5. GO 1.st MENU

Volbou "1" až "4" zvolíme zdroj dat pro disassembler. V "1" volíme vnější zařízení, v "2" standardní display list, po "3" program načte data z paměti a "4" načte data z pracovního souboru. Po načtení dat začne zpětný překlad do mnemonických značek. Tyto značky jsou shodné s mnemonikou (instrukcemi) assembleru. Je-li v datech nepřeložitelný kód, je vypsáno hlášení "Err" a chybný kód. Zároveň se ozve zvukový signál. Ukončení překladu je oznámeno zprávou "O.K.".

Volbou "5" se vrátíme do základního menu.

Lukáš ČASAR

S L A B I K Á R A T A R I S T Y

TEXT V GRAFICKÝCH MODECH (1)

=====

Široké grafické možnosti bývají vedle možnosti zvukových označovány za jednu z největších předností osmibitových počítačů ATARI. Patnáct druhů grafických režimů "zabudovaných" a možnost vytvořit si prakticky jakýkoliv grafický mod pomocí tvorby vlastních display-listů je opravdu velkou a ojedinělou předností. Bohužel však u většiny grafických modů není možné jednoduchým způsobem na obrazovce vypsat jakýkoli text. K tomu jsou určeny speciální textové mody, z nichž jsou prakticky použitelné pouze 3: GR.0, GR.1 a GR.2. V těchto režimech však je naopak zase dost složité cokoli namalovat. Chceme-li v modu GR.8 vykreslit nějaký graf a popsat jej, musíme sáhnout buď po jiném programovacím jazyku (např. TURBOBASIC XL - má příkaz TEXT), nebo napsat k vlastnímu programu pomocné rutiny, které nám umožní vypsat text i v grafickém prostoru. Tento článek se bude zabývat různými způsoby řešení právě takových rutin. A začneme tím nejjednodušším.

Text v GR.8

Zobrazit text v grafickém prostoru GR.8 je opravdu nejjednodušším případem. Tento grafický mod je totiž velice podobný textovému modu GR.0. Naše pomocná rutina bude provádět to, co v případě GR.0 provádí operační systém počítače. Tedy převezmě ASCII kód aktuálního znaku, podle tohoto kódu určí kód interní, podle interního kódu vypočítá adresu v paměti ROM, na které je uložen 1. bajt znaku. Dále podle zadané pozice určí

adresu ve videopaměti a sem postupně z paměti ROM přenese 8 bajtů, tvořících zobrazovaný znak.

V dále uvedeném programu je na řádcích 28000 až 28030 rutina, která provádí všechny zmíněné činnosti. Před jejím použitím je nutné v hlavním programu dimenzovat a naplnit proměnnou VETA\$, dále pomocí proměnných POSX a POSY určit polohu textu na obrazovce. POSX může nabývat hodnot 0 až 39 (jako u GR.0), POSY 0 až 159 (u GR.8), nebo až 191 (u GR.8+16).

```

TW 10 REM *** PISMO V GR.8 - DEMO ***
EV 20 DIM VETA$(120):GRAPHICS 8:COLOR 3
OV 30 POSX=0:POSY=0:VETA$="PISMO":GOSUB 2
     8000:POSX=6:POSY=4:VETA$="v grafickem"
     :GOSUB 28000:POSX=18:POSY=9
QE 40 VETA$=[ MODU c. 8 ]:GOSUB 28000
HK 50 POSX=0:POSY=64:VETA$="šířku titulu ještě"
     :GOSUB 28000
LE 60 POKE 756,204:POSX=0:POSY=72:GOSUB 2
     8000
AF 70 STOP
EK 28000 MEM2=256*PEEK(756):MEM1=MEM2+512
     :MEM3=MEM2+768:VIDEO=PEEK(88)+256*PEEK
     (89)
ML 28010 FOR POCET=1 TO LEN(VETA$):ADRESA
     =40*POSY+POSX+VIDEO:KOD=ASC(VETA$(POCE
     T)):INV=(KOD)127:KOD=KOD-INV*128
AK 28020 MEM=(KOD(<=31)*(MEM1+8*KOD)+(KOD)
     31 AND KOD(<=95)*(MEM2+8*(KOD-32))+(KOD
     >95)*(MEM3+8*(KOD-96)))
IO 28030 FOR PAR=0 TO 7:POKE (ADRESA+40*P
     AR),(ABS(255*INV-PEEK(MEM+PAR))):NEXT
     PAR:POSX=POSX+1:NEXT POCET:RETURN

```

Na řádku 28000 je určena adresa znakové sady v ROM (MEM2), pomocné adresy v ROM (MEM1 a MEM3) a adresa počátku videopaměti (VIDEO). Řádek 28010: pro každý znak z proměnné VETA\$ je určena adresa ve videopaměti (ADRESA), na kterou bude uložen 1. bajt z ROM. Dále je určen ATASCII kód znaku (KOD) a zjištěno, zda je inverzní (INV=1). V tom případě je kód znaku převeden na normální. Řádek 28020: na tomto řádku je určena adresa prvního bajtu znaku v paměti ROM podle interního kódu. Řádek 28030: zde je proveden přenos 8 bajtů z ROM do videopaměti tak, aby se ukládaly pod sebe (s krokem 40 bajtů). Pro INV=1 je provedena nesace. Po zobrazení celého znaku je zvýšena o 1 proměnná POSX. Stejným způsobem budou zobrazeny všechny další znaky a potom se provede návrat do hlavního programu.

Na řádcích 10 až 70 je krátce demonstrováno použití pomocné rutiny a její možnosti. Před přepisováním do počítače přepněte na mezinárodní sadu znaků (POKE 756,204), před odstartováním programu vraťte normální sadu (POKE 756,224).

ZE ZAPISNIKU PROGRAMATORA

SOUND LOADER

Nejprve chci poděkovat autorovi tohoto programu (jemuž anonymita neubírá zásluh), že nám umožnil hrátky se zvukem.

Program SOUND LOADER umí za pomocí tvarovače signálu pro TURBO 2000 "digitalizovat" akustický signál a pak ho reprodukovat ze záznamu v paměti. Zvuk není sice příliš věrný, nicméně hudba i mluvené slovo jsou srozumitelné.

Použití programu.

Po natažení do počítače se ozve zabzučení. Do datového magnetofonu vložte běžnou zvukovou kazetu se záznamem, který chcete digitalizovat. Stiskněte obvyklé PLAY+RETURN. Program nahrává po dobu asi 25 sekund (přepne interface v magnetofonu na TURBO a spustí motorek). Po zaplnění paměti digitalizovaným signálem se magnetofon zastaví a počítač vás vyzve dvojím zabzučením ke stisku RETURN a k poslechu. Novou nahrávku lze pořídit po stisku RESET.

Popis programu.

Zdrojový text programu SOUND LOADER je uveden na dalších stránkách. Můžete jej přepsat do vhodného assembleru, např. Editor Assembler, či MAC 65. Je možné použít i jiné assemblery (ATMAS II), avšak budete muset změnit mnemokódy pseudoinstrukcí. Celý zdrojový text je bohatě komentován a jeho pochopení by zkušenějším programátorem ve strojovém kódě nemělo činit potíže.

Rutiny programu využívají přímé čtení stavu sériového portu. Významový 4.bit se postupně ukládá na adresu \$0400 rotací. Po záznamu 8.bitu se obsah celého Bytu ukládá do paměti. Pro data je využita oblast \$0700 až \$BFFF. Během činnosti jsou zakázána všechna přerušení a postupně je přemazáván obsah obrazovky a DISPLAY LIST. Proto se nelekňte nesmyslu, který spatříte, po nahrání zvuku. Normálního zobrazení lze dosáhnout jen RESETem.

Námětem k zamýšlení je vytvoření komfortního programu pro digitalizaci a reprodukci zvuku s možností záznamu dat v systému TURBO. Některé rutiny je možno uložit do CRSBUF (od \$0400), jiné do zásobníku (1.stránka paměti). Vlastní program lze umístit do RAM pod operační systém.

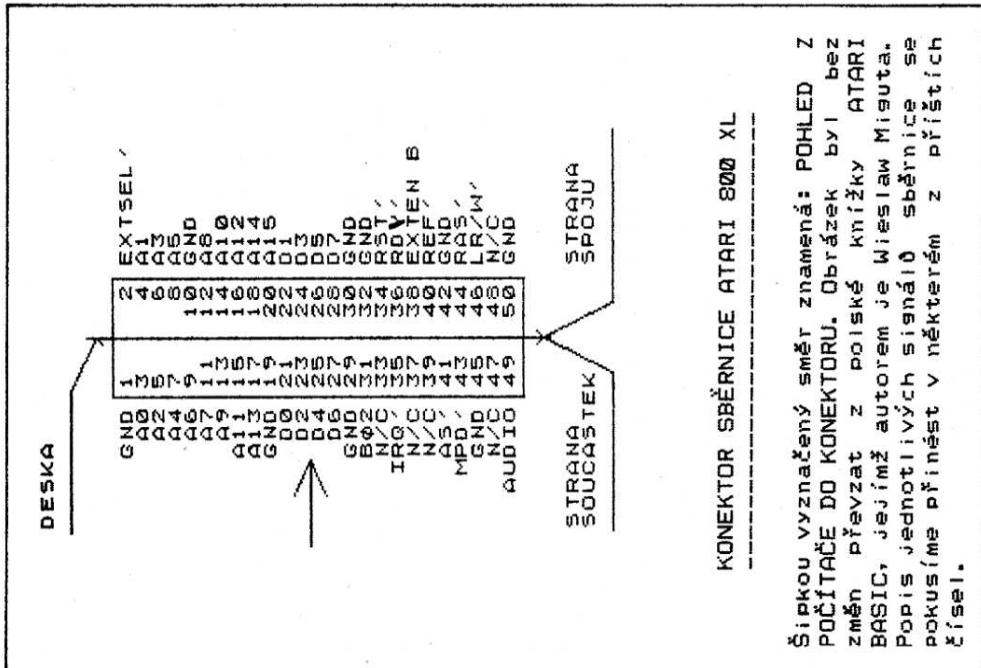
Hodně zábavy !

-OK1UXX-

* * *

1000 ;	SOUND LOADER	LDX #30	1370	adresa nazvu
1010 ;		LDY #405	1388	iprogramu
1020 ;	program pro digitalizaci zvuku s pouzitim	JSR \$C542	1398	typis nazvu programu
1030 ;	stvarovace TURBO 2000	LDA #\$81	1408	icekani na stisk libovolne
1040 ;		JSR #TFC	1418	iklavesy pri vystupu pomocí C10
1050 ;	#= \$0505 ipocatek programu	JSR #TFC	1428	iklavesy pri vystupu pomocí C10
1060 ;		JSR JML	1438	digitalizace
1070 ;	nejprve instalace do systemu pri inicializaci	LDA #\$82	1448	ipocatecky nastaveni
1080 ;		JSR #TFC	1458	icekani na stisk libovolne
1090 ;		JSR JMS	1468	iklavesy pri vystupu pomocí C10
0000 ;	0505 ASCI	JMP JMS	1478	ipriprav dalsi reproduksi
1000 ;	LDA #3C	1488 ;		
1100 ;	STA #D302	1498 ;		
1110 ;	STA #D303	1508 ;		
1120 ;	STA #D308	1518 ;		
1130 ;	LDA #8P2	1528	JML	inluji hizsi brite
1140 ;	LDA #095	1538	STA #32	icitace palesti
1145 ;	STA #095	1548	STA #34	ia koncové adresy
1150 ;	LDA #8P0	1558	STA #35	ipocatecky adresy ukladani zvukovych
1155 ;	LDA #8P2	1568	LDA #8P7	idat \$0700 (vysssi brite)
1160 ;	LDA #8P2	1578	STA #35	koncová adresa
1170 ;	LDA #8P5	1588	RTS	(posledni data na \$FFFF)
1180 ;	LDA #8P5	1598 ;		
1190 ;	STA #053	1608 ;		
1195 ;	LDA #FFF	1618 ;		
1200 ;	STA #D301	1628 ;		
1210 ;	LDA #8P0	1638 ;		
1220 ;	STA #D244	1648 ;		
1230 ;	QLC	1658 ;		
1240 ;	RTS	1668 ;		
1250 ;		1678	JML	inavrat
1255 ;	#= \$0530	1688	STA #D2E	z inicializace
1260 ;		1698	STA #D4E	inavrat z inicializace
1265 ;		1708	JSR JMS	inavrat z inicializace
1270 ;		1718	LDA #8C8	skutecky start programu
1275 ;		1728	STA #D4E	
1280 ;	.BYTE \$D0,\$53,\$EF,\$75,\$E,\$54	1738	STA #19	
1285 ;	.BYTE \$2B,\$6C,\$EF,\$61,\$64,\$5	1748	STA #D2E	
1290 ;	053534 6E64	1758	JSR JMS	
1295 ;	053534 6A65	1768	JML	ipocatecky nastaveni
1300 ;		1778	STA #D4E	
1310 ;		1788	STA #D4E	
1320 ;	#= \$0530	1798	JSR JMS	ipocatecky nastaveni
1330 ;		1808	JML	
1340 ;		1818	STA #19	
1350 ;		1828	STA #D2E	
1355 ;	JSR JML	1838	JSR JMS	ipocatecky nastaveni
1360 ;		1848	JML	
1365 ;		1858	STA #19	
1370 ;		1868	STA #D2E	
1375 ;		1878	JSR JMS	

05013	SH06203	1750	STA \$0392	zastav motor	STA \$0393	2140	začni TURBO
05015	SH06303	1770	STA \$0393	inštav command	LDA #0000	2150	typu vsechna přerušení
05019	SH06303	1780	RTS		05388 A900	2150	!POMSK
		1790 ;			0538A 8510	2150	STA \$10
		1800 ;		Cteni zvukovych dat	053C 80E722	2170	STA \$02AE
		1810 ;			053F 80E744	2180	!TREDEN
05019	A9000	1820	JMP LDA #0000	inuluž zasobnik	0532 80E804	2190	STA \$0400
0501C	SH06004	1830	STA \$06000		0535 20E9065	2200	JSR JH10
0501F	R2088	1840	LDX #0000	!opakuj 8-krat:	0538 6900	2210	LDA #0000
05E1	A00F1D2	1850	JMP LDA \$1020F	!cteni seriového portu	053A 80EAD4	2220	STA \$040E
05E4	2910	1860	AND \$10	!vyber jen 4.bit (prime čteni)	053D 8510	2230	STA \$10
05E5	18	1870	CLC	inuluž přenos	053F 80EAD2	2240	STA \$02AE
05E7	20001	1880	BNE JH5	!iskok pro 4. bit nastaven ("1")	0542 A93C	2250	LDA #43C
05E9	38	1890	SEC	inštav přenos	0554 80E9213	2260	STA \$0392
05EA	2E004	1900	JMP RTL \$0A0000	!odrotuž přenos (carry) do zasobniku	0557 80E9303	2270	STA \$0393
05ED	A910	1910	LDY \$10	!doba cekani	055A 68	2280	RTS
05EF	88	1920	JMP DEY	!cekej (opakuj prazdny		2290 ;	Vystup zvukovych dat
05F1	D9F7	1930	BNE JH5	!cyklius \$10 (16) kрат		2310 ;	
05F2	CA	1940	DEX	!pobr zasobniku az naphnis cely		2320	JM10 LDA (\$32), Y !data-byte z pameti
05F3	D9EC	1950	BNE JH4	!pobr zasobniku (8-bit)atoru	05AB B132	2330	STA \$0A000
05F5	A000000	1960	LDA \$060000	!zasobnik do akumulatoru	05B0 80E904	2340	LDX #0000
05F8	9132	1970	STA (\$32), Y	!akumulator do pameti	05B2 6208	2350	!onakuj 8-krat
05FA	E532	1980	INC \$32	!nizsi byte ukazovatka pameti:	05B3 8510	2360	!oarotuž databit do prenosu (carry)
05FC	D9B2	1990	BNE JH7	!presioc,neni-ii \$32 nulovy	05B7 9802	2370	!nastav \$, bit v akumulatoru
05FF	E533	2000	INC \$33	!zvys vissi byte ukazovatka pameti:	05B9 A91F	2380	!presioc,neni-ii prenos
05008	A532	2010	JMP LDA \$32	!stromnej nizsi byte	05B8 80E9102	2390	JH12 STA \$02001
05012	C534	2020	CMP \$34	!s koncovou adresou	05B8 E910	2400	LDY \$10
0504	D9B7	2030	BNE JH8	!presioc, nenii-ii konec	05C0 88	2410	JH13 DEY
05065	F533	2040	LDA \$33	!stromnej vissi byte	05C1 D9F7	2420	BNE JM13
05088	C535	2050	CMP \$35	!s koncovou adresou	05C3 CA	2430	DEX
0509A	D901	2060	BNE JH8	!presioc, nenii-ii konec	05C4 D9E1	2440	BNE JM11
0509C	60	2070	RTS	!phna pamet	05C5 E932	2450	INC \$32
050D	A00005	2080	JMP JH5	!cteni dalšího bytu	05C8 D902	2460	BNE JM14
		2090 ;			05CA E933	2470	INC \$33
05E18		2100	==	\$065386 ; Reprodukce	05CC F532	2480	JM14 LDA \$32
		2110 ;			05CE C534	2490	CMP \$34
05E88	A914	2120	JMP LDA \$34	;dec 52	05D0 D907	2500	BNE JM15
05E82	SH06203	2130	STA \$0392	!spust motor	05D2 A933	2510	LDA #33



KONEKTOUR SBERNICE ATARI 800 XL

**Šípkou vyznačený směr znamená: POHLED Z
POČÍTAČE DO KONEKTORU.** Obrázek byl bez
změn převzat z polské knížky ATARI
BASIC, jejímž autorem je Wiesław Misuta.
Popis jednotlivých signálů sběrnice se
pokusíme přinést v některém z příštích
čísel.

Výše uvedený výpis programu SOUND LOADER je zároveň ukázkou práce s programu TEXGEN 1.0, jehož popis byl v minulém čísle zpravodaje. Tímto programem pořízený zdrojový text byl dále pomocí programu "ATMAS-MAC/65" převeden do formy, kterou vyžaduje makroassemblyr MAC/65. Pomocí editoru byly přidány komentáře, převzity řádky a nakonec příkazem "ASM ,#P" získán uvedený výpis. Cílový kód, nahrazený do počítače přes získáte příkazem "ASM ,#C".

Poznámka redakce:

05E4 0535	2526	CMP #55		05A8 JM10	05B2 JM11
05D6 D901	2538	DNE JM15		05C0 JM13	05D9 JM15
0538 69	2548	RTS		05D1 JM14	05EA JM5
05D9 AC9E05	2550	JM15 JNP JM10	istovnej s koncovou adresou	05E1 JM4	05F8 JM9
	2558	; VSTUP probina pres QUDC registr (\$02281)		05D0 JM8	
	2570	4.bit=1 vstup			
	2580	0,-3-bit=1 nastost			
	2590	tj. \$10=1 ticho			
	2600	\$1F=1 max. nastost			
	2618	;			
*** ASSEMBLY ERRORS: 0 22168 BYTES FREE					
PAGE 4					
SYMBOLS					
0592 JM0	0598 JM1	05A8 JM10		0592 JM10	
05B8 JM12	06C0 JM13	05CC JM14		05D9 JM15	
05AF JM2	05DA JM5	05E1 JM4		05EA JM5	
05EF JM6	05F8 JM7	05F0 JM8		05F8 JM9	

TOS a zrychlené programy v BASICu

Většina uživatelů systému TURBO 2000 jistě zná program "ZRYCHLOVAC", který převádí pro TURBO programy vytvořené v jazyce BASIC a uložené ve standardním formátu na kazetu příkazem CSAVE

Problémy s takto zrychlenými programy nastávají, chceme-li je spustit pod některým kazetovým TURBO operačním systémem (dále TOS), abychom např. využili výhod zařízení "T:" (resp. "D:") nebo instalované nestandardní tiskárny (např. BT-100). Programy zpracované zrychlovačem jsou ukládány a spuštěny od adresy \$0700 (hex). Pokus o jejich zavedení pod řízením TOSu vede k chybovému hlášení "ADR ERROR". TOS tak brání přemazání oblasti, ve které je sám uložen (zpravidla od \$0B00 do \$1FFF) nebo kterou používá jako buffer pro TURBO data (od \$0700).

Naštěstí programy zpracované zrychlovačem obsahují před původním programem v BASICu ještě přídavnou strojovou rutinu, která po malé modifikaci umožňuje jejich uložení a spuštění od jiného vhodného místa v paměti. Je ovšem nutno současně odpovídajícím způsobem změnit hodnoty ukazatelů v tabulce na počátku programu v BASICu.

Pro nové umístění zrychlených programů byl zvolen prostor od adresy \$2000, tj. těsně nad TOS (s rezervou pro jeho případné rozšíření). Popsaný postup je však zcela obecný a s jinou hodnotou přičítané konstanty může sloužit k posunutí zrychlených programů do libovolného místa.

Dále uvedené příkazy se týkají použití monitoru TM 2003, lze však použít ekvivalentní příkazy jiného monitoru.

Postup přesunu zrychleného programu systému TURBO 2000 v jazyce BASIC monitorem TM 2004 :

1. H - HEX
(volba hexadecimální aritmetiky)
2. I - TLOAD
FROM: 2000
(přejetí původní hlavičky programu)
3. I - TLOAD
FROM: 2000
po ukončeném nahrávání vypíše monitor
BYTES: délka - zapamatovat, bude zapotřebí !
(nahrání původního zrychleného programu do paměti na nové místo)
4. M - MONITOR
FROM: 2000

nyní změnit obsah následujících adres přičtením konstanty
\$19 = \$20-\$07 k původnímu obsahu adresy :

\$2004 , \$20F3 , \$20F5 , \$20F7 ,
\$20F9 , \$20FB , \$20FD , \$20FF

režim monitoru ukončit klávesou BREAK
(posunutí všech ukazatelů od stránky \$20)

5. T - TURBOSAVE

FROM: 2000

HEAD

NAME: jméno programu.

FROM: 2000

BYTES: původní délka programu.

START: 2000

(uložení upraveného programu na kazetu)

6. R - TURBOSAVE

(případné opakování uložení)

Na závěr ještě převod dôležitých čísel do desítkové soustavy
pro ty, kteří nemají rádi soustavu hexadecimální a vynechají bod
uvedeného postupu :

\$700 = 1792 , \$2000 = 8192 , \$19 = 25

Upravené adresy :

8196 , 8435 , 8437 , 8439 ,
8441 , 8443 , 8445 , 8447

Přeji příjemnou zábavu.

(H.V.)

TERMINY SCHŮZEK: Pravidelné schůzky členů našeho klubu se konají
ve Výcvíkovém středisku branců, Pod Juliskou 2, Praha 6,
sobotu, každý sudý týden, od 13:00 hod. Tyto schůzky jsou určeny
všem členům klubu, tedy pražském i mimopražském! Nepravidelné
schůzky - přednášky bývají v kinosále Národního technického
muzea, Kostelní ulice, Praha 7 - Letná. Terminy těchto schůzek
bývají uvedeny v "programech klubu elektroniky", nebo bývají
upřesněny na sobotních schůzkách.

=====

ATARI 602, technická zpravodaj pro mikrolelektroniku a výpočetní
techniku. Vydává 602. ZO Svazarmu pro potřeby vlastního aktívů.
Odpovědný redaktor: J. Skála. Adresa redakce: 602. ZO Svazarmu,
Jiřího vnitrova 8, Praha 6, 160 41. Telefon: 341 409. Povolenlo ČVTEI pod
číslem 87005. Cena 9 Kčs dle ČČD č. 1030/202/86
Náklad 500 výtisků.

Praha, květen 1989

Igi/2024