

WORLD
WIDE

188

ATARI KLUB Praha

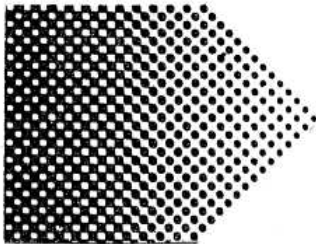


ROČNÍK II.

Vydává 487. ZO Svazarmu —
ATARI KLUB v Praze 4.
Šéfredaktor a vedoucí redakční rady
JUDr. Jan Hlaváček.
Zástupce šéfredaktora ing. Stanislav
Borský.
Obálku navrhl RNDr. J. Tamchyna.
Adresa redakce:
487. ZO Svazarmu - ATARI KLUB Praha
REDAKCE
poštovní příhrádka 51
100 00 Praha 10
Řídí redakční rada: V. Bílek, ing. J. Bis-
kup, RNDr. J. Bok, CSc., ing. S. Borský,
ing. V. Friedrich, ing. O. Hanuš, RNDr.
L. Hejna, CSc., Z. Lazar, prom. fyz.,
CSc., ing. M. Vavřda.
Otisk povolen se souhlasem redakce
při zachování autorských práv a uve-
dením pramene. Rukopisy nevyžáda-
né redakcí se nevracejí. Za původnost
a věcnou správnost ručí autor.
Vychází šestkrát ročně. Neprodejné.
Členům klubu distribuováno zdarma.
Nepravidelné přílohy na objednávku
jsou kompenzovány zvláštním klubo-
vým příspěvkem.
Rozsah čísla 76 stran. Neprošlo jazy-
kovou úpravou.
Tiskne PORS, závod 001, reprografický
provoz.
Do tisku předáno v 1/1988
Vydávání schváleno OV Svazarmu
Praha 4 a OŠK ONV Praha 4.
Evidenční číslo ÚVTEI 86 042.
© ATARI KLUB Praha, 1988

OBSAH

Ze života Svazarmu	
Liberecká patnáctka a její ujednání (Dr. Hlaváček)	2
Software '87	5
Opět v Auroře	5
Zprávy výboru	
Kroužky našeho Atari Klubu se rozrůstají (výbor)	6
Dohody o spolupráci s Atari Kluby (výbor)	7
Ozvali se ataristé ze Sovětského svazu	8
Druhý klub ataristů v Praze (výbor)	8
Co je nového v dílně J. Richtera	9
Redakce připravuje (redakce)	10
Pro začátečníky	
Popis operačního systému DOS 2.5 pro počítače ATARI 130 XE a 800 XL (M. P.)	11
Technické novinky	
Telefax v ČSSR	21
Otázky a odpovědi	
Bloky IOCB a co s nimi (JABS)	22
Tipy a triky	
Pomůcky pro programování (Ing. P. Jandík)	27
Náhrada příkazu CSAVE „název progra- mu“ (Ing. J. Oňa)	33
Klávesnice počítačů ATARI XL/XE (M. Houša)	34
Uživatelské programy	
SpeedCalc — popis programu a návod k jeho používání (Ing. P. Jandík)	36
Basic ZX Spectrum na ATARI XL/XE (S. Pavlíček, J. Richter, M. Oškera)	44
ATARI STATISTICS 1 — návod na použi- tí programu (Ing. E. Blachová)	49
Koutek techniky	
CENTRONICS PRINTER INTERFACE (Ing. V. Fajta)	55
Připojení magnetofonu a systém TURBO 2000 (Ing. Petr Havlíček)	63
Prodloužení životnosti barvicí pásky pro tiskárny ATARI 1029 (I. Martišek)	67
Studený start pro počítače ATARI (Ing. K. Jiříčka)	68
Děrovač disket (Olvis)	68
Listárna	
Jak správně posílat kazety a diskety poš- tou (Dr. Hlaváček)	70
Několik poznámek k šíření systému TURBO 2000 (Dr. J. Tichota)	73
Loader TURBO 2000, COPY 2000 a TURBO BASIC XL 2.0/TB/ (J. Richter)	75



ze života svazarmu

Ve dnech 27.—29. listopadu 1987 se uskutečnilo v Liberci setkání zástupců ATARI KLUBŮ z celé ČSSR. Setkání zorganizovali z vlastní iniciativy soudruzi z pobočky ČSVTS v Liberci. Obeslali pozvánkou jim známé kluby, jiné kluby si informaci o konání akce vyměnily. V Liberci se tak setkali zástupci celkem 15 klubů a jedna skupina ataristů, která dosud nenašla střechu nad hlavou.

LIBERECKÁ PATNÁCTKA A JEJÍ UJEDNÁNÍ

Protože se jednalo o zcela neformální a dobrovolné setkání zástupců různých společenských organizací sdružujících ataristy, jednání probíhalo formou volné diskuse k problémům, se kterými se setkávají všechny kluby. Cílem bylo nalézt co nejoptimálnější řešení sjednocení činnosti ATARI KLUBŮ, pracujících pod hlavičkou Svazarmu, SSM, ČSVTS či jiné společenské organizace.

Přestože se tedy jednalo o setkání neoficiální, došlo při něm podle našeho názoru ke kvalitativnímu pokroku ve vzájemné spolupráci ATARI KLUBŮ, což spatřujeme jednak v tom, že se setkání zúčastnil dostatečně reprezentativní počet klubů a zejména pak v tom, že zde byla přijata dvě podstatná ujednání. Je tedy do jisté míry možné hovořit o „ujednání liberecké patnáctky“.

K jakým ujednáním se tedy v Liberci dospělo?

Za prvé v otázce zlepšení šíření publikačních materiálů jednotlivých klubů a za druhé v přijetí druhého standardu v systému přenosu dat mezi počítačem a magnetofonem.

Vyšlo najevo, že vydavatelské povolení a tiskové možnosti mají z přítomných v současné době pouze AK Praha, Olomouc, Liberec, Hodonín, Tímače a Bratislava. Ve vlastní ediční činnosti mají jednotlivé kluby různé záměry, ale v podstatě chtějí vydávat v rámci svých klubů Zpravodaje a k tomu různé monografie, metodiky, návody, manuály apod. Ostatní kluby (zejm. menší) až na výjimky ediční činnost nevyvíjí a očekávají pomoc odjinud. Jen některé z těchto menších klubů mohou nabídnout aktivní spolupráci (např. překladatelská činnost apod.).

Soudruzi ze Slovenska informovali, že se dohodli na vydávání jednoho Zpravodaje v Bratislavě pro všechny slovenské kluby. Tímto také skončí vydávání tímačského zpravodaje, jak již bylo v jeho posledním čísle potvrzeno. V ČSR bude zřejmě stále vydáváno (alespoň v r. 1988) zpravodajů několik,

Mezi zúčastněnými zástupci klubů došlo k ujednání významnému v tom smyslu, že prostřednictvím v Liberci jmenovaných "tiskových tajemníků" ze všech přítomných klubů (kteří si vzájemně vyměnili adresy) se budou kluby napříště informovat o vlastních publikačních záměrech. Hlavní smysl je spatřován v tom, že bude možné operativně získat přehled o publikačních plánech, ale na druhé straně i o zájmu o připravovaný titul, což povede ke snadnějšímu stanovování nákladu v tiskárnách. Samozřejmě nelze předpokládat, že by některý klub byl schopen zajistit vytištění knihy pro 80 000 ataristů v ČSSR. Všechno bude vázáno mj. technickými možnostmi tiskárny.

Podaří-li se rozvinout tento systém vzájemné informovanosti, mohlo by to přispět k potřebám jednotlivých členů klubů. Navíc se předpokládá omezení zbytečné duplicitní činnosti, zejm. překladatelské. V této souvislosti se ale konstatovalo, že nemůže být na škodu, když k jednomu tématu vyjde v různých klubech i několik titulů, které bude možné vzájemně „vyměnit“.

S ohledem na značný počet ataristů - hráčů dále došlo ke konkrétnímu závazku ze strany pobočky ČSVTS v Liberci. Soudruzi přislíbili, že redakčně zpracují a vydají 1. díl návodů ke hře. Byla přijata jednotící forma těchto manuálů. Tisk bude proveden ve velkém nákladu pro všechny kluby. Jediné (ale to nejpodstatnější), co k tomu liberečtí soudruzi

15

Dr. Jan Hlaváček

potřebují, jsou překlady herních návodů. Ostatní kluby se zavázaly, že zašlou vše, co mají k dispozici.

V této souvislosti se obracíme na všechny členy pražského AK, aby v této iniciativě pomohli i oni. Kopie návodů, které vlastníte, zasílejte přímo na adresu: Jiří Minčíč, Okresní komunální podnik, nám. Bojovníků za mír 2, 469 52 Liberec.

Druhým zásadním ujednáním z Liberce je to, že všemi přítomnými kluby byl přijat systém zrychleného přenosu dat TURBO 2000 autora J. Richtera jako 2. standard.

Úmyslně nepíšeme „... v ČSSR“, protože ačkoliv se v Liberci sešel dostatečně reprezentativní počet klubů, přece jenom tam nebyly všechny, a potom, jak se ukázalo následně, některé kluby nejsou ve výrocích svých představitelů jednotní. K tomu se ještě vrátíme. Jak vyplynulo z diskuse v Liberci, snahou většiny klubů je zůstat navzájem kompatibilní při výměně programů. Jestliže se ze známých důvodů opouští firemní systém, všichni přítomní se shodli na tom, že systém TURBO 2000 je z řady důvodů v této době nejvhodnější. Zdůrazňována byla především dostatečná rychlost přenosu dat, spolehlivost, funkce vyhledávání začátků programů, relativní dostupnost „zturbovaných“ programů aj.

V našem stanovisku ke spontánnímu návrhu ostatních klubů jsme zdůraznili, že systém TURBO 2000 byl jako 2. standard vyhlášen v rámci pražského ATARI KLUBU a ostatním klubům byl pouze doporučen. V žádném případě se nejedná o diktát. Dále jsme připomenuli, že se stále jedná o systém vývojový, který ve své konečné verzi může mít i modifikovanou podobu.

I tak byla diskuse krátká a jednoznačná. Trochu nás mrzela výtka soudruhů z Košic, kteří zkoušejí systém nahrávání okolo 5000 Bd. Projevili zájem konzultovat s J. Richterem, ale ten se jim do konání setkání v Liberci neozval. Stále a znovu opakujeme, že pro tak progresivní záležitost, jakou je zrychlený přenos dat, která se ve své podstatě dotýká většiny členů klubů, je nutné najít společný jazyk zejm. těch, kteří systémy a jejich úpravy vymýšlejí. Především

oni si musí k sobě nalézt cestu. Jestliže J. Richter byl první, který poskytl svůj vynález širokému kolektivu pražských ataristů a pražský ATARI KLUB zajistil rychlou informaci v podobě publikace, nesmějí se dnes hněvat ti, kteří svůj systém (a proč nepředpokládat, že třeba lepší) bránili před rozšířením a hráli si na svém písečku. Jak se ukázalo, v mnoha případech sobeckost, uzavírání se do hranic klubů a „klubečků“ k ničemu není. Zbytečná jsou pak slova o šíření „chýrů“, „konstrukčních zmetků“ a kdoví čeho, jak je dnes částí „ukřivděných“ ataristů systém TURBO 2000 nazýván.

V této souvislosti je potřebné z objektivních důvodů doplnit informaci o libereckém setkání o dvě zajímavé okolnosti, které vyplynuly již několik dní po jeho skončení. Především z dopisu od kolegů z Tlmačů vyplynulo, že přesně v době, kdy se sešli ataristé v Liberci a kde byli i zástupci AK Tlmače, v samotných Tlmačích se konalo podobné oblastní setkání. Nic proti tomu, ale zatímco zástupce tohoto klubu v Liberci hlasuje za systém TURBO 2000, doma je v téže době přijat jako 2. standard „systém 5700“ (přesný název neznáme), pocházející z Nových Zámků. Takže jak je vidět, hovořit o jednotě nelze, a bylo by to asi i zbytečné. O tom, jak je to všechno složité svědčí i výše uvedený příklad. I jeden klub je schopen v jednom dnu přijmout systémy dva.

Zřejmě čas a zejména praxe ukáží, na čem se ujednotíme. Můžeme se ale také dočkat toho, že než se naši páni vynálezci domluví, a my se ujednotíme, tak většina ataristů se zmůže na disketové jednotky, anebo dokonce 8bitové počítače ATARI už budou k vidění jen v muzeích. My už jsme prostě takoví. Příklady by se našlo bezpočet, a nejen v klubech. Než bychom se rychle a operativně dohodli na jedné společné věci dotažené do konce, udělali všechno pro urychlené zajištění výroby nového vynálezu, tak budeme vyvíjet, ověřovat, kalkulovat, diskutovat, zdůvodňovat, kritizovat, pomlouvat. . .

Liberečtí zaslouží uznání za kvalitní organizační i obsahové setkání. Pozitivní bylo, že setkání se nezvrhlo hned na začátku na kopírování programů, jak bývá jindy běžným zvykem. Pokud jde o to nešťastné kopírování, které zpravidla rozbije pořadatelům plány, i na to byli perfektně připraveni. Přinesli takovou „samici“, na kterou by se snad mohli napojit všichni účastníci setkání, kdyby s sebou měli magnetofony. Zdá se, že právě v otázce výroby hardwarových zařízení nás ještě Liberečtí mile překvapí. Zdá se, že již v mnohém mají jasno.

Z diskuse na jednání v Liberci vyplynul i staro-nový problém — vytvoření centralizujícího prvku v činnosti ATARI KLUBŮ. Návrh na vytvoření „nadklubového řídicího orgánu“ byl v bouřlivé diskusi zamítnut. Argumentovalo se objektivními důvody (začlenění ATARI KLUBŮ do různých společenských organizací s různými stanovami a organizačními strukturami), ale také poučením z praxe. Jde o to, že žádná z existujících organizací nemůže být pro ostatní řídicí. V době setkání nikoho nenapadlo navrhnout, aby se jednotlivé organizace v meziobdobí mezi společnými setkáními (tyto by se měly konat dvakrát ročně) střídaly ve funkci koordinátora, či administrativního centra, tedy onoho spojujícího prvku, po kterém všechny kluby volají. Podstata by spočívala v tom, že klub, který byl pověřen zorganizováním setkání, by do příštího setkání zprostředkoval spojení všech se všemi, ale také připojení dalších nových klubů. Již v úvodu jsme konstatovali, že libereckého setkání se neúčastnili kolegové z Ostravy, oba brněnské kluby, svazáci z Mělníka, kluby z Trenčína a Žiliny a jiné. Přesto se lze domnívat, že i tyto a další kluby by měly figurovat v adresáři ATARI KLUBŮ, vydaného v Liberci. Bylo by záslužné, aby Liberečtí soudruzi nepřítomné kluby informovali, a aby se tyto kluby zúčastnily jarního setkání, které budeme z pověření ČÚV Svazarmu organizovat v Praze.

Jestliže se váženému čtenáři zdá, že jsme tímto příspěvkem zbytečně potiskli papír, nesdílime tento názor. ATARI KLUB — to nejsou jenom počítače, programy, honba za jejich bezplatným získáním, bitva o literaturu (také co možná nejlevněji). Je to i náročná a vyčerpávající (a nutno říci i bezplatná) práce mnoha funkcionářů, kteří mají zájem, aby klub fungoval.

SOFTWARE '87

ÚV Svazarmu uskutečnil ve dnech 4. a 5. prosince 1987 v ZK ROH pracovníků obchodu v Pařížské ulici v Praze 1. ročník přehlídky počítačových programů Svazarmu. Organizačním zajištěním (na dobré úrovni) byla pověřena 602. ZO Svazarmu.

Náš ATARI KLUB předváděl několik programů. Samozřejmě především programy systému TURBO 2000, dále pak program, u kterého se velice rychle vžilo označení „vykřadač sinclairů“ (TRAPSAT) dále databankový program MINSY a několik dalších.

Zájem veřejnosti byl značný. Došlo zde i k navázání kontaktů s jinými značkovými kluby ATARI z celé ČSSR, jejichž zástupci do Prahy přijeli.

Škoda, že akci chyběl závěr, tj. především nějaké vyhodnocení na přehlídce předváděných programů, nebo alespoň vydání jejich seznamu (s uvedením autora, organizace, kde jej lze získat atd.). Alespoň my jsme se tyto informace nedozvěděli. Možná že budou k nalezení ve Sborníku, jehož vydání pořadatel slíbil.

HI

OPĚT V AUROŘE

Tentokrát na akci nazvané KDO SI HRAJE, NEZLOBÍ, aneb vánoce s počítači v Auroře. V maratonu začínajícím 18. prosince (s přestávkou svátečních vánočních dní) a končícím předposlední den roku 1987 náš klub zajišťoval stánek ATARI.

Tisíce dětí prověřily nejen kvalitu počítačů (a hlavně joisticků), ale také naše již ostřílené lektory — Frantu Tvrďka, René Hánu, Mirka Ondříška a Vlastu Bílka, kterým tentokrát pomáhali i Jan Pilný a Petr Vacek. Jak se vyjádřil šéf stánku Franta Tvrdek, mnohdy se i zapotil při hledání odpovědi na položenou otázku. Nejvíce jej prý překvapilo zjištění, že se značně rozšířil počet těch dětí, které již vidí pod pokličku, tedy velmi dobře se orientují v konstrukci počítače, v assembleru a jiných programovacích jazycích a v řadě dalších vysoce odborných záležitostech. A to je především pro jejich budoucnost dobré.

Za zmínku stojí i to, že jsme zde prakticky poprvé spolupracovali s Klubem výpočetní techniky ATARI nově vytvořeným u 602. ZO Svazarmu koncem roku loňského roku. O vzniku tohoto klubu informujeme na jiném místě.

HI

KONTAKT - DALŠÍ OBLASTI ZÁJMU

Hledáme další zájemce v oblastech:

- šachy (hry, řízení turnajů, ELO, žebříčky),
- využití tiskáren ATARI 1027, 1029
- vyučovací programy.

red.



ODKOUPÍME

ATARI KLUB Praha odkoupí od soukromých majitelů mikropočítače ATARI XL/XE, magnetofony XC 12, disketové jednotky 1050, tiskárny 1029 i 1027 a další hardwarová zařízení. Nákup může být realizován kupní smlouvou za odhadní nebo dohodnutou cenu. Své nabídky zasílejte na adresu klubu nebo je sdělte kterémukoliv členu klubu.

KROUŽKY NAŠEHO ATARI KLUBU SE ROZRŮSTAJÍ

Celý loňský rok jsme usilovali o zapojení jednotlivých členů do pracovních kroužků. Důvody jsme obsáhle vysvětlili. Do konce mi-

nulého roku se podle údajů pracovníků spravujících členskou evidenci podařilo vytvořit jak v Praze tak i mimo Prahu celkem 43 pracovních kroužků. Jejich seznam s adresami vedoucích kroužků uvádíme dále.

K tomuto seznamu je potřebné doplnit, že do doby, kdy čtete tyto řádky může dojít ke změnám. Vedoucím kroužků jsme poskytli seznamy našich členů, kteří bydlí v sídle kroužku a jeho okolí, s cílem napomoci v jejich soustředění. Někde se to povedlo, jinde se nedomluvili. U mimopražských spočívá složitost mj. v tom, že v některých případech se ze skupiny našich členů vytvořila organizace nová (kolegové nyní u nás hostují), nebo v rámci vytvořené ZO je několik našich hostů, několik hostů jiné ZO atd. V tomto stavu vytvořit pracovní kroužek je obtížné. Je to však pořád lepší, než když k sobě lidé nenašli cestu. Nejvíce problémů z toho pak má distribuční skupina, která musí vybavovat stovky a tisíce zásilek. V souvislosti s šířením publikací jiných klubů, jak o tom informujeme na jiném místě, se tento problém ještě zvětší.

výbor

ZPRÁVY VÝBORU



Sídlo kroužku

Vedoucí, adresa klubu nebo vedoucího kroužku

Aš	Zd. Šeberle, Dlouhá 32, 352 01 Aš
Benešov	Petr Říha, Jiřího Franka 1644, 256 01 Benešov
Beroun	ing. V. Blažek, Tyršova 51, 266 01 Beroun
Brno	Št. Hrtoň, 505. ZO Svazarmu, Žitná 17, 605 00 Brno telefon (Z.) 27 806, (D.) 62 8403
Březnice	Fr. Bártík, Pionýrů 125, 262 72 Březnice
České Budějovice	ing. J. Novák, Krčínova 50, 370 11 České Budějovice
Chomutov	J. Kohlík, Palackého 4274, 430 00 Chomutov
Dvůr Králové	D. Hoffman, Krátká 1876, 544 01 Dvůr Králové
Dol. Rožínka (Uran. doly)	ing. M. Dvořák, Tyršova 485, 592 31 Nové Město na Moravě
Havlíčkův Brod	A. Janovský, Herálec - Pavlov 19, 582 55
Hradec Králové	ing. V. Fajta, ORGATEX, Veverkova 1631, 500 05 Hradec Králové
Kladno	V. Havelka, Pražská 2974, 272 00 Kladno
Litoměřice	Dr. Ct. Sedlák, Stavbařů 2002, 412 01 Litoměřice
Mladá Boleslav	MUDr. V. Kudlička, Leninovo nám. 1060, 293 01 Mladá Boleslav
Nymburk	Jan Pilný, 288 02 Všechlapy 150, okr. Nymburk
Nové Město na Moravě	ing. R. Borek, Pavlovova 1510, 592 31 Nové Město na Moravě
Pelhřimov	Alfr. Bruch, Cholossiova 754, 393 01 Pelhřimov

Pízeň	ing. J. Čechura, E. Krásnohorské 23, 323 11 Pízeň
Rovečné	ing. Z. Knotek, 592 65 Rovečné 139, okr. Žďár nad Sázavou
Slaný	ing. Jan Rada, Arbesova 1425, 274 01 Slaný
Strakonice	J. Zeman, Tržní 1154, 386 01 Strakonice
Sokolov	Jar. Buček, U Hřiště 85, 357 07 Oloví
Tábor	Kol. Tekale, Berlínská 2756, 390 00 Tábor
Turnov	V. Vohnický, Žižkova 1340, 511 01 Turnov
Znojmo	Jin. Krobot, 671 23 Chvalovice 92
Slovensko	
Banská Bystrica	RNDr. I. Serváček, Volkova 1, 974 00 Banská Bystrica
Bratislava	K. Vavrovič, Togliatiho 46, 821 07 Bratislava
Šafárikovo	ing. A. Kudlák, Školská 11, 982 01 Šafárikovo
Giraltovce	Duš. Macura, Čapajevova 508, 094 31 Hanušovice nad Toplou
Praha	
Žižkov, Praha 3	Jar. Štěpán, Nad lukami 11, 130 00 Praha 3
VŠCHT, Praha 4	M. Ondříšek, Chemická 952, 148 29 Praha 4
Chirana Modřany, Praha 4	ing. Pavel Rada, Mezihorská 56, 140 00 Praha 4 - Modřany
Ohradní, Praha 4	Richard Beneš, Ohradní 1334, 140 00 Praha 4
Podolí kolejí, Praha 4 Praha 5	V. Pechanec, Na lysině 12, 140 00 Praha 4
VÚ Železnic, Praha 5	Milan Jebavý, U Mrázovky 16, 150 00 Praha 5
Královka, Praha 6	ing. M. Smrčka, Poláčkova 1629, 251 01 Říčany
Strahov, Praha 5	ing. J. Jícha, Hařusova 1330, 149 00 Praha 4
Řepy, Praha 6	S. Pavlíček, Strahov bl. II pokoj 208, 160 00 Praha 6
Bohnice, Praha 8	Zd. Černý, Krolmusova 372/69, 163 00 Praha 6
Tesla, Praha 8	Old. Víšek, Křivenecká 443, 181 00 Praha 8
Praha 10	M. Lacina, Zelenohorská 513, 180 00 Praha 8
Gymnasium	ing. M. Tichý, Pízeňská 41, 150 00 Praha 5
Přípotoční	Roman Řípa, Gercerova 902, 102 00 Praha 10

DOHODY O SPOLUPRÁCI S ATARI KLUBY

Od počátku činnosti našeho klubu jsme hledali pracovní kontakty s jinými ATARI KLUBY v ČSSR. Již ve 3. čísle (1987) Zpravodaje AK jsme informovali, že připravujeme dohodu o spolupráci s ATARI KLUBEM působícím v Tlmačích. Od té doby jsme uzavřeli tuto oficiální formu spolupráce s dalšími kluby, jejichž adresář dále uvádíme.

Kromě toho jsme byli požádáni nebo jsme nabídli uzavření dohody o spolupráci s ataristy v Trenčíně, Mostu, Gottwaldově, Brně (141. ZO), Hodoníně a v Praze 6 - Dejvicích.

Dohody o spolupráci uzavřené

- ZO Svazarmu — klub elektroniky při SES,
vedoucí ing. Jaroslav Burjaniv,
Čsl. armády 24, 935 21 Tlmače
- ZO Svazarmu — ATARI KLUB, pošt. schr. 137,
772 11 Olomouc
505. ZO Svazarmu, Žitná 17, 605 00 Brno
(předseda ZO ing. Ivan Sommer, vedoucí
sekce ing. Štěpán Hrt'ň)
303. ZO Svazarmu Bratislava, vedoucí
ing. Pavol Doršič, Rajčianská 6,
821 07 Bratislava
- Pobočka ČSVTS Liberec, vedoucí ing. Jiří
Minčíč, Okresní komunální podnik,
nám. Bojovníků za mír 2, 460 54 Liberec

Členům klubu dlužíme informaci o obsahu dohod o spolupráci, jak o tom píšeme ve 3. čísle (1987) Zpravodaje AK na str. 4. Opisovat celý text dohody nepovažujeme za účelné. Proto jen stručně: V „Obecných ustanovení“ se uvádí důvod a smysl uzavírané dohody. „Předmětem dohody“ je ustanovení o vzájemném poskytování všech publikačních materiálů s upřesněním počtů a principů výměny a úhrady, dále o vzájemné výměně softwaru, poskytování dalších technických informací, zapůjčování literatury, návodů pro stavbu hardwarového vybavení apod. a posléze ustanovení o vzájemných osobních kontaktech zástupců obou klubů. „Závěrečná ustanovení“ pak právně specifikují platnost a účinnost dohody. Uzavřené dohody jsou stvrzeny podpisy předsedů výborů ZO a vzaty na vědomí nadřízenými orgány Svazarmu.

Dohody o spolupráci nejsou chápány jako neměnné dogma, nýbrž jako prostředek nezbytného potvrzení právního vztahu dvou organizací.

výbor

OZVALI SE ATARISTÉ ZE SOVĚTSKÉHO SVAZU

Prostřednictvím Dr. Jiřího Boka, CSc. jsme získali kontakt na uživatele výpočetní techniky ATARI v SSSR. Jedná se o skupinu, vytvářející klub v rámci VLKSM v Podolském okrese (podmoskevská oblast), který má plnit úlohu centralizující organizace. Předpokládá sdružení ataristů, vydávání publikačních materiálů, výchovu mládeže k aplikacím výpočetní techniky. Soudruzi potvrdili, že mají velký zájem uzavřít s námi dohodu o spolupráci, jejímž předmětem by byla zejm. výměna publikačních materiálů a programů. Nabízejí i možnost pozvání našich zástupců do SSSR k pracovnímu setkání. Na realizaci dohody se pracuje.

Shodou okolností jsme se na libereckém setkání ataristů seznámili se soudruhem Viktorem Bajdunem, kandidátem technických věd, který několik měsíců působil na pracovním stáži v Ústavu kybernetiky SAV v Bratislavě.

Kromě toho, že poskytl pro náš Zpravodaj AK příspěvek k užívání jazyku LISP 65 (t.č. u recenze), sdělil nám, že má kontakty na klub uživatelů výpočetní techniky ATARI, kterou v Moskvě zorganizoval šachový mistr světa G. Kasparov. Prostřednictvím s. Bajduna očekáváme zprávu o moskevských ataristech.

HI

DRUHÝ KLUB ATARISTŮ V PRAZE

Pro správnou orientaci ataristů žijících v Praze považujeme za potřebné informovat, že u 602. ZO Svazarmu v Praze 6 - Dejvicích byl vytvořen nový Klub uživatelů výpočetní techniky ATARI. O přípravě tohoto počínu 602. ZO jsme získali první informace na podzim loňského roku. Teprve na setkání ataristů v Liberci jsme měli možnost poprvé jednat s vedoucím tohoto klubu ing. Karlem Příhodou. Vzájemně jsme si vysvětlili genezi vzniku obou klubů, jakož i naše stanovisko k výpadům a nevhodným pomluvám některých funkcionářů 602. ZO vůči našemu klubu a vůči počítačům ATARI (viz například články D. Dočekala zveřejněné ve Vědě a technice mládeže, Světu práce a jiných časopisech). V souladu s tím, co nám bylo řečeno na ČÚV Svazarmu, jsme od ing. Příhody dostali ujištění, že ataristé z Dejvic mají zájem na normální přátelské spolupráci mezi oběma kluby.

Aby nedocházelo ke zmatkům, dohodli jsme se předně na tom, že náš klub bude používat název „ATARI KLUB Praha“, dejvičtí pak „Klub uživatelů výpočetní techniky ATARI u 602. ZO Svazarmu“.

Dohodli jsme se dále na tom, že členům obou klubů bude bez problémů umožňováno (pokud projeví zájem) hostování nebo přehlášení v druhém klubu, tak jak to umožňují Stanovy Svazarmu.

Kolegové ataristé z 602. ZO nás následně požádali o uzavření oficiální dohody o spolupráci. Protože prozatím nemají úplně konsoolidovanou členskou základnu, postrádají programové vybavení a zejména pak literaturu, budeme je vybavovat literaturou vydávanou

naším klubem. Jak se dál spolupráce rozvine a jaký bude další vývoj, to ukáže čas.

Kolegům Klubu uživatelů výpočetní techniky u 602. ZO Svazarmu přejeme, aby tentokrát byly splněny sliby dané jim vedoucími činovníky 602. ZO (my už v tomto smyslu svou zkušenost máme), aby co nejrychleji překonali počáteční etapu konstituování klubu, která nikdy nebývá lehká a aby se jim v práci dařilo.

výbor

CO JE NOVÉHO V DÍLNĚ J. RICHTERA

Množí se dotazy, ale šíří se také různé fámy. K čemu? K systémům zrychleného přenosu dat TURBO 2000 autora Jiřího Richtera.

Jak píšeme na jiném místě, na jedné straně se vede upřímná snaha o sjednocení ataristů, na druhé straně to ale vypadá, že každý klub chce jít vlastní cestou - tedy alespoň pokud se jedná o vyřešení nejslabšího prvku počítačů ATARI, kterým je zřejmě systém přenosu dat. Tady se odvíjí pěkná vřava. Počítačový národ ataristů se rozděluje na několik skupin: nadšené zastánce TURBO 2000, aktivní spolupracovníky, odborné kritiky, ale také „vynálezce vynalezeného“, lépe řečeno vykradače, dokonce i uražené ješity atd. Jako vždy, všude a ve všem. Vedle toho pak stejně pilně jako J. Richter pracují další šikovní vynálezci, kteří přišli nebo přicházejí se systémy novými. K našim uším pak doléhají zprávy, že se jedná o systémy lepší než TURBO 2000. V některých případech tomu tak určitě je. Ale právě tady asi nachází živnou půdu ještě jedna zvláštní skupina, a tu bychom mohli označit slovem „fámisté“ (kdyby to bylo slovo uznávané pravidly českého pravopisu). Zajímavé je, že nejčastějším argumentem pro „lepší než TURBO 2000“ je prezentována rychlost přenosu dat, tedy rychlost ještě větší, než Richtrových 2074 Bd. Že to ale v praxi úplně normálních a běžných uživatelů (a těch je drtivá většina) bude mj. znamenat náročné shánění super-kvalitních kazet, to se nedodává.

Snaha o nalezení ještě lepšího je ale na-

prsto zákonitá a doplníme, že svým způsobem i příznivá.

Systém TURBO 2000, který zaznamenal nejrychlejší a největší rozšíření (a nejen v pražském ATARI KLUBU) vedle do té doby připravovaných nebo již experimentálně fungujících jiných systémů, vyvolal aktivitu mnoha činnorodých lidí. Tak se daří společnými silami vyplatit to, s čím přišel J. Richter. Za této situace je pochopitelné, že u těch, kteří v téže době na jiném místě republiky již věděli jak na to, nebo dokonce uměli nahrávat rychle, mohl systém TURBO 2000 publikovaný v příloze II Zpravodaje AK Praha napomoci v dokončení vlastní práce, ale také mohl vyvolat překvapení, kritiku, opovržení ...

Teď by mělo jít o to dokončit jednotlivé vývojové cesty, rychle rozhodnout, který ze systémů je neoptimálnější (nejdokonalejší nemůže být žádný) a dohodnout se. Zdá se, že až na to poslední (dohoda), se věc podaří. Příznivou aktivitu v tomto směru vyvinula redakce časopisu ELEKTRONIKA, která požádala jednoho ze skutečných znalců počítačů ATARI XL/XE/ a dodejme nezatiženého nutností k přechodu na některý ze stávajících systémů zrychleného přenosu dat z magnetofonu/, aby soustředil poznatky o jednotlivých systémech a provedl vyhodnocení. Nebude to mít lehké.



Pro členy pražského ATARI KLUBU a hosty naší organizace přinášíme nejaktuálnější (jak nám to dovoluje výrobní lhůta časopisu) informaci přímo od J. Richtera. Je z 13. 1. 1988:

„Systém TURBO 2000 se neustále vyvíjí a musím dodat, že mi značnou měrou pomáhají kamarádi. V otázce programového vybavení se postupně rozšiřuje řada systémů využívajících „dvoutisícovky“ k nahrávání dat. Dokončuje se nový typ zařízení, které umožní nahrávání dat novou rychlostí a bude využitelný např. v BASICu, v textových editorech apod. Nová forma nahrávání řeší mnoho problémů, kterými argumentují moji odpůrci, protože původní systém TURBO 2000 sloužil pouze pro nahrávání hotových

programů ve strojovém kódu (především her) a nebyl tedy určen pro BASIC.

Vyvíjí se samozřejmě i úprava magnetofonu. Jsem si dnes vědom, že dříve navržená a publikovaná úprava magnetofonu byla asi nejslabším článkem systému TURBO 2000. Na svoji obhajobu musím dodat, že nebyť obrovského tlaku ze všech stran loni v létě, když všichni chtěli TURBO a já měl zkoušky a pár dní před vojnou, mohl jsem připravit úpravu dokonalejší. V té době jsem mohl dát jen to, co bylo v poznámkách na papíře a nikoliv to, co jsem měl již v hlavě. Proto jsme se tomuto nedostatku intenzivně věnovali. Ukázalo se, že hardwarovou úpravu magnetofonu lze stejným výsledkem řešit několika způsoby zapojení, některé z nich pak přinášely další výhody a zlepšení. Dobré bylo, že jsme mohli čerpat i z několika skutečně výborných námětů a doporučení od kolegů z celé republiky, za což jim touto cestou upřímně děkuji. V této souvislosti se omlouvám všem, kteří čekali na odpověď k nějakému dílčímu problému. Na korespondenci mi již nezbývá čas.

Dnes tedy existuje několik nových zapojení s plošným spojem i bez něj, které umožňují spolehlivě nahrávat podstatně vyšší rychlostí, než umožňuje dosud rozšířená úprava. Domnívám se, že brzy dáme k dispozici nejdokonalejší variantu.

Pro tyto nové úpravy magnetofonu je již připraven i software, který umožňuje nahrávat programy libovolnou rychlostí v rozsahu 2000 až 5000 Bd (zaokrouhleno). Tím snad uspokojíme i ty, kterým je 'dvoutisícovka' málo. Všechno, o čem zde hovořím, připravujeme ke zveřejnění, ale nikdo by se neměl zlobit, že nespěcháme. Všechno chce svůj čas. A poučení z minula je toho dokladem."

HI

REDAKCE PŘIPRAVUJE...

Každý pořádný časopis informuje své čtenáře o tom, co najde v příštím čísle. S tímto cílem jsme také my zavedli tuto minirubriku. Protože ale zřejmě nejsme „pořádný“ časopis, občas se nám stalo, že jsme nemohli dostát slibu. Náš časopis se ale tvoří (lépe řeče-

no slepuje) v podstatně jiných podmínkách, než to dělají profesionálové. Těmito řádky bychom tedy chtěli požádat o shovívavost čtenářů, zejm. pokud jde o termíny vydání příslibených publikačních materiálů.

I v tomto roce budeme průběžně informovat, co připravujeme na stránky běžných čísel Zpravodaje AK a jaké další rozsáhlejší materiály (přílohy Zpravodaje AK) budou vydány.

Podle edičního plánu bychom chtěli v 1. pololetí 1988 vydat:

- Mikroprocesor 65C02 a R6520 (PIA), 20 str. — popis připravený kolegy z Brna.
- Fajta, V.: Základy programování v assembleru — učebnice cca 250 stran
- Chasim, M.: Programování v assembleru počítačů ATARI — překlad (ing. K. Šmuk), 120 stran
- Lawrow, S. D.: MAC/65 — překlad referenční příručky (ing. P. Jandík), 70 stran. (Z předmluvy překladatele: „Při své programátorské praxi jsem se musel prokousat přes několik assemblerů, než jsem konečně došel k poslední verzi MACu, který mi vyhovuje téměř sto procentně. Jeho rychlost je hlavní devizou.“)
- Přerušeni při vertikálním zatemnění — jeho využití pro jemné rolování a generaci hudby — upravené překlady z Analog Computing (ing. J. Hůrek), cca 50 str.
- Sedláček, V., Friedrich, V.: LOGO — učebnice, cca 80 stran
- KYAN PASCAL — překlad doplněný poznatky překladatele (M. Ondříček)

Tak jak píšeme na jiném místě, rozvíjí se výměna publikačních materiálů s jinými kluby. V současné době máme k dispozici publikaci vydanou olomouckým klubem „Průvodce assemblerem“ a publikaci vydanou hodonínským klubem „Textové módy“. V Olomouci jsme dále objednali několik stovek publikace „Atari kouzlí“. Nabídku na odběr několika návodů uživatelských programů (z nichž některé máme připravené k vydání i v našem Zpravodaji AK) máme z Timačů.

Prostřednictvím našeho bulletinu QUICK REPORT budeme informovat o způsobu od-

běru a objednávání jak našich, tak dalších publikací. Musíme vyřešit nejjednodušší, ale dokonalý způsob objednávání a distribuce. Získali jsme určitou zkušenost s distribucí našich prvních dvou příloh, ale bohužel i zkušenost negativní. Určitá část členů klubu nepovažovala za vhodné za zaslané přílohy zaplatit. Tomu se musíme přizpůsobit.

V této souvislosti dále sdělujeme, že olomoucký klub nám nabízí neomezený odběr jejich klubového zpravodaje. Problém spočívá v tom, že neznáme zájem našich členů. Zajistit odběr pro všechny členy a hradit jej z klubových příspěvků nepřipadá v úvahu (to bychom finančně nezvládli). Navíc je tu

ona již několikrát vzpomínaná otázka distribuce. Současných pět pracovníků distribuční skupiny by nebylo schopno rozesílat olomoucký zpravodaj individuálně všem. Potřebujeme proto udělat průzkum zájmu, přičemž v této chvíli ještě neslibujeme, že budeme zajišťovat odběr tohoto časopisu. Mj. jde i o to, že skutečně zajímavé příspěvky publikované v olomouckém zpravodaji budeme podle dohody přetiskovat. Obracíme se proto na členy, kteří by měli zájem o odebrání olomouckého zpravodaje, aby napsali na korespondenčním lístku na adresu naší redakce. Podle výsledku průzkumu přijmeme rozhodnutí.

red

Pro začátečníky

POPIS OPERAČNÍHO SYSTÉMU DOS 2.5 PRO POČÍTAČE ATARI 130 XE A 800 XL

M.P.

Úvod

Tento článek vznikl překladem, opravením věcných chyb a doplněním příručky v německém jazyce, kterou dodává TUZEX k disketové jednotce 1050. Příklady komunikace uživatele s počítačem uvádíme dvojjazyčně, anglicky a německy. Přednost jsme dali původní americké verzi, která je mezi našimi uživateli rozšířenější.

ATARI DOS 2.5

S disketovou stanicí 1050 byl v roce 1983 představen DOS 3. Využíval sice disketu na kapacitu 127 kbyťů, v praxi se však neosvědčil. Později byl proto zaveden DOS 2.5 navazující na osvědčený DOS 2.0 a zachovávající vysokou kapacitu paměti.

Jména souborů

DOS 2.5 ukládá data a programy do souborů na disketách. Každý soubor má jméno, které se objeví v adresáři diskety. Jméno souboru sestává ze dvou částí, z první části, která může

mít až 8 znaků a z nepovinného přídávku o maximálně třech znacích. Přípustnými znaky jsou velká písmena a číslice. Čárky, pomlčky a podobná zvláštní znaménka nejsou povolena. Jméno nesmí začínat číslicí. Nepovinný přídavek slouží k označení druhu souboru a je od jména oddělen tečkou. Přípustná jména souborů jsou kupř.:

PROGRAM.BAS SOUBOR3.DAT ZPRAVA.5 K8506.PRT TEST1

Neplatná jména souborů hlásí DOS 2.5 (dále jen DOS) jako chybu 165. Nepřípustná jména jsou kupř.:

85.06.07.DEN 1.PROGRAM Skolrok.REC

Číslo disketových stanic

Počítač ATARI může najednou pracovat až se čtyřmi disketovými stanicemi. Hlavní stanice, dodávající do počítače DOS, má číslo 1. Další stanice jsou kabelem připojeny vždy na stanici předešlou a jsou vzestupně očíslovány. Číslo stanice se nastavuje dvěma posuvnými přepínači v otvoru na zadní straně stanice. V návodu ke stanici najdete kombinace přepínačů pro vytvoření potřebných čísel.

V mnoha případech (kupř. při kopírování souborů) musí dostat DOS číslo disketové stanice, které se stane částí jména souboru. Má-li se kupř. kopírovat soubor "DOPIS.TXT" ze stanice 1 na stanici 2, zní adresová část správného zadání:

D1:DOPIS.TXT,D2:DOPIS.TXT

Není-li zadáno číslo disketové stanice, dosazuje DOS sám číslo 1. Příklad:

DOPIS.TXT,D2.DOPIS.TXT

Divoké karty (Jokery)

Pro pohodlí uživatele má DOS připraveny dva druhy tzv. "divokých karet". Jsou to vlastně náhradní znaky. Hvězdička "*" nahrazuje libovolný řetězec znaků, otazník "?" znak jen jeden. Jestliže jste zapomněli přesné jméno souboru, zadáte z něho jen znaky, které si pamatujete a zbylé neznámé nahradíte hvězdičkou. Hledáte-li kupř. soubor, který začíná znaky "TA", dáte jen "TA*.*". DOS nalezne všechny soubory na disketě, které začínají těmito znaky. Hledáte-li soubory s přídávkem ".BAS", bude zadání "*.BAS". Následující jména souborů splňují zadání "K?N*.*":

KONFER KTN144.DAT KONCERN.001

Menu — Jídelníček

Z vestavěného BASICu (popř. z BASICu v modulu na starší sérii 400/800) zavoláte DOS příkazem "DOS". Poté se přetáhne menu-program (DUP.SYS) z diskety ve stanici 1 do počítače a na stínítku se objeví menu DOSu. Je-li v tomto okamžiku v paměti počítače BASICovský program, bude porušen. Zabráníte tomu tím, že použijete program "MEM.SAV" (viz příkaz "N"). Menu umožňuje řídit DOS. Zadáte písmeno požadovaného příkazu a stisknete <RETURN>. DOS ukáže v příkazové řádce na dolním okraji stínítka, která dodatečná zadání potřebuje. Všechna zadání zakončujete klávesou <RETURN>. Chybová hlášení vyřadíte klávesou <BREAK>.

Vysvětlení jednotlivých příkazů v menu:

A - DISK DIREKTORY (Adresář diskety)

DISK INHALT

Adresář obsahuje seznam všech souborů na disketě. Po zvolení tohoto příkazu zadáte jméno hledaného souboru, číslo disketové stanice (D2:PROGRAM.BAS) a periferní zařízení, na kterém se výsledek příkazu má objevit. Ve jménu souboru můžete využít "divokých karet" (D2:*.BAS). Nespecifikujete-li blíže jméno souboru, budou vypsaný všechny soubory na disketě ve stanici 1, jako kupř. odpovídá zadání D1:*.*. Periferním zařízením je normálně vždy obrazovka (E:). Pro vypsání na tiskárnu zadávejte "P:". Zadání čísla stanice a jména souboru musí být od zadání periferie odděleno čárkou, kupř. D1:*.*,P: apod. Dvojtečku za písmenem označujícím periferii nesmíte vynechat, jinak bude na disketu uložen soubor s písmenem periferie jako jménem!

Soubory označené v adresáři ostrými závorkami "<...>" obsazují na disketě sektory od 720 výše. Neoznačené soubory obsazují sektory do 707 a mohou být zpracovány i pod operačním systémem DOS 2.0S. DOS 2.5 zpracovává oproti tomu všechny v adresáři uváděné soubory, tj. až do sektoru 1010. Diskety naformátované příkazem "P" mohou být zpracovány pod operačním systémem DOS 2.0S v rozsahu do 707 sektorů.

B - RUN CARTRIDGE (Zpět na ROM-modul)

ZUM STECKMODUL

Příkaz zprostředkuje návrat na vestavěný BASIC (v počítačích XL a XE) nebo na zasunutý ROM-modul (Editor/Assembler, LOGO atd.). Není-li zasunut žádný modul, vrací se DOS zpět k BASICu.

C - COPY FILE (Kopírovat soubor)

DATEI KOPIEREN

Příkaz dovoluje přenášet data z diskety na disketu. Zadáváte k tomu zdrojové a cílové zařízení. Textové soubory mohou být přímo z diskety přenášeny na tiskárnu (D1:TEXT.DAT, P:). Obsahuje-li disketa s DOSem také program MEM.SAV, ptá se DOS, smí-li použít programovou část paměti. Nabídne-li paměť (odpověď "Y", resp. "J", tj. "ANO"), smaže se program MEM.SAV a BASICovský program v RAM paměti počítače je ztracen. Kopírování probíhá poměrně rychle. Odpovíte-li "N", tj. "NE", je i tehdy kopírovací procedura možná, ale probíhá bez využití programové části paměti, a to podstatně déle. Máte-li více než jednu disketovou stanici, je pořízení archivních kopií důležitých souborů tímto příkazem velmi jednoduché:

D1:PROGRAM.BAS,D2:PROGRAM.BAS

Toto zadání kopíruje soubor "PROGRAM.BAS" z diskety ve stanici 1 na disketu ve stanici 2. Zadání

D1:*.*,D2

kopíruje všechny soubory ze stanice 1 do stanice 2 kromě těch, které nesou přídavek ".SYS", tj. které jsou součástí DOSu.

Ke spojení dvou souborů se využije doplňku "/A". Zdrojový soubor se poté napojí na cílový soubor:

D1:SOUBOR.1,D1:SOUBOR.2/A

"Divokých karet" lze použít jen ve zdrojovém souboru. Příkaz "C" nepracuje s kazetovým magnetofonem.

D — DELETE FILE(S) (Mazat soubor-y)

LOESCHE DATEI

Příkaz slouží k mazání nepotřebných souborů. Jednoduše zadáte jméno souboru, DOS 2.5 prohledá adresář diskety a pro jistotu se vás zeptá, zda tento soubor opravdu chcete smazat. Otázku můžete vyřadit doplňkem "/N" za jménem souboru. Používání "divokých karet" je v tomto příkazu nebezpečné. Když si ke všemu ještě vyřadíte dotaz "pro jistotu", nemáte pak žádnou kontrolu o mazaných souborech. Zabezpečené soubory (viz příkaz "F") musí být před mazáním odblokovány.

E — RENAME FILE (Změnit jméno souboru)

NAMEN AENDERN

Při přejmenování souborů je třeba nejdříve zjistit, zda na disketě neexistuje jiný soubor se stejným jménem jako je nové jméno souboru. DOS se ptá po starém, na disketě vyznačeném jménu souboru a po jeho jménu novém. Zadejte obě jména oddělená čárkami. Jestliže je soubor zabezpečen podle příkazu "F", hlásí DOS chybu 167. Použití "divokých karet" je dovoleno, může ale vést ke ztotožnění jmen. Je proto třeba s nimi pracovat opatrně. Nastane-li takový případ, zadejte k jeho nápravě nové jméno souboru bez "divokých karet". DOS poté změní jméno jen v pořadí prvního nalezeného souboru.

F — LOCK FILE (Zabezpečit soubor)

DATEI SICHERN

Důležité soubory (vlastně neexistují žádné nedůležité) můžete příkazem "F" zabezpečit proti náhodnému přepisu nebo vymazání. Pokus o takovou operaci, provedenou třeba i z běžícího programu, je zakončen chybovým hlášením 167, protože v adresáři souborů na disketě je u zabezpečených souborů ochranné označení. Tyto soubory poznáte podle hvězdičky před jménem. Ochrana však nepůsobí při formátování a kopírování celých disket! Nejspolehlivější ochrany proti přepisu celé diskety dosáhnete přelepením zapisovacího výřezu na levé straně diskety neprůsvitným materiálem.

G — UNLOCK FILE (Odblokovat soubor)

DATEI FREIGEBEN

Příkaz odstraní zabezpečení souboru (viz příkaz "F"). Soubor lze pak přejmenovat, přetahovat nebo mazat.

H — WRITE DOS FILE (Zapisovat DOS soubory)

DOS SCHREIBEN

Chcete-li pracovat s dosud prázdnou a naformátovanou disketou — viz příkaz "I" — musíte na ni napsat oba DOS soubory. Zvolíte-li tento příkaz, ptá se DOS po čísle disketové stanice, na které má psát. Zvolená stanice se na chvíli rozeběhne, DOS při tom zkouší, zda byla disketa správně naformátována. Poté se pro jistotu táže, zda má příkaz provést. Odpovíte-li "ANO" (tj. "Y" nebo "J"), stanice se znova rozeběhne a soubory DUP.SYS a DOS.SYS se na disketu přepíše.

I — FORMAT DISK (Formátování)

FORMATIEREN

Před tím, než disketa může uchovávat data a programy, musí být v ATARI-formátu rozdělena na stopy a sektory. Při tom jsou všechna na disketě dosud uchovávaná data vymazána! Ubezpečte se proto před formátováním, zda jste do stanice vložili správnou disketu! DOS žádá číslo disketové stanice. Jeho další otázku "pro jistotu" můžete zodpovědět kladně

"Y/J" jediné po nové kontrole, že je zasunuta správná disketa. I zkušení programátoři si při formátování smazali svoje důležité a vzácné diskety! Plná disketa obsahuje 126 250 znaků, což odpovídá mnoha dnům pilné práce! Proces formátování trvá asi 20 sekund. Disketová stanice se rozběhne a slyšíte tiché cvrkání. Jestliže stisknete během formátování klávesu (BREAK), můžete opět zadávat příkazy, ačkoliv disketová stanice formátuje dále. Příkaz "I" formátuje diskety do formátu pro stanici 1050 na využitelnou kapacitu cca 127 k bytů, přesněji využívá sektory 1 až 1010 s vynecháním sektoru 720. Zbylé sektory do 1040 jsou, zjednodušeně řečeno, rezervovány pro DOS. Formát nelze přecházet na stanici typu 810! Chcete-li mít diskety čitelné i na tomto typu, použijte k formátování příkaz "P".

J - DUPLICATE DISK (Kopírovat disketu - celou) **DISK KOPIEREN**

Tento příkaz slouží k okopírování celé diskety, přičemž se originál kopíruje sektor za sektorem. Cílová disketa je před kopírováním automaticky formována ve formátu originálu. Proto nelze kopírovat z vyšší psací hustoty (stanice 1050) na nižší hustotu (stanice 810). Některé diskety jsou proti kopírování chráněny výrobcem. Dbejte proto doporučení v návodech k programům na takových disketách.

K - BINARY SAVE (Ukládání binárních dat) **BINAER SAVE**

Ke správnému využití tohoto příkazu musíte ovládat dokonale programování ve strojovém kódu. Pro programy v BASICu není tento příkaz použitelný!

Příkazem můžete ukládat na disketu programy a data ve strojovém kódu (kupř. nově definované soubory znaků). DOS se ptá na další dodatečná zadání, která musíte od sebe oddělovat čárkami:

SAVE-GIVEFILE,START,END,(,INIT,RUN)
nebo v německé verzi:
DATEINAME,START,ENDE,(,INIT,RUN)

Nejdříve zadáte jméno zvoleného souboru. Za **START** dosadíte hexadecimální adresu, na které blok dat začíná a za **END** adresu posledního bytu. **INIT** a **RUN** mají význam jen při strojovém kódu a nezadávají se vždy. Adresa **INIT** (hex) se přenáší do buněk paměti **S02E2** a **S02E3**. Po natažení bloku souboru, jestliže byla zadána adresa **INIT**, volá DOS přes tento vektor vaši subroutinu, která musí být ukončena **RTS**. DOS poté ukládá následující blok, ke kterému opět může být zadána **INIT**-adresa. Tento postup se opakuje, až je dosaženo konce souboru.

Adresa **RUN** (hex) se přenesení do vektoru **S02E0** a **S02E1**. Po natáhnutí všech bloků souboru startuje DOS váš program přes tento vektor. K tomu příklad:

Vytvořili jste assemblerem, editorem nebo jinak program, který pokrývá rozsah paměti od \$6400 do \$7000 a jehož startovací adresa je \$6600. Program má dostat jméno **PROG.COM**. Po zavolání DOSu a zvolení příkazu "K" zadáte následující řádku:

D1:PROG.COM,6400,7000,,6600

DOS zajistí žádaný rozsah paměti na disketě ve stanici 1. **INIT**-adresa nebyla zadána, proto po koncové adrese následují dvě čárky.

Úvodní program k tomuto programu (kupř. titulní obrázek) můžete uložit na disketu obdobným způsobem. Předpokládejme, že úvodní program obsazuje paměť od \$0600 do \$0680. Startovací adresa je \$0600. Zadávací řádka vypadá pak následovně:

D1:TITOB.COM,600,680,600

V tomto případě je startovací adresa (\$0600) dosazena jako INIT-adresa a RUN-adresa je vynechána. Příkazem "C" spojíte oba souborové bloky do jediného souboru:

C

COPY—FROM,TO

nebo v německé verzi: KOPIE—VON,AUF

D1:TITOB.R.COM,D1:PROG.COM/A

Soubor TITOB.R.COM nyní sestává ze dvou bloků (\$0600—\$0680 a \$6400—\$7000), které lze z diskety do počítače natáhnout příkazem "L". Nejdříve je natažen a proveden úvodní program a po něm program hlavní.

L — BINARY LOAD (Natažení binárního programu z diskety)

BINAER LOAD

Příkazem jsou přetaženy z diskety do operační paměti počítače programy ve strojovém kódu a jiné soubory, které byly uloženy na disketu příkazem "K". Programy ve strojovém kódu jsou ihned automaticky odstartovány, jelikož obsahují INIT-a/ nebo RUN-adresu. Doplnkem "/N" za jménem souboru automatickému startu programu zabráníte. Po natažení binárního programu do počítače se objeví opět DOS menu.

M - RUN AT ADDRESS (Start od adresy)

START BEI ADR.

Příkazem můžete startovat programy, které byly nataženy příkazem "L" avšak neobsahují startovací adresu. Startovací adresa se zadává hexadecimálně po vyžádání DOSem. Následující příklad ilustruje použití příkazu pro docílení tzv. studeného startu systému, kterým se ATARI přivede do stavu, v kterém se normálně nachází po zapnutí:

M

RUN FROM WHAT ADDRESS?

německy: START—ADRESSE (HEX)?

E 477

N - CREATE MEM.SAV (Vytvořit MEM.SAV)

MEM.SAV ANLEGEN

Zavoláte-li z BASICu DOS, přepíše menu-program DUP.SYS dolní část paměti pro BASICovské programy a tak poruší váš BASICovský program. Zabráníte tomu tak, že na pracovní disketu vložíte soubor MEM.SAV. Před tím než DOS přetáhne menu do paměti, přesvědčí se, je-li na disketě tento soubor. Je-li tam, zajistí DOS tu část paměti, kterou svým menu přepisuje, na disketu a teprve potom menu-program přetáhne do počítače. Poté můžete provádět téměř všechny příkazy DOSu, aniž by se něco stalo s vaším BASICovským programem. Po návratu k BASICu příkazem "B" přetáhne DOS zpět ochráněný obsah části paměti do počítače a můžeme opět s uchovaným BASICovským programem normálně pracovat. Program MEM.SAV pracuje jen s disketovou stanicí 1, či se 130XE RAM-diskem (viz RAM-disk).

O - DUPLIKATE FILE (Duplikovat soubor)

DATEI DUPLIZ.

Příkaz slouží ke kopírování souboru na jediné disketové stanici. Poté co zadáte jméno souboru, který se má kopírovat, požaduje DOS vložení diskety s tímto originálem do stanice. Soubor je přetažen do paměti počítače. Pak se podle instrukcí DOSu vyjme zdrojová disketa ze stanice a vloží cílová disketa, na kterou se má soubor okopírovat. Podle délky programu a podle obsazení paměti může se tato výměna disket dít několikrát. Použití "divokých karet" je samozřejmě dovoleno.

P - FORMAT SINGLE (Formátovat pro stanici typu 810)

FORMATIERE 707

Příkaz pracuje stejně jako příkaz "I", formát má však jednoduchou hustotu (single density). Uživatelé je, jak na stanici 1050, tak na stanici 810, přístupno na takto naformátované disketě 707 sektorů, sektory 708 až 719 jsou rezervovány pro DOS. Další poznámky k tomuto formátu naleznete v popisech příkazů "A" a "I".

RAM-DISK

Počítač ATARI 130XE je vybaven RAM pamětí 128KB. Tato paměť je rozdělena na dva bloky po 64K. První blok tvoří systémovou paměť a je, jako u série 600/800XK, k dispozici pro programy. Zbýlé 64K mohou být použity jako čtyři paměťové bloky po 16K. Podrobnosti o tom jsou v příručce počítače.

Při natažení DOSu z diskety se automaticky startuje program RAMDISK, který umožňuje využit druhých 64K paměti jako tzv. RAM-disk. Systémové soubory DUP.SYS a MEM.SAV se automaticky do něho přetáhnou. Na RAM-disku zbyde ještě 412 sektorů k dalšímu použití. Zavoláte-li nyní DOS, váš BASICovský program se uchová v souboru MEM.SAV a do paměti se přetáhne menu-program DUP.SYS z RAM-disku. Celá procedura trvá díky rychlým přenosům necelou sekundu.

Pseudodisketovou stanici RAM-disk adresujte jako stanici 8, kupř. LOAD"D8:PROG.BAS". Využije se dobře jako rychlá dočasná paměť pro programy a data. Je třeba mít však na paměti, že všechny tam uložené informace jsou při vypnutí počítače ztraceny. Jejich trvalé uchování v RAM-disku není proto možné.

RAM-disk rovněž ulehčuje práci s DOsem, kupř. při kopírování:

Chcete třeba zkopírovat originál DOSu na disketu a máte jen jedinou disketovou stanici. S příkazy "J" nebo "O" byste museli několikrát měnit diskety. RAM-disk vám tuto výměnu ušetří. Nejdříve zformátujte prázdnou disketu, jak je popsáno pod "I" a instalujte na ni příkazem "H" DOS soubory. Poté vložte opět DOS 2.5 disketu do stanice a zkopírujte příkazem "C" všechny soubory s přídavkem ".COM" na RAM-disk:

SELECT ITEM OR RETURN FOR MENU

v německé verzi: BITTE WAEHLEN,RETURN->MENUE

C

COPY--FROM,TO?

v německé verzi: KOPIE-VON,AUF?

D1 :*.COM,D8:

Tyto čtyři soubory jsou nyní jako kopie na RAM-disku. Vyjměte originální disketu DOSu ze stanice a vložte tam disketu předem upravenou "I" a "H". Z RAM-disku přeneste čtyři soubory DOSu na tuto novou disketu:

SELECT ITEM OR RETURN FOR MENU

v německé verzi: BITTE WAEHLEN,RETURN- MENUE

C

COPY--FROM,TO?

v německé verzi: KOPIE-VON,AUF?

8:*.COM,1:

Vynechání znaku "D" před číslem stanice je přípustné. Nová pracovní disketa DOSu, kterou jste takto vytvořili, obsahuje všechny soubory původní DOS diskety a originální disketu můžete uschovat.

Příkaz "J" (Kopírovat disketu - celou) NEpracuje s RAM-diskem, protože jeho kapacita nepostačuje pro obsah jedné celé diskety.

Smažete-li na RAM-disku příkazem "D" menu-program DUP.SYS, neexistuje pak žádná

možnost pracovat přímo s DOSem, protože BASICovský příkaz "DOS" neúčinkuje. V tomto případě systém vypnete a zapnete a RAM-disk poté znovu instalujete. Všechny ostatní BASICovské funkce, které zasahují do RAM-disku, pracují bez souboru DUP.SYS. Chcete-li využít přídavnou paměť jiným způsobem, vymažte soubor RAMDISK.COM z pracovní diskety nebo změňte příkazem "E" jeho jméno (kupř. na "RAMDISK.VYP").

POMOCNÉ PROGRAMY (UTILITY) v DOSu 2.5

Jsou to pomocné programy pro řízení systému a pro programování. Na originální DOS 2.5 disketě jsou tři takové programy:

- * COPY32.COM kopíruje a transformuje data zapsaná ve formátu DOSu 3 do formátu DOSu 2.5.
- * DISKFIX.COM pomáhá při různých problémech, které se mohou vyskytnout při použití DOSu 2.5.
- * SETUP.COM dovoluje změnu důležitých parametrů DOSu 2.5.

Přetažení pomocného programu:

Programy jsou v podobě binárního souboru a jsou volatelné příkazem "L" z DOS menu. Chcete-li kupř. pracovat s programem COPY32.COM, zadáte příkaz "L" a poté jméno souboru COPY32.COM.

COPY32.COM

Program pracuje stejně jako kopírovací příkaz z DOS menu, ale kopíruje mezi dvěma úplně rozdílnými formáty. Po přetažení budete vyzváni, abyste zadali čísla stanic, ve kterých jsou zasunuty diskety v DOSu 3 ("FROM" resp. "VON") a v DOSu 2.5 ("TO" resp. "AUF"). Při jediné disketové stanici zadáte jen "1" - budete pak diskety měnit. COPY32 vám ukáže adresář, podle kterého vyberete soubor ke kopírování. Tento soubor je přetažen do počítače. Při provozu s jedinou disketovou stanicí musíte zaměnit diskety a odstartovat tlačítkem (START). Z paměti počítače přenáší COPY32 soubor na disketu ve formátu DOSu 2.5. Při delších souborech je nutné diskety několikrát sobě měnit. Soubory, jejichž délka překročí 124700 Bytů umí COPY32 konvertovat jen na dvou disketových stanicích. Vyskytne-li se při kopírování chyba, vydá program chybové hlášení a přeruší svoji činnost.

DISKFIX.COM

Program má menu, podobně jako DOS 2.5. Lze volit jeden z pěti příkazů:

1. CHANGE DRIVE * (Číslo disketové stanice)
LAUFWERKS *
2. UNERASE FILE (Nemazat)
UNLOESCHEN
3. VERIFY DISK (Disketu ověřit)
DISK PREUFEN
4. RENAME FILE BY (Jméno souboru změnit podle čísla)
NAME NACH NUMMER AENDERN
5. QUIT - RETURN TO DOS (Zpět na DOS)
DOS

1. CHANGE DRIVE (Číslo disketové stanice)
LAUFWERKS

Tímto příkazem určujete, s kterou disketovou stanicí má DISKFIX pracovat. Stisknete-li klávesu příslušné číslice (1, 2, 3, 4 nebo 8), vrátíte se zpět do menu, kde DISKFIX toto nové číslo stanice ukáže. Zadání nesprávného čísla není programem přijato.

2. **UNERASE FILE** (Nemazat) **UNLOESCHEN**

Chybovat je lidské! Jestliže jste smazali nedopatřením nějaký soubor, můžete jej případně tímto příkazem "2" znovu vytvořit za předpokladu, že jste mezitím na disketu nezapisovali. DOS 2.5 označí mazaný soubor v adresáři diskety a dá sektory tohoto programu opět k dispozici pro zápis. Soubor je ve stavu vymazání ještě kontaktovatelný. Jestliže jste však sektory označeného souboru použili pro zápis dalšího souboru, je označený soubor nenávratně ztracen.

DISKFIX také může zprostředkovat přístup k souborům, které jste jednou otevřeli, ale později neuzavřeli třeba tím, že jste nedopatřením přerušili běh programu. Po zvolení příkazu "2" vložte disketu s postiženými daty do disketové stanice a stiskněte tlačítko **START**. Poté se na stínítku objeví seznam prvních 32 souborů na disketě. Mazané soubory poznáte lehce podle písmena "D" před jménem souboru, otevřené soubory jsou označeny písmenem "W". Zadejte nyní číslo souboru, který chcete znova zpracovat, nebo stiskněte jen **RETURN** — pak se vám ukáže dalších 32 souborů. DISKFIX vám dá ke zvolenému souboru další informace, které jsou důležité pro programátora a táže se vás, jestli chcete tento soubor opět vytvořit. Po vašem zadání prověřuje DISKFIX soubory na disketě (viz příklad "3"). Nakonec se soubor opět v normálním stavu objeví v adresáři diskety a je možno jej znovu vyvolat.

3. **VERIFY DISK** (Disketu ověřit) **DISK PRUEFEN**

DOS 2.5 buduje na každé disketě tabulku přístupných, tj. volných sektorů, která se jmenuje VTOC (Volume Table Of Contents). VTOC se při každé operaci zápisu aktualizuje, pokud se ovšem nevyskytne chyba. Příkaz "3" přezkoušuje tuto tabulku podle pamatovaných dat a přepisuje ji. Otevřené soubory obdrží při tom status "vymazáno" a jejich sektory jsou dány k dispozici.

4. **RENAME FILE BY #** (Změnit jméno souboru podle čísla) **NAME NACH NUMMER AENDERN**

V případě, že některé soubory mají stejná jména (viz DOS 2.5, příkaz "E") můžete s DISKFIXem příkazem "4" změnit jména dvou souborů. DISKFIX vám opět ukáže adresář diskety, z kterého si pomoci odpovídajícího čísla vyberte příslušný soubor. Odpovíte-li na otázku "THIS FILE", resp. "DIESES FILE", ("Tento soubor") kladně, tj. "Y", resp. "J", otáže se vás DISKFIX po novém jménu. Respektujte doporučení pro tvoření jmen souborů — viz odstavec "Jména souborů" v úvodu tohoto popisu.

5. **QUIT — RETURN TO DOS** (Zpět na DOS) **DOS**

Příkaz "5" přerušuje činnost DISKFIXu a přivádí na obrazovku menu DOSu 2.5.

SETUP.COM

Po natažení pomocného programu SETUP.COM se objeví následující menu:

1. CHANGE CURRENT DRIVE NUMBER (Změnit číslo disketové stanice)
LAUFWERKS NUMMER AENDERN
2. CHANGE SYSTEM CONFIGURATION (Změnit konfiguraci systému)
SYSTEM-EINSTELLUNG AENDERN
3. SET UP AN AUTORUN FOR BOOT (Zřídít soubor AUTORUN)
AUTORUN DATEI ERSTELLEN
0. QUIT — RETURN TO DOS (Zpět na DOS)
ZURUECK ZUM DOS

1. **CHANGE CURRENT DRIVE NUMBER** (Změnit číslo disketové stanice)
LAUFWERKS-NUMMER AENDERN

V textu menu je sděleno, na kterou disketovou stanici příkazy účinkují. Příkazem "1" můžete změnit číslo této stanice (číslo 1 až 4). RAM-disku na 130 XE nelze použít!

2. **CHANGE SYSTEM CONFIGURATION** (Změnit konfiguraci systému)
SYSTEM-EINSTELLUNG AENDERN

Po zvolení příkazu se na stínítku objeví dosavadní stav systému:

CHANGE SYSTEM CONFIGURATION (Změnit konfiguraci systému)
ACTIVE DRIVES: 12 (Aktivní disketové stanice)
UP TO 3 FILES OPEN SIMULTANEOUSLY (Max. 3 soubory současně otevřeny)
DISK WRITES OCCUR WITH VERIFY (Zápis je s ověřením)
DO YOU WANT TO CHANGE ANY PART OF THAT CONFIGURATION (Y/N)? (Chcete změnit tuto konfiguraci?)

V německé verzi:

SYSTEM-EINSTELLUNG AENDERN (Změnit konfiguraci systému)
AKTIVE LAUFW.: 1 2 (Aktivní disketové stanice)
MAX. 3 DATEIEN GLEICHZEITIG OFFEN (Max. 3 soubory současně otevřeny)
SCHREIBEN ERFOLGT MIT PRUEFUNG (Zápis je s ověřením)
WOLLEN SIE DIESE EINSTELLUNG VERAENDERN (J/N)? (Chcete změnit tuto konfiguraci?)

Zadáte-li nyní "5", táže se SETUP po nových hodnotách nastavení.

Každá ohlášená disketová stanice vyžaduje 144 bytů, a každý otevřený soubor 128 bytu v RAM paměti. Ohlášenou stanicí ztratíte sice malou část paměti, získáte však mnoho místa na disketě. Poznamenejme, že se čtyřmi stanicemi a RAM-diskem nebo jen s jediným otevřeným souborem nepracují příkazy DOSu "C", "D" a "O" bez námitek. Program SETUP vás tak varuje před možnými chybami.

Při zápisu na disketu zkouší DOS, zda data byla zapsána bez chyby. Chyby při přenosu se vyskytují jen zřídka, ale jediný chybný byte zničí celý program. Pro časovou úsporu můžete přezkoušení přenášených dat vypnout, lze to doporučit však jen na datových disketách (texty atd.). Po zadání ukáže SETUP novou konfiguraci systému, kterou v případě souhlasu potvrdíte "Y", resp. "J" ("Ano"). Můžete se rovněž rozhodnout, zda toto nastavení DOSu uložíte na disketu.

3. **SET UP AN AUTORUN FOR BOOT** (Zřídít soubor AUTORUN)
AUTORUN DATEI ERSTELLEN

Při zapnutí počítače se přetáhne DOS 2.5 do paměti a na disketě se hledá soubor AUTORUN.SYS. Je-li tento soubor po ruce, pak je natažen a odstartován před tím, než je systémová kontrola předána na BASIC. Příkazem "3" můžete zřídit takový soubor, který:

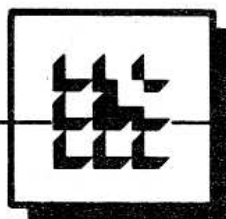
- x Inicializuje driver pro interfejsový modul ATARI 850 (pokud je připojen).
- x Přetáhne do paměti BASICovský program a odstartuje ho.
- x Proveďte oba úkony najednou.

Zvolíte-li příkaz "2" nebo "3", táže se SETUP po jménu BASICovského programu a přezkoušuje po zadání, zda tento program na disketě existuje. Poté je soubor AUTORUN.SYS přepsán na disketu a opět se ukáže menu.

0. QUIT - RETURN TO DOS (Zpět na DOS)

ZURUECK ZUM DOS

Přerušeni činnosti programu SETUP.COM a návrat menu DOSu na obrazovku.



TECHNICKÉ novinky

Telefax v ČSSR

Službu TELEFAX zavede od 1. ledna 1988 Správa spojů. V jejím rámci mohou telefonní účastníci, jejichž stanice je vybavena schváleným faximilním zařízením, přenášet ve vnitrostátním i mezinárodním styku obrazové a textové informace s jinými účastníky této služby. Bližší informace poskytují mezinárodní telefonní a telegrafické ústředny v Praze a Bratislavě, ředitelství telekomunikací Bratislava a ředitelství spojů v jednotlivých krajích.

(Rudé právo, 31. 12. 1987, s.2)

PŘÍPRAVA ZAVÁDĚNÍ TELEVIZNÍ INFORMAČNÍ SLUŽBY

Po zásadním rozhodnutí o převzetí soustavy WST s úrovní 1,5 (existující čtyři úrovně - 1, 1,5, 2 a 3 - se rozlišují podle složitosti a náročnosti), která je vhodná pro české a slovenské texty s úplnými diakritickými znaménky, zahájil čs. průmysl jednání s firmou Philips-Mullard. Ta má dodávat příslušné integrované obvody i celé dekodéry do televizních přijímačů, s jejichž tuzemskou výrobou se počítá od roku 1990. Zkušební vysílání má být zahájeno v roce 1988 v síti druhého televizního programu. Na přípravě se podílí ministerstvo spojů, Čs. televize a čs. elektronický průmysl.

(QUO VADIS, elektronika '88, OIS VTEI FMEP, Praha 1988, s. 108-110)

Veškeré vstupní a výstupní operace v počítačích ATARI jsou řízeny prostřednictvím řídicích bloků IOCB, očíslovaných 0 až 7 a nacházejících se ve třetí stránce paměti RAM od adresy \$0340 /832/. Bloky jsou zpracovávány procedurou CIOV začínající na adrese \$E456 /58454/, která je vstupním rozhraním mezi CIO (Central I/O Utility/ a uživatelem. CIO zpracovává požadavek uživatele a provádí výkonné operace na definovaném zařízení.

Bloky IOCB 0, 6 a 7 jsou používány operačním systémem:

IOCB0 — obsluha obrazovkového editoru,

IOCB6 — výstup na obrazovku v grafických modech,

IOCB7 — operace LOAD, SAVE, LIST, CLOAD, CSAVE a LPRINT.

Na tuto skutečnost je třeba pamatovat při volbě čísla kanálu, který hodláme v programu používat.

Každý z bloků obsahuje 16 bytů paměti, jejichž význam je následující:

BLOKY IOCB A CO S NIMI

JABS



Adresa v IOCB	Název	Význam
0	ICHID	index do systémové tabulky zařízení, hodnota dosazena CIO ze systémové tabulky jako výsledek operace OPEN a není měněna dokud je kanál otevřen; po uzavření kanálu ICHID = \$FF
1	ICDNO	číslo zařízení, dosazené CIO jako výsledek operace OPEN, odlišuje zařízení stejného typu (např. D1: a D2:)
2	ICCMD	kód příkazu, který má vykonat CIO
3	ICSTA	status — je aktivován po proběhnutí každé operace CIO a signalizuje svým MSB (tj. 7 bitem) výsledek provedené operace, (1 = chyba, 0 = bez chyby), zbývající bity představují číslo chyby
4, 5	ICBAL ICBAH	počáteční adresa bufferu obsahujícího: — data, jedná-li se o čtení/zápis, — jméno zařízení/jméno souboru, jde-li o OPEN
6, 7	ICPTL ICPTH	počáteční adresa procedury PUT BYTE příslušného driveru nastavená CIO během operace OPEN
8, 9	ICBLL ICBLH	délka datového bufferu pro čtení/zápis, aktualizována po každé vykonané operaci CIO
10,	ICAX1	pomocné parametry pro OPEN nebo upřesnění

11	ICAX2	požadavku na výkonné zařízení; pro OPEN jsou v ICAX1 vyhrazeny: bit2 pro vstup a bit3 pro výstup (otevření kanálu je charakterizováno hodnotou 1)
12 až 15	ICAX3 až ICAX6	pomocné parametry vyhrazené CIO, závislé na zařízení a prováděné operaci

Z hlediska uživatele jsou tedy zajímavé tyto parametry:

ICCMD — Command Byte

ICBAL, ICBAH — Buffer Address (adresa bufferu)

ICBLI, ICBLH — Buffer Length/Byte Count (délka bufferu/počet bytů)

ICAX1, ICAX2 — Auxiliary Information (pomocná informace), výjimečně ICAX3—ICAX6

Tyto parametry je nutno do IOCB dodat. Zajímat nás bude ICSTA — Status, coby výsledek provedené operace.

ICCMD — Command Byte

Operace	ICCMD	Operace	ICCMD
OPEN	\$03	DRAW	\$11
GET RECORD	\$05	FILL	\$12
GET CHARACTER	\$07	RENAME	\$20
PUT RECORD	\$09	DELETE	\$21
PUT CHARAKTER	\$0B	LOCK	\$23
CLOSE	\$0C	UNLOCK	\$24
GET STATUS	\$0D	NOTE	\$25
		POINT	\$26
		FORMAT	\$FE

ICAX1 — ICAX6 — Auxiliary Information

Popis provedeme podle jednotlivých zařízení:

1. Klávesnice /K:/

Přípustné operace:

OPEN, CLOSE, GET CHARACTER, GET RECORD a GET STATUS

Doplňující informace nevyžaduje.

2. Display /S:/

Přípustná operace:

OPEN, CLOSE, GET CHARACTER, GET RECORD, PUT CHARAKTER, PUT RECORD, GET STATUS, DRAW, FILL.

Pro operaci OPEN:

ICAX1 — bit 4 = 1 nemaže obrazovku při OPEN

bit 5 = 1 grafická obrazovka s textovým okénkem

ICAX2 — bity 0—3 mód obrazovky (0—11)

DRAW — speciální příkaz, vykreslí spojnici mezi okamžitou polohou kurzoru do pozice určené ROWCRS (\$0054) a COLCRS (\$0055) barvou uloženou v ATACHR (\$02FB)

FILL — speciální příkaz, vyplní vymezenou oblast určenou barvou a definovaných znakem, zadání je obdobné jako u DRAW. ROWCRS, COLCRS obsahují souřadnice koncového bodu, FILDAT (\$02FD) obsahuje datovou předlohu.

3. Obrazovkový editor (E:)

Připustné operace:

OPEN, CLOSE, GET CHARACTER, GET RECORD, PUT CHARACTER, PUT RECORD, GET STATUS.

Pro operaci OPEN:

ICAX1 — bit0 = 1 tzv. "forced read", tj. okamžité čtení dat po provedení OPEN.

4. Magnetofon (C:)

Připustné operace:

OPEN, CLOSE, GET CHARACTER, GET RECORD, PUT CHARACTER, PUT RECORD, GET STATUS.

Pro operaci OPEN:

ICAX2 — bit 7 = 1 kontinuální čtení/zápis.

5. Tiskárna (P:)

Připustné operace:

OPEN, CLOSE, PUT CHARACTER, PUT RECORD, GET STATUS.

Pro operaci PUT CHARACTER, PUT RECORD:

ICAX2 — tiskový mód: \$4E — normální tisk

\$53 — boční tisk

\$53 — široký tisk.

6. Disketa (D:)

Připustné operace:

OPEN, FILE, OPEN DIRECTORY, CLOSE, GET CHARACTER, GET RECORD, PUT CHARACTER, PUT RECORD, GET STATUS, NOTE, POINT, LOCK, UNLOCK, DELETE, RENAME, FORMAT.

OPEN FILE

OPEN Input — ICAX1 = \$04 — otevření souboru pro vstup

OPEN Output — ICAX1 = \$08 — otevření souboru pro výstup počínají 1. bytem souboru (nechráněného)

OPEN Append — ICAX1 = \$09 — otevření souboru pro výstup počínaje bytem následujícím za posledním bytem již existujícího souboru

OPEN Update — ICAX1 = \$0C — otevření souboru pro aktualizaci pomocí GET, PUT, NOTE a POINT

OPEN DIRECTORY

ICAX1 = \$06

NOTE — vrací umístění následujícího bytu, který bude čten (zapsán)

ICAX3 — LSB sektoru diskety

ICAX4 — MSB sektoru diskety

ICAX5 — relativní umístění bytu v sektoru

POINT — zadává umístění následujícího bytu, který bude aktualizován

ICAX3 — LSB sektoru diskety

ICAX4 — MSB sektoru diskety

ICAX5 — relativní umístění bytu v sektoru

ICSTA — CIO Status Byte

\$01	001	operace bez chyb
\$80	128	přerušeni klávesou BREAK
\$81	129	pokus operace OPEN na otevřeném kanálu
\$82	130	neexistující zařízení
\$83	131	otevřeno pouze pro zápis
\$84	132	neplatný příkaz
\$85	133	zařízení/soubor není otevřen
\$86	134	neplatné číslo kanálu IOCB
\$87	135	otevřeno pouze pro čtení
\$88	135	konec souboru (EOF)
\$89	137	zkomolený zápis
\$8A	138	timeout (zařízení neodpovídá)
\$8B	139	zařízení nepřijímá
\$8C	140	chyba na vstupu sériové sběrnice
\$8D	141	kurzor mimo rozsah
\$8E	142	zahlcení datové sériové sběrnice
\$8F	143	chyba kontrolní sumy datové sériové sběrnice
\$90	144	chyba způsobená zařízením
\$91	145	chybný mód obrazovky
\$92	146	nepřípustná operace pro volaný driver
\$93	147	nedostatek paměti pro požadovaný mód obrazovky
\$A0	160	chyba v čísle diskety
\$A1	161	příliš mnoho otevřených diskových souborů
\$A2	162	plná disketa
\$A3	163	chyba diskety
\$A4	164	pomíchané soubory diskety
\$A5	165	chybný název souboru
\$A6	166	chyba v délce dat pro POINT
\$A7	167	soubor blokován
\$A8	168	nepřístupný příkaz diskety
\$A9	169	plný adresář (64 souborů)
\$AA	170	soubor nenalezen
\$AB	171	chybná operace POINT

Hodnoty do IOCB bloků lze ukládat buď příkazy POKE, nebo procedurou ve strojovém kódu. Samotvé provedení operace je možné pouze ve strojovém kódu. Proceduru lze však vytvořit řetězcem v BASICu. Označme pro-

ceduru jako CIO a uveďme několik příkladů. Při programování v assembleru zadáváme číslo kanálu vynásobené 16ti do X registru procesoru.

```
<CIO> A$="h_" Vd♦":A$(3,3) =CHR$(IOCB*16) :A=USR(ADR(A$))
```

zdrojový text procedury

```
104      PLA
162 xxx   LDX # IOCB*16
32  86 228 JSR CIOV
96       RTS
```

Příklad 1: CLOSE#1

BASIC: IOCB=1	ASS: LDX#16
POKE 832+IOCB*16+2,12	LDA#12
GOTO CIO	STA 834,X
	JSR 58454

Příklad 2: OPEN# 3,4,0,"C:"

BASIC: B\$="C:"	ASS: LDX#48
AH=INT(ADR(B\$)/256)	LDA#3
AL=ADR(B\$)-AH*256	STA 834,X
IOCB=3	LDA#AL
POKE 832+IOCB*16+2,3	STA 836,X
POKE 832+IOCB*16+4,AL	LDA#AH
POKE 832+IOCB*16+5,AH	STA 837,X
POKE 832+IOCB*16+8,2	LDA#2
POKE 832+IOCB*16+9,0	STA 840,X
POKE 832+IOCB*16+10,4	LDA#4
POKE 832+IOCB*16+11,0	STA 842,X
GOTO CIO	LDA#0
	STA 841,X
	STA 843,X
	JSR 58454

Příklad 3: Zápis obsahu SM (Screen Memory) v módu 0 na magnetofon po otevření kanálu IOCB5

BASIC: IOCB=5	ASS: LDX#80
POKE 832+IOCB*16+2,11	LDA#11
POKE 832+IOCB*16+4,PEEK(88)	STA 834,X
POKE 832+IOCB*16+5,PEEK(89)	LDA 88
POKE 832+IOCB*16+8,224	STA 836,X
POKE 832+IOCB*16+9,3	LDA 89
GOTO CIO	STA 837,X
	LDA#224
	STA 840,X
	LDA#3
	STA 841,X
	JSR 58454

kde $224+3 \times 256=992$ počet bytů SM v 0

Zpracováno podle zahraničních pramenů - Zientara, W.; BAJTEK 2/1987, s.8.





TIPY TRIKY

Pomůcky pro programování

ing. Petr Jandík, Praha

Výpisy souboru

Často používanou operací je výpis souboru. Následující program poslouží k nej-různějším účelům.

Funkce 1 vypisuje soubor ve formě hexadecimálních listingů, používaných časopisem Analog Computing, včetně kontrolních součtů. Tyto součty lze zkontrolovat programem **Unicheck**. Funkci lze použít nejen k zaslání programu do zmíněného časopisu, ale po odstranění kontrolních součtů můžeme tímto způsobem generovat do svých programů v BASICu strojové podprogramy, obrázky, alternativní znakové sady nebo jiná data.

Funkce 2 slouží k témuž účelu při programování v assembleru. Převádí jakákoliv binární data do tvaru zdrojového textu, přičemž se předpokládá, že data jsou opatřena na začátku počáteční a koncovou adresou podle konvencí DOSu. Program **Filedump** může třeba posloužit, když si některým ze znakových editorů vytvoříme vlastní znakovou sadu a potřebujeme ji začlenit ve zdrojové formě do svého programu.

Poznámka k programu: ve špičatých závorkách uvádím kombinaci potřebnou k napsání speciálního znaku. Jedná se o řádky 1360 a 3080.

Program FILEDUMP

```
10 -----
20 REM Vypisy souboru ve forme
30 REM hexa nebo ascii povelu DATA
40 -----
50 EXEC UVOD
55 DO
60   GET #2,BYTE
70   IF HEDEC=1
72     IF KONVERT=1
74       IND=BYTORG+BYTCNT*2
76     ELSE
80       IND=BYTORG+BYTCNT*4
82     ENDIF
85     LIN$(IND,IND+1)=HEX$(BYTE)
90   ELSE
100    IND=BYTORG+3*BYTCNT
110    LIN$(IND,IND+2)="000".POM$=STR$(BYTE)
120    LIN$(IND+3-LEN(POM$),IND+2)=POM$
130  ENDIF
140  BYTCNT=BYTCNT+1
```

```

150 ON KONVERT EXEC CKS1,CKS2
160 IF BYTENT=BYTLIN
170 ON KONVERT EXEC NLN1,NLN2
180 LIN$=INIT$:BYTENT=0
190 ENDIF
200 IF OUTFIL$="E:" OR OUTFIL$="P:"
210 IF INKEY#CHR#($1F+128)
220 GET KEY
230 ENDIF
240 ENDIF
250 LOOP
1000 -----
1010 REM proc UVOD - zadani parametru
1020 -----
1030 PROC UVOD
1035 CLOSE
1040 DIM FILE$(15),OUTFIL$(15),POM$(15):HEDEC=0
1050 CLS :POSITION 11,3:? "F I L E D U M P"
1055 POSITION 8,5:? "(c) 1986 Petr Jandik"
1060 POSITION 0,7:? "-----"
1070 POSITION 6,9:INPUT "Jmeno souboru:";FILE$
1080 I=LEN(FILE$)
1090 IF I<1 THEN GOTO 1050
1100 IF I<3
1110 EXEC DOPFILEN
1120 ELSE
1130 IF FILE$(2,2)<>" " OR FILE$(3,3)<>" "
1140 EXEC DOPFILEN
1150 ENDIF
1160 ENDIF
1170 POSITION 6,10:INPUT "Vystup:";OUTFIL$
1180 IF OUTFIL$=""
1190 OUTFIL$="E:"
1200 ELSE
1210 I=LEN(OUTFIL$)
1220 IF I<2
1230 EXEC DOPOUTFIL
1240 ELSE
1244 IF I=2
1245 IF OUTFIL$(2,2)<>" "
1246 EXEC DOPOUTFIL
1247 ENDIF
1248 ELSE
1250 IF OUTFIL$(2,2)<>" " OR OUTFIL$(3,3)<>" "
1260 EXEC DOPOUTFIL
1270 ENDIF
1280 ENDIF
1290 ENDIF
1295 ENDIF
1300 ? "Druh vypisu: 1 - Analog Comp.,hexa"
1310 POSITION 15,12:? "2 - Asmblr tab,hexa"
1360 POSITION 15,19:INPUT "<ESC/SHIFT - DELETE
BACKSPACE>Volba";KONVERT
1365 POKE #02F0,1
1370 POSITION 36,10+KONVERT:? "<=="
1380 IF KONVERT>2 THEN EXEC NENI:POSITION 36,10+KONVERT:? "
":POSITION 2,11:GOTO 1300
1383 TRAP #SYNAD:OPEN #2,4,0,FILE$
1384 OPEN #3,8,0,OUTFIL$

```

```

1390 ON KONVERT EXEC INIT1,INIT2
1410 LIN$=INIT$
1420 EOF=0:BYTCNT=0
1450 ENDPROC
1460 -----
1470 REM Inicializace pro Anal.Comp.he
1480 -----
1490 PROC INIT1
1500 BYTORG=11:LINLEN=104:BYTLIN=45:LNNUM=1000
1510 DIM INIT$(LINLEN),LIN$(LINLEN)
1520 LININC=10
1530 INIT$(1)=" ":INIT$(LINLEN)=" "
1540 INIT$(2)=INIT$
1550 INIT$(6,9)="DATA":INIT$(101,101)=","
1560 FOR I=13 TO 100:INIT$(I,I)="0"
1570 NEXT I
1580 CHECKS=0:HEDEC=1
1590 ENDPROC
1600 -----
1610 REM inicializace Assembler
1620 -----
1630 PROC INIT2
1640 BYTORG=14:LINLEN=43:BYTLIN=8
1650 LNNUM=1000:LININC=1
1660 DIM INIT$(LINLEN),LIN$(LINLEN)
1670 INIT$=" ".BYTE $00"
1680 FOR I=16 TO 40 STEP 4
1690 INIT$(I,I+3)=",$00"
1700 NEXT I
1710 HEDEC=1:LIN$=" "
1720 GET #2,BYTE:GET #2,BYTE:GET #2,I
1740 GET #2,BYTE:LIN$="900 *="
1750 LIN$(8)=HEX$(I+256*BYTE)
1760 ? #3;LIN$:GET #2,BYTE:GET #2,BYTE
1770 ENDPROC
1800 -----
1810 REM Doplneni D: do filem
1820 -----
1830 PROC DOPFILEM
1840 POM$=FILEM$
1850 FILEM$="D1:":FILEM$(4)=POM$
1860 ENDPROC
1870 -----
1880 REM doplneni D: do outfil
1890 -----
1900 PROC DOPOUTFIL
1910 POM$=OUTFIL$
1920 OUTFIL$="D1:":OUTFIL$(4)=POM$
1930 ENDPROC
1940 -----
1950 REM kontrolni soucet fce 1
1960 -----
1970 PROC CKS1
1980 CHECKS=CHECKS+BYTE
1990 IF CHECKS>999 THEN CHECKS=CHECKS-1000
2000 ENDPROC
2010 -----
2020 REM kontrolni soucet fce 2
2030 -----

```

```

2240 PROC CKS2
2250   REM NESCITA SE
2260 ENDPROC
2510 -----
2520 REM tisk radky fce 1
2530 -----
2540 PROC WLN1
2550   LIN$(1,4)=STR$(LNNUM)
2560   LNNUM=LNNUM+LININC
2570   LIN$(102)=STR$(CHECKS)
2580   ? #3;LIN$
2590 ENDPROC
2600 -----
2610 REM tisk radky fce 2
2620 -----
2630 PROC WLN2
2631   LIN$(1,4)=STR$(LNNUM)
2632   LNNUM=LNNUM+LININC
2640   ? #3;LIN$
2650 ENDPROC
2960 -----
2970 REM obslouzeni konce souboru
2980 -----
2990 PROC ENDATA
3000   IF BYTCNT>0
3010     ON KONVERT EXEC WLN1,WLN2
3020   ENDIF
3030   CLOSE :POSITION 4,20
3040   END
3050 ENDPROC
3060 -----
3070 # SYNAD:IF ERR=136 THEN EXEC ENDATA
3080   POSITION 4,18:?"<ESC/SHIFT - DELETE BACKSPACE>CHYBA
3090   ";ERR;" na radce ";ERL
3100 END

```

Větší programy se lépe editují a upravují pomocí textového editoru, jako je třeba SpeedScript. Při tom lze využít četné funkce, mezi nimi i globální hledání a nahrazování. Touto funkcí však lze i natropit mnoho škod, které budeme v rozsáhlém a nepřehledném programu těžko hledat.

Máme-li předchozí fungující verzi, pomůže nám program KOMPLI, který srovná oba texty podle čísel řádků a vypíše nám na obrazovku, resp. na tiskárnu všechny rozdíly. Program lze použít na jakýkoliv programovací jazyk, který používá čísla řádek. Pokud texty čísla řádek nemají, můžeme je přidat třeba programem ADDLN. V tomto případě má srovnání smysl je tehdy, když jsme nepřidávali ani neubírali řádky. Podmínkou je, že vstupní soubory musí být textové, což dosáhneme ve většině případů povelu LIST.

```

10 DIM A$(255),B$(255),INF1$(16),INF2$(16),L$(255)
15 INF1$="D:";INF2$="D:";TRAP 5020
20 ? "SROVNANI SOUBORU":INPUT "SOUBOR 1 ",L$:INF1$(3)=L$
30 INPUT "SOUBOR 2 ",L$:INF2$(3)=L$
40 INPUT "LISTING rozdilnych radek (E/P) ",L$:IF LEN(L$)=0 THEN
L$="E:"
50 CLOSE :OPEN #1,4,0,INF1$:OPEN #2,4,0,INF2$
60 IF L$<>"E:"
70 EDIT=0:OPEN #3,8,0,L$
71 ? #3;"<ESC/ESC><ESC/CTRL-N>
LISTING<ESC/ESC><ESC/CTRL-O>"
72 ? #3;"A = ";INF1$;"
B = ";INF2$
80 ELSE
90 EDIT=1
100 ENDIF
105 POC_ROZDILU=0
110 AKEY=0:EXEC READA
120 BKEY=0:EXEC READB
130 DO
135 IF AKEY=40000 AND BKEY=40000
137 IF EDIT=0 THEN ? #3;"pocet rozdilu = ";POC_ROZDILU:CLOSE
:END
138 IF EDIT=1 THEN ? "pocet rozdilu = ";POC_ROZDILU:END
139 ENDIF
140 AKEY=VAL(A$):BKEY=VAL(B$)
150 IF AKEY=BKEY
160 IF A$<>B$
180 EXEC LISTA:EXEC LISTB
190 ENDIF
195 EXEC READA:EXEC READB
197 POC_ROZDILU=POC_ROZDILU+1
200 ELSE
210 IF AKEY<BKEY
220 EXEC LISTA:EXEC READA
221 POC_ROZDILU=POC_ROZDILU+1
230 ENDIF
240 IF BKEY<AKEY
250 EXEC LISTB:EXEC READB
251 POC_ROZDILU=POC_ROZDILU+1
260 ENDIF
270 ENDIF
1000 LOOP
4100 -----
4110 PROC READA
4115 TRAP #ATEST
4120 INPUT #1,A$:GO# ENDA
4121 # ATEST
4122 IF ERR=136
4123 AKEY=40000
4124 ELSE
4125 EXEC ERROR
4126 ENDIF
4127 # ENDA
4130 ENDPROC
4200 -----
4210 PROC READB
4215 TRAP #BTEST
4220 INPUT #2,B$:GO# ENDB
4221 # BTEST
4222 IF ERR=136
4223 BKEY=40000
4224 ELSE

```

SROVNANI

```

4225      EXEC ERROR
4226      ENDIF
4227      # ENDB
4230 ENL PROC
4300 -----
4310 PROC LISTA
4320      IF EDIT=1
4330          ? "a";A$
4340      ELSE
4350          ? #3;"a";A$
4360      ENDIF
4370 ENDPROC
4400 -----
4410 PROC LISTB
4420      IF EDIT=1
4430          ? "b";B$
4440      ELSE
4450          ? #3;"b";B$
4460      ENDIF
4470 ENDPROC
5000 -----
5010 PROC ERROR
5020      ? "CHYBA C. ";ERR;" NA RADCE ";ERL
5030      CLOSE :END
5040 ENDPROC
10000 END

```

Program KOMPLI je zajímavý také tím, že představuje obecně nejobvyklejší schéma zpracování dvou sekvenčních souborů, tvořící základ postupů hromadného zpracování dat.

Poznámka k výpisům programů: V některých případech se používají zvláštní znaky, které se docilují kombinací kláves. Jsou značeny ve špičatých závorkách '<>'. Lomítko znamená postupně, pomlčka současné stisknutí kláves.



Poznámka: V čísle 6/87 si opravte chybu v programu AMATOMAC v ř. 660. Namísto "á =" zde má být "x=".

Náhrada příkazu CSAVE „název programu“

ing. Juraj Oňa, Bratislava

Jedním z nedostatků ATARI BASICu je absence příkazu CSAVE „název souboru“, který umožňuje vyhledání žádaného programu na kazetě bez nutnosti přetočení kazety na začátek, vynulování počítadla a přetočení pásky na počátek požadovaného programu, případně (při menší evidenční pečlivosti uživatele) bez zbytečného nahrání celého programu, který ale právě nehledáme.

To mě vedlo k vytvoření kratičkého programu v BASICu, který uvedený nedostatek částečně odstraňuje.

```
1 OPEN #1,8,128,"C":DIM A$(99):READ A$:? #1:A$:CLOSE #1:DATA CLOSE #7:? "  Progra  
m:.....":CLOAD  
2 ? "POKE 842,12:POKE 784,1:CSAVE:END 2 1  " :POKE 842,13:END
```

Program se nahraje k hlavnímu programu, který chceme uložit na kazetu a odstartuje se příkazem RUN.

Program pracuje následovně:

1. Uloží na kazetu název programu, který vložíme (napíšeme) do mezery v 1. řádku.
2. Uloží hlavní program pomocí příkazu CSAVE, ale před tím se sám vymaže, takže na kazetu se uloží pouze hlavní program.

Pro zavedení takto uloženého programu použijeme příkaz ENTER "C:". Po nahrání 1. datového bloku se na displeji ukáže název programu a ozve se kontrolní tón operace CLOAD. Jestliže chceme program nahrát, stlačíme RETURN, jestliže ne, stlačíme BREAK a pokračujeme ve vyhledávání programu příkazem ENTER "C:".

Praktické poznámky:

- Při psaní hlavního programu musíme rezervovat řádky 1 a 2 pro nahrávací program.
- Tímto způsobem můžeme na kazetu uložit různé příkazy v přímém režimu a zobrazovat libovolně dlouhý název programu (který ale nesmí obsahovat čárky, protože by se proměnná A\$ rozpadla).
- V příkazu DATA v 1. řádku a v příkazu PRINT ve 2. řádku můžeme použít povolené zkratky, čím se program ještě zkrátí a poskytne více prostoru pro název.

Poznámka redakce:

Tento zajisté zajímavý příspěvek byl podroben obvyklé recenzi. První z recenzentů upozornil na to, že ačkoliv teoreticky by měl program fungovat, z více pokusů se mu to povedlo jen jednou.

Druhý recenzent (ing. Jan Jícha, CSc.) si s miniprográmkem pohrál a napsal nám: „2. řádek

který obsahuje čtení z obrazovky, pracoval nespolehlivě v závislosti na momentální poloze kurzoru na obrazovce. Po jeho úpravě na tvar

```
2 ?"ť " :POSITION 2,5:71POKE 842,12:POKE 764,1:CSAVE:END↑ 2↑ 1↑↑ " :  
POKE 842,13:END
```

tj. přidání příkazu pro mazání obrazovky na začátku 2. řádku a příkazu POSITION 2,5 pracuje program bez problému. Za prvním příkazem END ve 2. řádku mohou být použity i šipky nahoru. Autorovo řešení ale vypadá na obrazovce elegantněji (znak negativní šipka dolů je ESC SHIFT INSERT a nejen posouvá tisk o řádek vzhůru, ale zároveň na začátek řádku).“ V závěru svých doporučení dodal: „Ve věku TURBA to asi praktický význam nemá, ale jako TIP a TRIK je to dobré a využitelné i jinak. Proto doporučuji uveřejnit.“ S tím se redakce ztotožňuje a současně děkuje jak autorovi, tak oběma recenzentům.

Klávesnice počítačů

Marek Houša, Praha

ATARI XL/XE

V teorii:

Klávesnice počítačů ATARI XL/XE je využitelná ve třech módech: módu malých písmen, velkých písmen (+ SHIFT) a grafiky (+CONTROL).

Každý z těchto módů je v tabulce kódů klávesnice v paměti ROM určen 64 byty. Z toho plyne, že celá tabulka klaviatury obsahuje 192 bytů.

Činnost systému při užití klávesnice:

Je-li stisknuta jakákoliv klávesnice nebo kombinace kláves, запиše POKEY daný kód na adresu 53769. Odtud kód putuje přes operační systém do registru CH (764 = \$2FC). Tento kód operační systém převede na ATASCII hodnotu, kterou запиše do registru ATACHR (763 = \$2FB).

Abychom mohli přetvářet tabulku kódů, musíme porozumět funkci registru CH. Významy jednotlivých bitů v tomto registru jsou:

Bit	7	6	5	4	3	2	1	0
	Control		Kód klávesy					
	Shift							

Bity 0—5: kód klávesy

Bit 6: SHIFT (— stisknut, — nestisknut)

Bit 7: CONTROL (— stisknut, — nestisknut)

Je-li v tomto registru např. hodnota 29 (binárně 0001 1101), znamená to, že byla stisknuta klávesa 5 a nebylo přitom použito žádného přepínače. Následující přehled kódů klávesnice pro mód malých písmen a číslic nám objasní vznik hodnoty 29 (24 + 5).



	0	1	2	3	4	5	6	7
00	L	J	;			K	+	*
08	O		P	U	RET	I	-	=
16	V		C			B	X	Z
24	4		3	6	ESC	5	2	1
32	,	SP	.	N		M	/	INV
40	R		E	Y	TAB	T	W	Q
48	9		O	7	BACK	8	<	>
56	F	H	D		CAPS	G	S	A

SP = mezera /space/

```
10 REM * ADRESA TABULKY V ROM *
20 A=PEEK(121)+PEEK(122)*256
30 PRINT "ZADEJ NOVOU ADRESU TABULKY V RAM"
40 INPUT NA
50 REM LO/HI BYTE OBOU TABULEK
60 LOA=PEEK(121):HIA=PEEK(122)
70 HINA=INT(NA/256):LONA=NA-HINA*256
80 REM PREKOPIROVANI TABULKY DO RAM
90 FOR I=0 TO 191
100 POKE NA+I,PEEK(A+I)
110 NEXT I
120 REM ZMENY V NOVE TABULCE
130 POKE NA+23,121:REM z-->y
140 POKE NA+23+64,89:REM Z--Y
150 POKE NA+43,122:REM y-->z
160 POKE NA+43+64,90:REM Y--Z
170 REM NASTAVENI UKAZATELE NA POUZIVANOU TABULKU
180 POKE 121,LONA:POKE 122,HINA
```

V tabulce nejsou obsaženy kódy kláves SHIFT, CONTROL, BREAK, RESET, OPTION, SELECT, START, HELP. Významy těchto kláves nelze měnit, protože jsou zabezpečeny operačním systémem.

V praxi:

Pro inspiraci uvádím příklad jednoduchého využití tabulky pro předefinování významu jednotlivých kláves tak, abychom mohli používat jako na většině psacích strojů evropské výroby, tradiční standard QWERTZ.

(Registr KEYDEF (121, 122 = \$79, \$7A) plní funkci ukazatele na právě používanou tabulku klávesnicových kódů; za novou adresu tabulky lze zadat hodnotu např. 1536).

Jestliže jste pochopili popsané skutečnosti, nebude pro vás problémem používat výhod předefinování významů kláves ve vlastních programech, např. textových editorech.



SPEEDCALC — POPIS PROGRAMU A NÁVOD K JEHO POUŽÍVÁNÍ

Ing. Petr Jandík

Úvod

Při koupi počítače do domácnosti nastane pravděpodobně dříve či později chvíle, kdy padne varianta nerudovské otázky: co s ním? Nyní nastává nutnost vymyslet nějaké vhodné použití, bez něhož by se naše domácnost prostě neobešla. Takovým použitím by mohl být třeba rodinný rozpočet, který dozajista potřebuje nový, racionální přístup, a bez počítače jej prostě nelze zvládnout. Jak to

ale navléknout, aby veškeré potřebné práce zvládl laik? Naučit manželku programovat by byl těžký oříšek! A psát programy na porovnání má dáti — dal, dá také dost práce.

Řešení problému je celkem jednoduché. Na pomoc přicházejí programy, označované v anglické terminologii nepřeložitelným slovem "spreadsheet". Znamená to něco jako velký rozprostřený arch, na němž si můžeme malovat sloupce a řádky čísel a výpočtů i s příslušnými popisy a nadpisy. Ing. Biňovec navrhuje ve svém článku ABACUS ve druhém čísle (1987) časopisu ELEKTRONIKA název programování v rastru, mně se zdají lepší termíny tabulkový program a tabulkové výpočty. Na to, aby člověk rozuměl třeba následující tabulce, se jistě nemusí učit programovat a nemusí ani umět ovládat počítač:

AA	AB	AC	AD	AE	AF
1 měsíc:	leden				
2					
3 vydání:	2474.90				
4					
5 položka	1. týden	2. týden	3. týden	4. týden	celkem
6					
7 potraviny	300.00	350.00	290.00	310.00	1250.00
8 oblečení	0.00	995.90	10.00	0.00	1005.90
9 jízdné	30.00	33.00	29.00	35.00	127.00
10 zábava	50.00	20.00	0.00	22.00	92.00
11					
12 součet	380.00	1398.90	329.00	367.00	2474.90

K vytvoření takovéto tabulky programem typu "spreadsheet" také není třeba umět programovat. Stačí, když umíme zapnout počítač a spustit příslušný program. Při práci s ním nám už postačí běžná matematika.

Základním principem programu pro tabulkové výpočty je představa pracovního archu, tvořeného z jednotlivých buněk. Buňky tvoří síť sloupců a řádek, takže každou z nich lze jednoznačně určit označením sloupce a řádku, ve kterém se nalézá. To je důležité pro další práci s tabulkou, při vytváření vzorců a odkazů. O tom ale později.

Nejprve si musíme říci, co vlastně můžeme do jednotlivých buněk psát. Prohlédněme si výše uvedenou tabulku. V prvním okénku zleva, které můžeme označit jako AA1, je nápis "měsíc". Jedná se tedy o vysvětlující komentář k buňce AB1, ve které je napsán také text, tedy "leden". V celém sloupci AA jsou vysvětlující nadpisy k údajům v jednotlivých řádcích. Co ale řádek 6? Ten sice nic nevysvětluje, ale zlepšuje celkový vzhled tabulky, stejně jako řádek 11. Všechny buňky této řádky jsou naplněny znaménky minus, která se v tomto případě také považují za text. Blok buněk, jehož levý horní roh tvoří buňka AB7 a pravý dolní buňka AE10, obsahuje čísla. Tato čísla vznikla mimo počítač a do výpočtu vstu-

pují. Proto je můžeme nazvat vstupními hodnotami. Z těchto vstupních hodnot jsou vypočítány všechny dílčí i celkové součty v tabulce.

Jak ale program sdělit, jak má výstupní hodnoty ze vstupních vypočítat? K tomu slouží poslední typ dat, která se mohou do buněk ukládat, a sice výpočtové vzorce. Z pohledu na tabulku je zřejmé, že například hodnoty v řádce 12 vznikly součtem vstupních hodnot v příslušných sloupcích. Je samozřejmé, že s obyčejným sčítáním bychom daleko nedošli. Proto máme u tabulkových programů řadu dalších možností. Kromě základních aritmetických operací, jako je sčítání, odčítání, násobení, dělení a umocňování, máme dále k dispozici funkce pro součty, nebo průměry bloků buněk, odmocninu, přirozený i dekadický logaritmus i exponenciálu a další funkce. S takovýmto aparátem už lze dělat dost věcí.

☆☆☆

Na výše uvedeném principu je založena řada programů různých firem, realizovaných pro všechny rozšířenější typy počítačů od domácích, až po velké výpočetní systémy. Některé z těchto programů získaly světový věhlas, jako například známý VisiCalc firmy Visi Corporation. Další variace na stejné téma jsou pro počítače Atari osmibitové řady — SynCalc, Calc-Magic a veřejně dostupný SpeedCalc,

který byl otištěn v časopisu COMPUTE! a je u nás nejrozšířenější.

Nejdokonalejší z těchto programů je bezesporu CalcMagic, který má možnost používat ve vzorcích i výrazy, podmíněné výsledkem logických operací a mnoho dalších možností.

SpeedCalc je naproti tomu nejjednodušší, ale tím pádem i nejjednodušeji ovladatelný a proto také tak populární.

V dalším textu se budeme zabývat popisem programu SpeedCalc a jeho používání budeme demonstrovat na příkladech. Nejprve popíšeme význam jednotlivých polí obrazovky, která se po nastartování SpeedCalcu objeví.

1. Obrazovka SpeedCalcu

Obrazovku SpeedCalcu je možné rozdělit na tři části.

Nejvýše je umístěno **pole povelové** — informační, tvořené jedním řádkem. V něm se uživatel dorozumívá s programem, zadává parametry pro různé povelů a může si zde přečíst informace o jejich průběhu. Významy jednotlivých zpráv budou popsány u příslušných povelů.

Pod povelovým — informačním polem se nachází **pole pracovní**. Zde se zobrazuje obsah buňky, na které je právě nastaven kurzor. Pracovní pole má rozsah 3 řádky po čtyřiceti znacích. Tím je určen maximální rozsah zadávaných vzorců, čísel nebo textů. Největší rozsah má **hlavní část**, zabírající celý zbytek obrazovky. V ní je zobrazen výsek tabulky, se kterým se na poslední pracovalo, nebo výsek, který byl příslušným povelům vyvolán pro další práci.

2. Pohyb kurzoru a okénka

Pracovní tabulka SpeedCalcu má 50 sloupců krát 100 řádek. To je samozřejmě mnohem více, než lze ukázat na obrazovce. Proto se musíme naučit **pohybovat se tabulkou a nastavovat okénko na buňky, které právě potřebujeme k práci.**

Nastavování okénka je možné dvojím způsobem:

Nejprve si vysvětlíme **nastavování okénka pomocí pohybu kurzoru**. Kurzorem pohybujeme běžným způsobem pomocí současného stisknutí klávesy CTRL a některé ze čteřice kurzorových kláves +, -, *, =. Klávesa, stisknutá současně s CTRL určuje směr pohybu kurzoru. Kurzor se pohybuje v okénku zvoleným směrem, dokud nenarazí na jeho okraj. Potom se začne stejným směrem pohybovat celé okénko, dokud nenarazí na konec tabulky.

Pokud bychom potřebovali přemístit kurzor přes celou tabulku, byl by zmíněný postup zdlouhavý. Proto máme k dispozici povel, kterým můžeme **levý horní roh okénka nastavit na zvolenou buňku**. Je to povel **"goto"**, který spustíme kombinací kláves (CTRL) a (G). V povelovém informačním poli žádá program zadání adresy buňky, určené sloupcem a řádkem. Chceme-li tedy, aby nám okénko v levém horním rohu ukázalo například buňku AD4, odpovíme touto adresou na dotaz "GOTO;"

K rychlému **přesunu kurzoru do levého horního rohu** okénka slouží povel (CTRL) — (H) (pomlčka mezi klávesami znamená jejich současné stisknutí).

Pokud se kurzor v levém horním rohu okénka již nachází, přesune se celé okénko do základní polohy v levém horním rohu tabulky.

3. Zadávání dat

Data lze zadávat do určité buňky tím, že na ni **nastavíme kurzor** a napíšeme to, co do ní chceme vložit. Obsah buňky se nejprve objevuje v pracovním poli, kde jej můžeme podle potřeby **korigovat** klávesou (DELETE/BACK SPACE).

Zadávat dat ukončíme **ukončovacím znakem**, kterým je buď klávesa RETURN nebo některá z kombinací pro pohyb kurzoru. Po jejím stisku se obsah pracovního pole přesune do příslušné buňky, a pokud se jedná o vzorec, provede se jeho výpočet a v hlavním poli se v příslušné buňce objeví výsledek.

O jaký **typ dat** se při zadávání jedná, si program sám určí podle jejich obsahu. Pokud jsou v pracovním poli pouze numerické znaky, to jest číslice, znaménka, tečka, nebo písmeno **E**, jsou data pokládána za **číslo**. Pokud je prvním znakem vkládaných dat rovnítko, budou data pokládána za **vzorec**. Všechny ostatní případy se uloží jako **text**. To, že program sám rozhoduje o typu dat a uživatel jej nemůže ovlivnit, přináší občas potíže při některých hlavičkách, začínajících třeba tečkou. Pak si musíme pomoci přidáním numerických znaků.

Typ dat, určený programem, se po stisknutí ukončovacího znaku objeví na pravém konci informačního a povelového pole. Pokud se kurzorem posuneme na buňku, která již nějaká data obsahuje, objeví se typ dat v povelovém a informačním poli a obsah buňky v pracovním poli.

Pokud napíšeme nová data, jsou jimi původní data nahrazena.

a) Zadávání čísel

Program určí **data** jako **numerická**, pokud jsou tvořena číslicemi 0 až 9, znaménky + a —, desetinnou tečkou, nebo písmenem **E** při semilogaritmickém tvaru.

Pokud obsahují **jakýkoliv jiný znak**, považuje je program za **text**, jak již bylo vysvětleno. Z toho také vyplývá, že do buňky nelze bez určitého triku vložit text, který by se skládal jen z číslic.

Nyní se však zabýváme numerickými daty. Příklady platných čísel jsou například 123, .001, —65535, nebo 1.2E—4. Napíšeme-li 65,535, je to chápáno jako text kvůli čárce uprostřed čísla. Maximální povolený rozsah čísel je stejný, jako u **Atari Basicu** a je vlastně dán způsobem práce a formátem čísel s pohyblivou řádovou čárkou, který používá vestavěná matematická ROM. Hodnota čísla se tedy může pohybovat od $-1.7E+97$ do $1.7E+98$. Pokud tento **rozsah překročíme**, objeví se v příslušné buňce **zpřava ERROR**.

Pokud nepoužíváme semilogaritmický tvar čísla, nemůžeme tuto chybu způsobit, protože nás program nenechá zadat

více, než 36 číslic. Ačkoliv je možné zadávat tolik číslic, výpočty jsou prováděny jen na 9 platných míst, takže asi nemá valný význam zadávat vstupní hodnoty na více, než 10 platných cifer.

Při použití semilogaritmického tvaru je třeba postupovat opatrně, protože se program při **překročení maximálního rozsahu** může zhroutit.

Numerická data jsou v povelovém — informačním poli indikována písmenem **N**. Pokud je zadané číslo delší, než nastavená šířka buňky, jsou po stisknutí ukončovacího znaku v místě buňky vidět samé hvězdičky. To neznamena ztrátu dat, protože v pracovním poli je vidět vždy celý obsah buňky, bez ohledu na její momentálně nastavenou šířku. Pokud sloupec dostatečně rozšíříme, místo hvězdiček se objeví správná data.

b) Zadávání textů

Textová data slouží k popisu významu jednotlivých buněk a ke grafické úpravě tabulky. **Text může obsahovat libovolné znaky** s tím, že pokud by obsahoval pouze znaky numerické, pokládal by jej program za číslo. V takovém případě bychom museli do buňky napsat za potřebný text dostatečný počet mezer, a na místo, které již nebude v okénku vidět, napsat nenumernický znak. Je-li totiž text v buňce delší, než je její šířka, je z něj v buňce vidět jen odpovídající část. Celý je samozřejmě vidět v pracovním poli.

Po napsání textu do pracovního pole a stisknutí ukončovacího znaku, se **text** uloží do buňky a na pravém konci povelového — informačního pole se objeví písmeno **T**, které signalizuje, že data byla programem přijata jako text.

c) Zadávání vzorců

Vzorce jsou tvořeny matematickými výrazy, ve kterých se mohou uvádět **čísla, proměnné, operátory matematických operací a funkce**.

Aby byla data programem rozpoznána jako vzorec, musí začínat vždy znakem

"=". Není-li na počátku rovnítko, budou data chápána jako text, nebo bude ohlášena chyba.

Čísla se ve vzorcích uvádějí podle stejných zásad, jako při zadávání číselných dat.

Jako **operátory** lze použít tyto symboly:

operátor	význam
+	sčítání
-	odčítání
*	násobení
/	dělení
^	umocňování
=	rovnost

Při vyčíslování vzorců je **důležitá priorita operací**. U všech programů typu "spreadsheet", tedy i u Speed Calcu, je **shodná s pravidly obecné matematiky**. Nejvyšší prioritu mají výrazy v závorkách. Je-li do sebe vnořeno více závorek, pak priorita klesá od vnitřních k vnějším.

Jako činitele operací lze použít nejen čísla, ale i označení buněk, přičemž nezáleží, zda obsahují číslo, nebo jiný vzorec. Při výpočtu se do vzorce dosadí momentální hodnota buňky.

Posledním prvkem, který můžeme ve vzorcích použít, jsou **funkce**. Všechny funkce kromě PI **musí povinně začínat "zavináčem"**. Následující tabulka ukazuje funkce, které můžeme ve SpeedCalcu použít:

Vzorce mají v programu zvláštní postavení. Při zobrazení v pracovním poli je vypsán vzorec, kdežto po zadání ukončovacího znaku (RETURN , pohyb kurzoru) se v buňce k němu příslušející objeví jeho vyčíslená hodnota. Pokud nelze hodnotu vyčíslit, třeba pokud buňka, na kterou se vzorec odkazuje, obsahuje textová data, nebo je prázdná, objeví se v buňce místo hodnoty "ERROR".

Syntakticky **správný vzorec je v povoleném - informačním poli indikován písmenem F**. Je-li argument funkce jednoduchý, může jím být buď číslo, nebo označení buňky. Je-li argumentem označen blok buněk, má tento tvar:

(buňka levého horního rohu)

(buňka pravého dolního rohu)

Jako příklady některých platných vzorců můžeme uvést například :

"=5*(8+3.2E-3)/AB7", nebo

"=  SUM(AB7:AE10)", kterým by se sečetly všechny výdaje z ukázkové tabulky. Má-li být třeba sečten řádek 8, napíšeme funkci třeba takto:

=  SUM(AB8:AE8),

zatímco pro součet sloupce bychom napsali:

=  SUM(AB7:AB10).

funkce	význam
 ABS (x)	absolutní hodnota
 AVE (x)	průměr bloku buněk
 EXP (x)	mocnina základu e (přirozená)
 INT (x)	celá část čísla (odříznutí desetinné části)
 LOG (x)	přirozený logaritmus (základ e)
 RND (x)	zaokrouhlení na nejbližší celé číslo
 SGN (x)	znaménko
 SQR (x)	odmocnina
 SUM (x)	součet bloku buněk
PI	hodnota Ludolfova čísla (3.14159265)

4. Editování tabulky

I když hlavním smyslem činnosti tabulového programu jsou výpočty, neobejdeme se bez dalších podpůrných funkcí. Jednou z nich je editování tabulky, které nám **umožňuje opravovat a měnit obsah jednotlivých buněk**.

Chceme-li změnit obsah některé buňky, nastavíme na ni kurzor. Její dosavadní obsah se objeví v pracovním poli a typ dat v povelovém a informačním poli. **Nový obsah do buňky vložíme stejně, jako by byla buňka prázdná**. Napíšeme do pracovního pole její obsah, který se stiskem ukončovacího znaku vloží do buňky a zobrazí podle typu dat.

Vyprázdnění buňky bez vkládání nového obsahu provedeme povel (CTRL)-(K). „K“ je počátečním slovem povelu Kill — vymazat, zničit, zabít.

Obsah celé tabulky lze vymazat povel (SHIFT) — (CLEAR). Protože se jedná o závažný destruktivní povel, ujišťuje se program dotazem „Are you sure?“, zda nešlo o náhodné stisknutí kláves. Po odpovědi „Y“ se celá tabulka vymaže. Odpovíme-li cokoliv jiného, povel se neprovede.

Potřebujeme-li **malou změnu obsahu buňky**, nemusíme psát celý nový obsah znovu, pokud použijeme **povel EDIT**, tedy **(CTRL)-(E)**.

Při tomto povelu se kurzor pohybuje v pracovním poli, přičemž je povolen jen pohyb doleva a doprava. Počítač pracuje při editaci ve vkládacím režimu. To znamená, že stiskneme-li klávesu, posune se text vpravo od kurzoru o jedno místo doprava a nový znak je vložen na místo kurzoru.

Pokud chceme nějaké **znaky vymazávat**, slouží k tomu klávesa **(DELETE BACK SPACE)** k mazání vlevo, a **(CTRL)-(DELETE BACK SPACE)** vpravo od kurzoru.

Ukončení editace buňky se provede **výhradně klávesou (RETURN)**, což je odlišné od zadávání nových hodnot.

Při editování buňky se zachovávají

všechny dříve nastavené parametry formátování a počtu desetinných míst.

Po ukončení editace se přeneseme změněný obsah z pracovního pole do buňky v hlavním poli. Pokud změním vzorec, provede se po ukončení editace výpočet podle nové formule a do buňky se uloží nová hodnota. Rozsah a průběh výpočtu po změně buňky závisí na zvoleném modu rekalkulace. Pokud při změně uděláme nějakou chybu, zobrazí se v buňce text „**ERROR**“.

5. Rekalkulace

Rekalkulace je jádrem celého tabulového programu a dělá z něj velmi účinný prostředek. Při rekalkulaci se prochází celá tabulka zleva doprava a shora dolů, přičemž jsou znovu vyčíslovány všechny nalezené vzorce.

Z toho vyplývá, že pokud se některý vzorec odkazuje na hodnoty vzorců v buňkách za ním, nemusí být výsledek správný, protože tyto buňky nejsou v daném okamžiku dosud přepočítány. Proto je někdy nutné provést rekalkulaci ve více průchodech. Nejlépe je ukázat tento jev na příkladu:

Mějme například v buňce AA1 vzorec $=2 \times AA2$, v buňce AA2 vzorec $=AA3/2$ a v AA3 hodnotu 4. Když změním hodnotu v AA3 na 6 a provedeme rekalkulaci, změní se hodnota v AA2 na 3, ale AA1 zůstane nezměněna, tedy 4, protože v době výpočtu není ještě vypočtena nová hodnota v AA2. Hodnota 6 se v AA1 objeví teprve po opakované rekalkulaci.

Rekalkulace může probíhat automaticky při každé změně tabulky. Protože však výpočty zabírají nějaký čas, který s rozsahem tabulky vrůstá, hodí se tento režim jen u **malých tabulek**.

U větších je lépe spouštět rekalkulaci ručně, což se provede povel **(CTRL)-(N)**.

Přepínání rekalkulačního režimu z ručního na automatický a zpět se provádí povel **(CTRL)-(R)**.

Pokud chceme zjistit, **v jakém režimu se**

rekalkulace právě nachází, aniž bychom jej změnili, stiskneme k této kombinaci ještě (**OPTION**)

6. Změny formátu

Standardní formát zobrazení čísel a hodnot vzorců je **se dvěma desetinnými místy**, zarovnaný doprava. Přestože je tento formát neobvyklejší, je při různých aplikacích potřeba jej změnit.

Při **změnách formátu** máme možnost měnit umístění dat v buňce povelom (**CTRL**)—(**F**), **počet desetinných míst** povelom (**CTRL**)—(**.**) a **šířku sloupce** povelom (**CTRL**)—(**W**). Přidáme-li k těmto kombinacím ještě (**OPTION**), platí povel pro **celou tabulku**. Nyní podrobněji k jednotlivým povelům:

a) Při **změně formátu** povelom (**CTRL**)—(**F**) se program zeptá: "FORMAT: LEFT, CENTER, OR RIGHT JUSTIFY?". stisknutím (**L**), (**C**), nebo (**R**) zvolíme umístění dat v buňce **vlevo**, **uprostřed** nebo **vpravo**. Povel platí pro buňku, na které je kurzor. Pokud při volbě povelu stiskneme současně ještě (**OPTION**), platí povel pro **celou tabulku**.

b) Počet **desetinných míst** můžeme změnit v rozsahu od **0 do 15** povelom (**CTRL**)—(**.**). Zadáme-li 15, nebudou se hodnoty zaokrouhlovat. Povel platí pro buňku, na které je kurzor, pokud není současně s povelom stisknuta klávesa (**OPTION**). Pak platí pro celou tabulku.

c) Jinak je tomu u povelu (**CTRL**)—(**W**), kterým se určuje **šířka celého sloupce** a při kombinaci s (**OPTION**) všech sloupců tabulky. Šířku lze měnit v rozsahu **4 až 36**. Přitom se obrazovka přeformátuje tak, aby byl na obrazovce ukázán maximální počet celistvých sloupců. Při zmenšování šířky sloupců se místo některých čísel mohou objevit hvězdičky, pokud je šířka sloupce menší než velikost čísla. Zvětšením šířky se čísla opět objeví.

7. Makroeditace

Makroeditaci se rozumí přesunové ope-

race s bloky buněk. Blok buněk je označen, stejně jako u funkcí, **levým horním a pravým dolním rohem**. Makropovely mohou pracovat ve **dvou režimech**, a sice v **relativním a absolutním**. Rozdíl mezi nimi si můžeme ukázat na příkladu:

Předpokládejme, že buňka AA2 obsahuje vzorec =AA1x5, a její obsah chceme přesunout do AB2. Při přesunu v režimu absolutním se obsah přenesse beze změny. V AB2 bude tedy =AA1x5. Při relativním režimu se vzorec upraví tak, aby vzájemná poloha na sebe se odkazujících buněk zůstala zachována. Původně se AA2 odkazovala na AA1, tedy nad sebe. Po relativním přesunu se vzorec upraví tak, že buňka AB2 se opět odkazuje na buňku nad sebou, tedy na AB1, a vzorec se změní na =AB1x5.

K dispozici máme **dva druhy přesunových operací**. Jednak je to **kopie**, kdy zůstane původní blok zachován a jeho obsah se zkopíruje na jiné místo tabulky, jednak **přesun**, kdy je původní blok po přenesení obsahu na jiné místo vymazán.

Absolutní kopie se spustí povelom (**CTRL**)—(**C**), **relativní povelom** (**OPTION**)—(**CTRL**)—(**C**).

Levý horní roh bloku je určen polohou kurzoru v okamžiku spuštění povelu. Povel tedy žádá určení pravého dolního rohu. To provedeme přesunem kurzoru na příslušné místo a stiskneme (**RETURN**). Poté určíme stejným způsobem levý horní roh nové pozice bloku a po stisknutí (**RETURN**)u začne kopírovací proces. Pracovní oblast se přitom používá jako rezervní paměť, proto se v ní ve fázi přesunu dat objevují různé nesmyslné symboly, což ovšem není na závadu. Po zkopírování se provede vždycky rekalkulace, bez ohledu na nastavený rekalkulační režim.

Přesun bloku buněk probíhá stejně jako kopie, ale spouští se povelom (**CTRL**)—(**M**) a (**OPTION**)—(**CTRL**)—(**M**).

8. Obhospodařování paměti

SpeedCalc má pro data uložená v buň-

kách k dispozici asi 20 kB paměti. Z toho vyplývá, že tabulku nelze zaplnit celou. Kromě toho nemůžeme ani vyčerpat celou kapacitu paměti, protože musí něco zbyť ještě na přesunové povely. Pokud nemá přesunový povel dostatek paměti, zhroutí se. **Velikost volné paměti** se zjišťuje povel **(CTRL)–(A)**.

Z výše uvedených důvodů se doporučuje **nepřekračovat cca 1600 obsazených buněk v tabulce**. Tento počet nechává dostatečnou rezervu i pro editaci a přesuny.

9. Práce s periferiemi

SpeedCalc má pro práci s periferiemi **tři povely**. K uložení tabulky slouží povel **(CTRL)–(S)**, který vyžaduje zadání specifikace výstupního zařízení a u disku ještě jména souboru. Specifikaci musíme ukončit klávesou **(RETURN)**. Označení jednotlivých zařízení najdeme v příslušných uživatelských návodech.

Načtení dat do paměti se provede povel **(CTRL)–(L)**. Oba tyto povely vyžadují stejné parametry. Lze je zrušit zadáním prázdné specifikace, tedy samotnou klávesou **(RETURN)**.

Pokud proběhne zvolená operace v pořádku, je v povelovém — informačním poli zpráva **"NO ERRORS"**. Nastane-li chyba, je to oznámeno zprávou **"I/O ERROR # "** a číslo, jehož význam najdeme v příslušném manuálu.

Obsah disku si můžeme prohlédnout povel **(CTRL)–(D)**. Mezi povely pro disk se dá počítat i **povel k přechodu do DOSu (CTRL)–(X)**. Tento povel znamená většinou zároveň ztrátu dat. Proto je jistěn bezpečnostním dotazem.

10. Tisk

Tisk je sice také práce s periferiemi, ale má zcela jiný význam než sčítání a ukládání. Tisk spustíme povel **(CTRL)–(P)**. Především je třeba zdůraznit, že tisk je pochod pouze výstupní. Zatímco načítání a ukládání pracuje s vnitřní reprezentací dat tabulky, tisk produkuje čitelnou formou ve znacích kódu ATASCII. Z toho

vyplývá, že **vytištěná data už nelze žádným způsobem dostat zpátky**.

Povel **(CTRL)–(P)** vyžaduje, stejně jako načítání a ukládání, **zadání jména zařízení** a u disku i **jméno souboru**.

Při tisku je **důležitá pozice kurzoru** v okamžiku vyvolání povelu. Kurzor určuje pravý dolní roh oblasti tabulky, která se bude tisknout.

Pro **tisk na obrazovku** zadáme **E**; na **tiskárnu P**; a do **diskového souboru D**: **jméno.ext**.

Tisk na disk má význam pro další úpravu tabulky textovým programem, kterým lze tabulku doplnit textem, stránkováním a podobně.

11. Barva a světelnost obrazovky

SpeedCalc umožňuje **měnit barvu** pozadí i textu **ve 128 odstínech**. Povel **(CTRL)–(B)** měníme barvu pozadí o jeden krok dopředu. Pokud k tomu ještě stiskneme **(OPTION)**, mění se barvy v opačném pořadí.

Světelnost textu měníme povel **(CTRL)–(T)** směrem dopředu, s přídavným **(OPTION)** směrem dozadu.

12. Slabá místa SpeedCalcu

Mezi **nedostatky** SpeedCalcu můžeme počítat **tři věci**.

První je nemožnost tisknout libovolné výšky z tabulky, takže velkou tabulku můžeme tisknout na pokračování jen velmi těžko. SpeedCalc také při tisku **neprovádí stránkování**.

Druhým nedostatkem je absence možnosti vstupu a výstupu dat tabulky v takzvané DIF konverzi, což znemožňuje například zobrazovat SpeedCalcem vypočtená data programem pro tvorbu grafů (**B/GRAPH**, **Syngraph** apod.), ani je přenést do databanky.

Třetím nedostatkem je nedokonalá inicializace programu při jeho spouštění. Projevuje se to tím, že po provozu některých programů, návratu do DOSu a spuštění SpeedCalc nefunguje a je třeba vypnutím počítače vymazat paměť. Potom teprve po novém spuštění funguje program tak, jak má.

BASIC ZX Spectrum na ATARI XL/XE

Slavomír Pavlíček, Jiří Richter, Miroslav Oškera

Mnozí z těch, kteří si pozorně přečetli příručku J. Richtera "TURBO 2000 — systém zrychleného přenosu dat z magnetofonů ATARI") zajisté nepřehlédli poznámku, ve které se autor zmiňuje o možnosti kopírovat programy vytvořené pro mikropočítače Sinclair ZX Spectrum na počítače ATARI XL/XE. Jakkoliv se to může zdát nepravděpodobné, je tomu tak. Pro zpracování (přetvoření) programů pro mikropočítače Sinclair ZX Spectrum byl vytvořen dále popsáný program TRAPSAT (Translation of Programmes from Sinclair to ATari). Jedná se o program, který převádí program vytvořený v jazyku BASIC mikropočítače Sinclair ZX Spectrum (dále jen "ZX-BASIC") do programovacího jazyku ATARI BASIC (dále jen "A-BASIC").

Postupným vývojem a odlaďováním programu a získáváním zkušeností s ním bylo zjištěno, že převod do A-BASICu je značně problematický (problém malých písmen). Proto byl k účelu úprav programů ZX-BASIC použit programovací jazyk TURBO BASIC (dále jen "T-BASIC"), který se maximálně osvědčil.

Seznam povelů pro SpeedCalc:

povel	význam
CTRL-A	kontrola volné paměti
CTRL-B	další barva pozadí
CTRL-C	kopie bloku — absolutně
CTRL-D	obsah disku
CTRL-E	editace aktuální buňky
CTRL-G	skok na vybranou buňku
CTRL-H	home-kurzor do levého horního rohu
CTRL-K	vymazání buňky
CTRL-L	zavedení SpeedCalc souboru
CTRL-M	přesun bloku — absolutně
CTRL-N	rekalkulace tabulky
CTRL-P	tisk
CTRL-R	přepnutí rekalkulace zap/vyp
CTRL-S	uložení SpeedCalc souboru
CTRL-T	zvýšení světelnosti textu
CTRL-W	změna šířky sloupce
CTRL-X	návrat do DOS
CTRL-.	změna počtu desetinných míst
SHIFT-CLEAR	vymazání tabulky
OPTION-CTRL-B	předchozí barva pozadí
OPTION-CTRL-C	kopie bloku — relativně
OPTION-CTRL-M	přesun bloku — relativně
OPTION-CTRL-R	status rekalkulace
OPTION-CTRL-T	snížení světelnosti textu
OPTION-CTRL-W	změna šířky všech sloupců

Popis programu TRAPSAT

Vlastní práce programu TRAPSAT spočívá v tom, že převádí text programu v ZX-BASICu na znaky ASCII. Znaky postupně ukládá do bufferu, odkud je rutina CIO převádí na kazetu (systém nahrávání LIST "C").

- *) Richter J. TURBO 2000- systém zrychleného přenosu dat z magnetofonů ATARI. Zpravodaj ATARI KLUBU Praha, příloha II, 487. ZO Svazarmu, Praha 1987.

Poznámka

Na tomto místě bychom chtěli upozornit na dekódovací tabulku. V tabulce jsou seřazena klíčová slova T-BASICu, která odpovídají klíčovým slovům ZXBASICu, uvedeným v komentáři. Slova, ke kterým nemá T-BASIC ekvivalent, jsou ponechána, a jejich význam je uveden dále v textu.

Je nutné připomenout, že některá slova mají více způsobů překladů do T-BASICu. V těchto případech byl vybrán nejužívanější význam. Obdobně v semigrafických či speciálních znacích byly vybrány znaky nejpodobnější.

RND	RND
INKEY	INKEY
PI	konstanta 3.141592
FN	volání jednopříkazové fce
POINT	LOC.
SCREEN	znak pod kurzorem
ATTR	poloha kurzoru na obrazovce
AT	viz text dále
TAB	viz text dále
VAL	VAL
CODE	ASC
VAL	VAL
LEN	LEN
SIN	SIN
COS	COS
TAN	TAN
ASN	ASN
ACS	ACS
ATN	ATN
LN	LN
EXP	EXP
INT	INT
SQR	SQR
SGN	SGN
ABS	ABS

Dekódovací tabulka



PEEK	PEEK
IN	vstup z portu
USR	USR
STRØ	STRØ
CHRØ	CHRØ
NOT	NOT
BIN	BIN
OR	OR
AND	AND
<=	<=
>=	>=
<>	<>
LINE	
THEN	THEN
TO	TO
STEP	STEP
DEF FN	definice jednoprikazove funkce
CAT	prikaz pro praci s mikrodrivy
	(dale md prikaz')
FORMAT	md prikaz
MOVE	MOVE
ERASE	md prokaz
OPEN#	OPEN#
CLOSE#	CL.#
MERGE	md prikaz
VERIFY	verifikace zaznamu na pasku
BEEP	prikaz pro zvuk
CIRCLE	CI.
INK	C.
PAPER	SE.3,9,
FLASH	zapini/vypinani blyskani textu
BRIGHT	zmena jasu textu
INVERSE	vypis inverzniho textu
OVER	zvlastni druh prepisovani
OUT	vystup na port
LPRINT	LPRINT
LLIST	L."P:"
STOP	STOP
READ	READ
DATA	D.
RESTORE	RES.



NEW	NEW
BORDER	SE. 4, 8,
CONTINUE	CONT
DIM	DIM
REM	.
FOR	F.
GO TO	G.
GO SUB	GOS.
INPUT	I.
LOAD	LD.
LIST	L.
LET	
PAUSE	PAUSE
NEXT	N.
POKE	POKE
PRINT	?
PLOT	PL.
RUN	RUN
SAVE	SAVE
RANDOMIZE	nastavení počáteční podmínky pro RND
IF	IF
CLS	CLS
DRAW	viz text dále
CLEAR	CLR
RETURN	RET.
COPY	COPY

Práce s programem TRAPSAT

Především je nutné zdůraznit, že program TRAPSAT rozhodně „neudělá“ z osobního počítače ATARI XL/XE mikropočítač Sinclair ZX Spectrum (jako to činí např. existující program na C 64/. Výsledný produkt programu TRAPSAT nám ponechává stále dost práce. Největší problémy nám budou dělat:

- a) závorky,
- b) formátování textu na obrazovku,
- c) grafika,
- d) dimenze řetězců,
- e) délka programového řádku.

Ad a)

Jednou z velkých starostí jsou závorky. V ZX—BASICu se mohou (ale nemusí) dávat funkční proměnné do závorek.

Ad b)

ZX—BASIC používá klíčové slovo TAB (n), které je možno nahradit řetězcem s n mezerami, a klíčové slovo AT x, y, které je možno nahradit POSITION x, y. Problém je ale v tom, že klíčové slovo AT může být použito za PRINT libovolně-krát, zatímco POSITION se musí psát před každým PRINT.

Při vypisování dlouhých textů na obrazovku nám může pomoci POKE 83,d kde d = PEEK (82)+32. To automaticky umístí rozdělovací znaménko na konec řádku a nakreslí správné obrázky v semigrafice.

Ad c)

Výstup v grafice nám přinese rovněž značné obtíže. ZX-BASIC používá relativní souřadnice u příkazu DRAW TO, které mají počátek vždy v posledním příkazu PLOT nebo MOVE.

Ad d)

ZX-BASIC může řetězce řadit do matic a programátoři dvoudimenzových řetězců hojně využívají.

Ad e)

Záludným problémem je prakticky neomezená délka programového řádku v ZX-BASICu. Nahrávání ENTER "C:" pracuje přes vstupní buffer. Ten má však omezenou délku. Pokud je nahrávaný řádek delší než je délka bufferu, vypíše T-BASIC ERROR 137 a řádek není vůbec uložen do paměti. Z tohoto důvodu byl program TRAPSAT vybaven možností výpisu na obrazovku. V současné verzi lze tedy program vytvořený v ZX-BASICu vypisovat na obrazovku, výpis zastavovat stlačením CONTROL - 1 (jako u A—BASICu) a libovolně si jej prohlížet.

Pokud ve výpisu uvidíme logický řádek delší než tři obrazové řádky, je lepší si řádek opsat a v T—BASICu jej opět, pokud je to nutné, dopsat. Naštěstí většina programů využívá "super-dlouhé" řádky zřídka.

Poznámka:

ZX-BASIC umožňuje nahrát na kazetu k programu i hodnoty proměnných. Hodnoty jsou uloženy bezprostředně za vlastním programem. Program považuje proměnné za programové řádky a pokračuje v překladu. To na obrazovce způsobí výpis nesmyslného textu, který je nahrán i na kazetu. Při nahrávání do T-BASICu se nám tyto neexistující řádky obvykle ztratí (ERROR 137 - viz. výše), ale ty co zůstanou, musíme vymazat sami.

Závěr

Program, který byl ve stručnosti popsán, umožňuje racionálním způsobem využít poctivou programátorskou práci kolegů - uživatelů mikropočítačů Sinclair ZX Spectrum. V mnoha případech lze tímto způsobem uspořit čas. Je to však relativní, protože mnohdy bude lepší (snadnější, rychlejší, pohodlnější) vytvořit nový program přímo na počítači ATARI, než se předělávat s programem původně vytvořeným pro ZX Spectrum.

Ke zpracování programu TRAPSAT nás však vedlo jednak to, že v některých případech může být účelné přepracování unikátních programů pro ZX Spectrum na ATARI, ale po pravdě řečeno také to, že pro ataristy není problém nahlédnout do kuchyně sinclaristů, jinými slovy, upravit si programy v ZX-BASICu pro vlastní potřebu.



ATARI STATISTICS 1 — návod na použitie programu

ing. Eva Blechová, Bratislava (preklad)

Z vlastnej skúsenosti vieme, že zdĺhavé matematické operácie sú jednou z oblastí, v ktorých je výborným pomocníkom počítač. Pre osem bitové počítače ATARI už v roku 1982 firma ATARI vyvinula súbor systémových programov nazvaných STATISTICS1. Slúžia pre výpočet hlavných štatistických vzťahov medzi údajmi, teda všade tam, kde potrebujeme narábať so štatistickými metódami. Ocenia ho najmä študenti, ale uplatnenie nájde aj v praxi. Samozrejme nie všetky rozhodnutia potrebujú štatistické spracovanie. Niektoré sa dajú urobiť intuitívne ihneď. Niektoré z nich sa ukážu ako chybné až po získaní širších štatistických údajov a toto rozhodnutie sa potom zamietne.

Na úvod vám priblížime trochu teórie zo štatistiky na príklade prediktného systému pre doporučenie žiakov ZŠ k ďalšiemu štúdiu.

Majme dvoch žiakov, ktorí navštevujú rôzne školy a mali skúšku z matematiky. Prvý, nazvime ho Jozef, získal 78 bodov, zatiaľ čo jeho priateľ Boris 86 bodov. Prvý záver je, že Boris je lepší žiak z matematiky. Ale táto úvaha je správna len na prvý pohľad, lebo neberie do úvahy, že Jozefova trieda dosiahla priemer 60 bodov so štandardnou odchýlkou 12, kým Borisova trieda dosiahla priemer 74 bodov so štandardnou odchýlkou 10. Z toho vyplýva, že Jozef dosiahol stupeň ohodnotenia $(78-60)/12 = 1,5$ štandardnej odchýlky prímeru celej triedy. Boris dosiahol $(86-74)/10 = 1,2$ štandardnej odchýlky triedy.

Z toho plynie záver, že Jozefove postavenie v triede je lepšie ako Borisove. Naša intuícia mala ale opačný záver. Rozdiel môže byť v obtiažnosti, alebo obsahu skúšky. Pri doporučení na ďalšie štúdium sa toto samozrejme zohľadňuje v prediktnom systéme, ale výpočet nie je na počítači ATARI, ale na počítačoch JSEP s platnosťou pre všetky školy v republike.



Zavedenie programu

Program je napísaný v jazyku BASIC a zavádza sa dvojicou inštrukcií CLOAD a RUN. Program sa spustí automaticky, objaví sa úvodné menu. Čím väčšiu pamäť má Váš počítač, tým viac údajov môžete naraz spracovávať.

Obrazovka úvodného menu

Voľba sa určí napísaním príslušného písmena A—F a stlačením RETURN. Každá databáza (súbor prvotných údajov) sa bude nazývať "NONAME", pokiaľ bude v pamäti. Ak sa uchová na kazetu, môže byť zadané meno na desať znakov. Na toto meno sa budete v budúcnosti odvolávať pri zavedení databázy z pásky.

STATISTICS I CASSETTE VERSION (DISC VERSTION)
DATABASE — "NONAME"

A: INPUT New Database vložiť nové údaje
 B: EDIT Data Points oprava údajov (editovanie)
 C: SAVE Database On Cassette uchovať údaje na páске
 D: LOAD Database from Cassette zavedenie údajov z pásky
 E: PROCESS data spracovanie údajov
 F: PRINT Data Points vytlačenie
 G: RETURN to BASIC návrat do BASICu
 PLEASE ENTER OPTION DESIRED stlačte požadovanú voľbu písmeno A až G
 (LETTERS A THROUGH G):

Popis volieb

A — INPUT NEW DATABASE (vložiť nové údaje)

Program analyzuje vstup nových údajov. Ak prvýkrát zadávate údaje po zavedení programu, obrazovka sa hneď prispôbi na vstup údajov. Ak už je program v činnosti a staré údaje sú k dispozícii v pamäti, ozvú sa 4 zvukové signály a nasleduje kontrolná otázka, či skutočne budú vstupovať nové údaje:

DESTROY OLD DATA Chcete zničiť staré údaje?
 ARE YOU SURE
 (Y/N)?

Ak na túto otázku odpoviete Y a RETURN, staré údaje sa zmažú a rozsvieti sa obrazovka pre vstup údajov. Ak odpoveď na otázku bude N, staré údaje sa nezmažú a znovu sa vypíše menu.

Obrazovka pre vstup údajov má tvar:
 nnnn AVAILABLE DATA POINTS
 x1: —

INPUT DATA THEN HIT START Zadávajte údaje a potom stlačte START

Poznámka: nnnn je počet prípustných údajov, závislý od pamäte Vášho počítača. Napr. u ATARI 800 XL je to 2047.

Kurzor je umiestnený vpravo hneď pri x1. Napište prvý údaj a stlačte RETURN. Hodnota i v xi vzrastá automaticky. Keď vstúpi 20 údajov, obrazovka začne rolovať smerom hore. Ak ste zadali všetky údaje, stlačte len RETURN. Ak bol zadáný nečíselný údaj, ozve sa zvukový signál a na spodku obrazovky sa rozsvieti oznam INVALID INPUT. V tomto programe možno zadať každé číslo, ktoré je platné pre programovanie v jazyku BASIC. Samozrejme čísla druhej a tretej mocniny musia byť z intervalu $-5E + 24$ do $5E + 24$. Ak zadáte viac údajov ako je kapacita pamäte, ozve sa tiež zvuková indikácia a vypíše sa oznam MEMORY FULL (preplnenie pamäte). Ďalšie údaje je možné zadávať až po zmazaní niektorých existujúcich.

B — EDIT Data Points (oprava údajov)

Táto činnosť dovoľuje pridávanie, výmaz a modifikáciu existujúcich údajov. Existujúce údaje sa zobrazujú v nasledujúcom tvare:

DATA LISTING FOR NONAME
 21 POINTS USED PAGE FORWARD
 x1
 x2
 x3

x9

USE CURSOR CONTROL KEYS TO EDIT DATA Nastavte kurzor editovacími klávesami
na opravovaný údaj

START — FOR MENU START — návrat k menu,

SELECT TO PAGE SELECT — stránkovanie,

OPTION — TO CHANGE PAGE DIRECTION OPTION — zmena postupnosti údajov

Na obrazovke je zobrazených len 9 hodnôt údajov, hore svieti správa PAGE FORWARD. Ak chcete vidieť ďalších 9 údajov, stlačte SELECT. Ak chcete opačný postup, stlačte OPTION. Tento spôsob sa zobrazuje ako PAGE REVERSE. Nemusíte prejsť cez všetky údaje. Stlačením inej klávesy ako OPTION, SELECT a START zmení sa obraz z listingového módu do DATA EDIT módu.

Údaje môžu byť opravované jedným z nasledujúcich spôsobov:

1. Nastavte kurzor na pozíciu údaj, ktorý chcete meniť, napíšte novú hodnotu a stlačte RETURN.
2. Napíšte príslušné xi, ktoré chcete meniť, jednu medzeru a novú hodnotu, stlačte RETURN.

Nové údaje môžu byť vkladané druhým spôsobom. Napíšte x a nasledujúce j po poslednom existujúcom xi, stlačte medzeru, napíšte hodnotu a stlačte RETURN. Ak medzi xi a hodnotou nie je medzera alebo bola zadaná nenumerná hodnota, zobrazí sa chybová správa a ozve sa zvuková indikácia.

nnnn AVAILABLE DATA POINTS
INVALID INPUT

INPUT DATA THEN HIT START

V tomto prípade údaj nie je zmenený a operácia sa musí zopakovať. Údaj sa môže zmazať napísaním príslušného xi a stlačením RETURN.

Ak chcete údaje v pôvodnom poradí, musíte ich modifikovať jeden po druhom, alebo sa musíte vrátiť do menu, vybrať si input mód, zničiť editované údaje a znovu založiť databázu.

C — SAVE Database On Cassette (uchovať údaje na pásku)

Doporučujeme, aby uchovanie Vašej databázy bolo na inej páske ako vlastný program. Tým predídete neúmyselnému zničeniu programu.

Na obrazovke sa objaví text:

ENTER NAME OF DATABASE TO BE SAVED Napíšte meno databázy, ktorú chcete
uchovať

HIT RETURN FOR DEFAULT Stlačte RETURN, ak chcete preddefinovať názov NONAME
NAME OF: "NONAME"

ENTER EXIT FOR RETURN Zadajte EXIT, ak chcete vrátiť činnosť do menu
TO MAIN MENU

ENTER NAME: — Napíšte meno: —

Napíšte nové meno maximálne na 10 znakov, stlačte RETURN a toto meno sa uloží pred Vaše údaje. Ak bol zadáný dlhší názov ako 10 znakov, zoberie sa len prvých 10 znakov a zvyšok sa ignoruje.

Potom sa zobrazí obrazovka:

POSITION TAPE Zložte pásku
— PRESS RECORD — stlačte RECORD
— HIT RETURN — stlačte RETURN

Pri nahrávaní treba na magnetofón stlačiť RECORD i PLAY súčasne. Ak stlačíte obe tlačítka, ozvu sa dva zvuky. Po stlačení RETURN sa začne uchovávať databáza. Ak sa databáza nahrá na pásku, objaví sa obrazovka:

DATABASE Databáza NONAME je uchovaná
NONAME
SAVED

Tento oznam svieti len krátky čas a potom sa znovu zobrazí menu.

D — LOAD Database from Cassette (zavedenie údajov z pásky)

Touto voľbou sa zavedú údaje uschované na páске do pamäti počítača. Ak sú už údaje v pamäti a zvolí sa táto činnosť, vypíše sa na obrazovke otázka:

DESTROY OLD DATA Chcete zničiť staré údaje?

ARE YOU SURE (A/N)?

(Y/N)?

Ak stlačíte N, dostanete sa znovu do menu. Odpoveď Y spôsobí zvukovú indikáciu a vypísanie nasledujúcej správy:

POSITION TAPE Zložte pásku
— PRESS PLAY — stlačte PLAY
— HIT RETURN — stlačte RETURN

Páska v magnetofóne musí byť nastavená na začiatok databázy, ktorú chcete nahrávať. Ak nevíete pozíciu databázy, nastavte pásku na začiatok a naťahujete jednu databázu za druhou, až kým nenatiahnete hľadajú. Naťahovanie sa začína stlačením RETURN. Ak sa databáza úspešne natiahla, objaví sa obrazovka:

DATABASE
NONAME Databázu NONAME natiahnutá
LOADED

Ak sa databáza nenatiahla správne, objaví sa správa:

CASSETTE RECORDER Chybná páska
ERROR

V oboch prípadoch sa znovu objaví menu. Ak sa databáza nenatiahla správne, doporučujeme pretočiť pásku a natiahnuť ju znovu.

E - PROCESS Data

(zpracovanie údajov)

Pri zadání tejto voľby sa s údajmi v pamäti vykonávajú jednotlivé operácie. O tom, že sa vykonáva operácia s údajmi, sa dozviete z nasledujúcej obrazovky:

DATA BEING Začala sa práca

PROCESSED s údajmi

PHASE 1 fáza 1

Vykonávanie operácií sa skladá zo 6 fázov, ktoré sa postupne zobrazujú. Niektoré operácie sa vykonávajú dlhšie ako iné. Najdlhšie sa vykonávajú fázy 4 a 5. Dĺžku ovplyvňujú i hodnoty údajov a ich zotriedenosť. Program hľadá najoptimálnejšie vykonávanie operácie. Fázy sú nasledujúce:

1. výpočet priemeru,
2. výpočet druhej odmocniny súčtu všetkých x_i umocnených na druhú, výpočet štandardnej odchylky, smerodajnej odchylky a vypuklosti,
3. príprava pre triedenie,
4. triedenie údajov,
5. zobrazenie najčastejšej hodnoty (mode),
6. nájdenie strednej hodnoty a asymetrie.

Ak máte veľký počet údajov, treba na spracovanie dosť dlhý čas. Po 9 minútach po vstupe z klávesnice, začne obrazovka meniť farbu. Tento proces sa zastaví stlačením ľubovoľnej klávesy na klávesnici. Ak sa skončí 6. fáza, objaví sa napríklad výsledok:

DATABASE NONAME
GENERAL STATISTICS
23 DATA POINTS

MEAN = 8.69565217
MEDIAN = 7
R.M.S. = 10.6648549

priemer
stred

STD DEV = 6.31329684
VARIANCE = 39.85770688
KURTOSIS = 0.55693344
SKEWNESS = 1.14595033

štandardná odchylka
smerodajná odchylka
vypuklosť
asymetria

MODES:

3 7

najčastejšie číslo

PRESS OPTION TO PRINT STATISTICS
PRESS START TO CONTINUE

Stlačte OPTION, ak chcete opis na tlačiarňu,
stlačte START, ak chcete pokračovať

Ak neboli v pamäti žiadne údaje alebo bol jeden údaj alebo boli príliš dlhé hodnoty údajov, na spodku obrazovky sa objaví červené pole s chybovou správou, napríklad:

DATA BEING PROCESSED PHASE 1

DIVISION BY ZERO OR VALUE OVERFLOW

Delenie nulou alebo pretečenie hodnoty

F — PRINT Data Points (vytlačenie)

V tejto činnosti sa údaje z pamäti vytlačia na tlačiareň (postačí aj najjednoduchšia ATARI 1020 na úzky papier).

Ak stlačíte START, tlač sa zastaví. Ak stlačíte OPTION a tlač nie je možná, ozve sa zvuková indikácia, ktorá ohlásí chybu. Ak tlačiareň nie je pripojená alebo tlač nie je možná, rozsvieti sa obrazovka:

PRINTER DOES NOT RESPOND

Po niekoľkých sekundách sa objaví menu.

G — RETURN to BASIC (návrat do BASICu)

Po stlačení tejto voľby sa program STATISTICS 1 hneď ukončí a riadenie počítača sa vráti programovaciemu jazyku BASIC. Všetky údaje, ktoré sú v tomto momente v pamäti počítača sa stratia, ak neboli pred tým uchované na páske alebo diskete.

V tomto programe sa používajú nasledovné štatistické výrazy a vzorce:

- Priemer** $\bar{x} = (x_1 + x_2 + \dots + x_n)/N$
kde $x_1, x_2, \dots$ sú údaje a N je ich počet.
- Median** — stred — po zoradení údajov od najnižšieho po najvyšší je median hodnota prostredného údaje, ak je nepárny počet údajov. Hodnota uprostred stredných hodnôt je median, ak je párny počet údajov.
- Modus** je hodnota alebo hodnoty, ktoré sa vyskytujú viackrát. Môže ich byť viac alebo žiadne.
- R.M.S.** alebo Root Mean Square hodnota je odmocnina zo sumy všetkých x , umocnených na druhú a vydelená počtom x .

$$R.M.S. = \frac{\sum_{i=1}^N x_i^2}{N}$$

- Štandardná odchylka** je druhá odmocnina z rozdielu všetkých údajov a priemeru. Rozdiel je umocnený druhou mocninou a vydelený počtom údajov minus 1.

$$s = \sqrt{\frac{\sum_{i=1}^N x_i^2 - N \bar{x}^2}{N - 1}}$$

$$s^2 = \frac{\sum_{i=1}^N x_i^2 - N \bar{x}^2}{N - 1}$$

- Smerodajná odchylka** je druhou mocninou štandardnej odchylky.
- Vypuklosť (vydutosť)** je suma relatívneho zvýšenia normálnej hodnoty. Vypuklosť je matematicky definovaná ako funkcia štvrtej mocniny rozdielu medzi každým x_i a priemerom delená počtom x_i , ktorý je vynásobený štandardnou odchylkou umocnenou na štvrtú. Finálny výsledok dostaneme, ak tento podiel umocníme na -3 .

$$a_4 = \left[\frac{\sum_{i=1}^N x_i^4 - N \bar{x}^4}{N \cdot s^4} \right]^{-3}$$

- Asymetria** je spoločný deliteľ nesúmernosti z hodnoty ako porovnanie normálnej hodnoty a je vypočítaná ako tretí moment priemeru. Je to suma rozdielu medzi každým x_i a priemerom umocnená na tretiu a delená počtom hodnôt x_i . Výsledok je potom delený štandardnou odchylkou umocnenou na tretiu. Pre normálnu krivku je táto hodnota nula.

$$a_3 = \frac{\sum_{i=1}^N x_i^3 - N \bar{x}^3}{N \cdot s^3}$$

Údaje môžete vytlačiť aj grafickou formou pomocou systémových programov ATARI GRAPH IT.

KOUTEK TECHNIKY



CENTRONICS PRINTER INTERFACE

ing. Václav Fajta, Hradec Králové

(pokračování z čísla 6/1987)

Poznámka redakce:

Úvodem k pokračování článku z minulého čísla Zpravodaje AK [6/1987, s. 46—49] připomínáme, že z technických důvodů jsou v dále uvedených výpisech použity znaky £ a §, přičemž £ = # a § = @.

Ve výpisech programů jsou rukopisné doplňky, které vyplynuly z autorské korektury, resp. ze snahy autora sdělit nejnovější variantu programů, avšak z technických důvodů tisku čísla již nebylo možné pořídit nové výpisy na počítačové tiskárně. Autor nás též bezprostředně před zahájením výroby čísla požádal o doplnění článku následujícím textem:

"Jelikož dostávám hodně dotazů k připojení tiskáren k mikropočítači ATARI, uvádím některé další údaje:

1. Programy CENTRONICS V1.0, V2.0 a V2.1 je možné provozovat na všech typech ATARI XL/XE, tedy i na 800 XL, 800 XE, 65 XE, 130 XE. Důsledně jsem dodržoval zásady programo-

vání, které uvádím v publikaci "Základy programování mikroprocesoru 6502" v 8. kapitole — Zásady programování v JSA. [Tato publikace vyjde pro členy AK v 1. pol. 1988 — pozd. red.] Proto nevidím důvod, proč by programy neměly "chodit" i na předcházejících verzích ATARI 600 a 800.

2. Programy CENTRONICS V1.0, V2.0 a V2.1 mohou řídit všechny tiskárny s rozhraním CENTRONICS, tj. např. ROBOTRON K6313, ROBOTRON K 6314, PRÁŠIDENT 6313, D-100, EPSON, SEIKOSHA, NL-10, SPEEDY, PRT-80 atd., popřípadě tiskárny s paralelním rozhraním blízkým normě CENTRONICS, jako např. CONSUL 2112, CONSUL 2111 atd. U dalších typů paralelního přenosu je nutné upravit podprogram ONECHAR. U některých tiskáren stačí zvolit správný typ zapojení [např. negativní signály pro DZM-180 podle zapojení ing. Komína].

3. Programy CENTRONICS lze implementovat i do textových editorů, např. do programu SPEED SCRIPT apod."



Řada dostupných tiskáren používá paralelní přenos dat podle normy CENTRONICS. Data se přenášejí po sedmi nebo osmi vodičích. Další dva vodiče slouží k řízení přenosu dat. Signál STROBE provádí zápis do tiskárny, signálem BUSY informuje tiskárna, že je zaneprázdněna zpracováváním přijatých dat. Vyskytují se i další pomocné signály, např. zda je založen papír apod.

K počítačům ATARI XL/XE lze připojit tiskárnu přes port PA obvodu 6520, který je vyveden na konektory pro křížové ovladače (joysticky). Port lze naprogramovat jako výstupní. Bity 0 až 6 přenášejí data pro tiskárnu, bit 7 slouží jako signál STROBE. Na vstupu TRIGGER lze připojit BUSY, případně další pomocný signál.

Protože výstupy obvodu 6520 nelze příliš zatěžovat, je třeba připojit tiskárnu přes oddělovací zesilovač. Lze použít obvodu MH

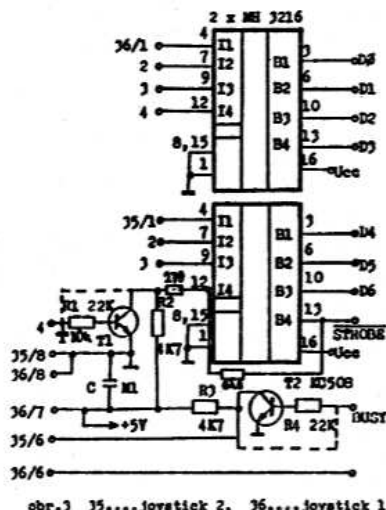
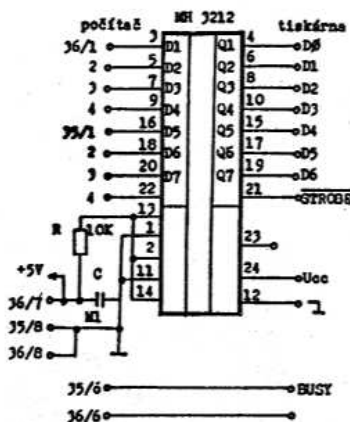
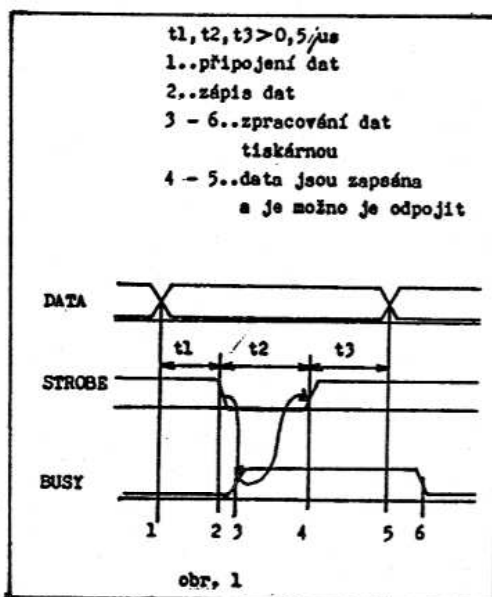
3212 (viz obr. 2), MH 3216 (obr. 3), případně MHB 8282 nebo MHB 8286 — zapojení jako MH 3212, ale s menším počtem řídicích vstupů a v menším pouzdře.

Některé tiskárny používají jinou polaritu signálů, než je na obr. 1. To je možné řešit úpravou programu nebo zařazením invertoru jako na obr. 3. Pokud není třeba signál invertovat, odpadne příslušný tranzistor a dva odpory a provede se propojení, vyznačené čárkovaně. Totéž platí i pro obvod podle obr. 2, kde nejsou inventory zakresleny. Pokud je třeba invertovat data (např. pro DZM-180), lze použít obvody MH 3226 místo 3216, případně MHB 8283 nebo MHB 8287.

Závěr a zapojení (obr. 2 a 3 zpracoval ing. František Komín)

OPRAVA

V článku Debug+ (ZAK 6/87) na str. 21 v 6. řádku má být správně . . . disassembler a **vypisovač** paměti. Na str. 23 pak má být v řádku 17 . . . "ADDRESS RANGE ERROR".



```

*   CENTRONICS PRINTER INTERFACE
*   V1.0  (c)1987  by Vaclav FAJTA
*   =====

```

```

VERSE    EPZ 1      ;1=normal 2=HardCopy
BEGIN    EQU $600
MEMLO    EQU $2E7
DOSVEC   EPZ $0A
PACTL    EQU $D302
PORTA    EQU $D300
PTRIG1   EQU $D011
HATABS   EQU $31A
SAVMSC   EPZ $58
AUX      EPZ $CC
DUM      EPZ $0

```

```
ORG BEGIN
```

```
* HLAVA SAMOZAVADECI CASTI
```

```

DFB 0
DFB (PEND-BEGIN+127)/128
DFW BEGIN ;zavadeči adr.
DFW INIT ;inic.adr.

```

```
* NAVAZUJICI SAMOSTARTUJICI KOD
```

```

LDA $3C ;vyp.mgf.
STA PACTL
LDA $PEND:L
STA MEMLO
LDA $PEND:H
STA MEMLO+1
LDA $INIT:L
STA DOSVEC
LDA $INIT:H
STA DOSVEC+1
CLC
RTS

```

```
* OVLIVNITELNE KONSTANTY hodnoty+1
```

```

LONGLINE DFB $FF ;pocet zn. na radku
KONSTFF  DFB 65 ;pocet radku na form.
KONSTROW DFB 23 ;radku na obrazovce
KONSTCOL DFB 39 ;sloupce na obr.

```

```

CODECR   DFB $0D ;Carriage Return
CODEFF   DFB $0C ;Form Feed
CODELF   DFB $0A ;Line Feed

```

```
* PROMENNE PROGRAMU
```

```

COUNTLIN DFB DUM ;pocitadlo zn. na r.
FORMFEED  DFB DUM ;citac radku na form.
SCRROW    DFB DUM ;cit.radek pro HARDC.
SCRCOL    DFB DUM ;cit.sloupce pro HC.

```

```
* TABULKA RUTIN PRO PARALELNI TISK
```

```

PRINTTAB DFW OPEN-1 ;OPEN
          DFW WRITE-1 ;CLOSE
          DFW RETURN-1 ;GET
          DFW WRITE-1 ;PUT
          DFW RETURN-1 ;STATUS
          DFW RETURN-1 ;SPECIAL
          JMP OPEN ;INIT
          DFB 0 ;dummy

```

```
* RUTINA OTEVIRAJICI 6520 PORTA NA OUT
```

```

OPEN      LDA $38 ;pro definici I/O
          STA PACTL
          LDA $FF ;vsechny bity output
          STA PORTA
          LDA $3C ;uzavreni definice
          STA PACTL
          LDA $80 ;^STROBE=1
          STA PORTA

```

```

RETURN    LDY $1 ;vyzaduje system
          RTS

```

```
* RUTINA RIZENI TISKU
```

```
* I: ..<A> ASCII predava OS
```

```

WRITE     CMP $9B ;EOL
          BNE TISK
CARRET    LDA LONGLINE
          STA COUNTLIN
          DEC FORMFEED
          BPL NOFORM
          LDA CODEFF
          JSR TISK
          INC COUNTLIN
          LDA KONSTFF
          STA FORMFEED
          INC COUNTLIN
          LDA CODECR
          JSR ONECHAR
          LDA CODELF
          JSR ONECHAR
          DEC COUNTLIN
          BEQ CARRET
          BNE RETURN

```

```
* PODPROGRAM VYSTUPU JEDNOHO ZNAKU
```

```

ONECHAR   LDY PTRIG1 ;BUSY=1
          BNE ONECHAR
          LDY $80 ;BUSY=0 tisk.READY
          ORA $80 ;^STROBE=1
          STA PORTA
          AND $7F ;^STROBE=0
          STA PORTA ;DATA pripravana
          STY PORTA
          RTS

```

* INICIALIZACNI A INSTALACNI RUTINA
 * provedena po nahrani, po RESET, po DOS

```
INIT    LDA &PRINTTAB:L
        STA HATABS+1
        LDA &PRINTTAB:H
        STA HATABS+2
        LDA KONSTFF
        STA FORMFEED
        LDA LONGLINE
        STA COUNTLIN
        JSR COPYRIGH
        JMP OPEN
```

* PODPROGRAM VYPISU COPYRIGHT

```
COPYRIGH LDA AUX
        PHA
        LDA AUX+1
        PHA
        LDA &AUTHOR:L
        STA AUX
        LDA &AUTHOR:H
        STA AUX+1
        LDY &0
        LDA (AUX),Y
        STA (SAVMSC),Y
        INY
        CPY &AUTHEND-AUTHOR
        BNE CLOOP
        BEQ AUTHEND
AUTHOR   DFB $80,$C8,$00,$48,$80,$A3
        DFB $A5,$AE,$B4,$B2,$AF,$AE
        DFB $A9,$A3,$B3,$80,$80
VERSION  DFB $B6,$90+VERSE,$8E,$90,$80
        DFB $C8,$00,$48,$80,$88,$E3
        DFB $89,$91,$99,$98,$97,$80
NAME     DFB $A6,$A1,$AA,$B4,$A1,$80
AUTHEND  PLA
        STA AUX+1
        PLA
        STA AUX
        RTS
PEND     EQU *
```

* ZAPIS OBJECT CODE CENTONICS V1.0

```
ORG $5000
MCLOSE 1
MPUTCH BEGIN,PEND
BRK
```

* ----- *

* CENTRONICS PRINTER INTERFACE
 * V2.0 (c) 1987 by Vaclav FAJTA
 * ----- *

```
VERSE   EPZ 2 ;1=normal 2=HardCopy
FYZ      EQU $5000

VER2     EPZ VERSE-1
VER1     EPZ 1-VERSE
BEGIN    EQU $600
MEMLO    EQU $2E7
DOSVEC   EPZ $0A
PACTL    EQU $D302
PORTA    EQU $D300
PTRIG1    EQU $D011
HATABS   EQU $31A
SAVMSC   EPZ $58
AUX      EPZ $CC
DUM      EPZ $0
```

ORG BEGIN,BEGIN+FYZ

* HLAVA SAMOZAVADECI CASTI

```
PNDVER   EQU PEND1*VER1+PEND2*VER2
DOSVER   EQU INIT*VER1+HARDCOPY*VER2

        DFB 0
        DFB (PNDVER-BEGIN+127)/128
        DFW BEGIN ;zavadeci adr.
        DFW INIT ;inic.adr.
```

* NAVAZUJICI SAMOSTARTUJICI KOD

```
LDA #$3C ;vyp.mgf.
STA PACTL
LDA #PNDVER:L
STA MEMLO
LDA #PNDVER:H
STA MEMLO+1
LDA #DOSVER:L
STA DOSVEC
LDA #DOSVER:H
STA DOSVEC+1
CLC
RTS
```

* OVLIVNITELNE KONSTANTY hodnoty+1

```
LONGLINE DFB $FF ;pocet zn. na radku
KONSTFF   DFB 65 ;pocet radku na form.
KONSTROW  DFB 23 ;radku na obrazovce
KONSTCOL  DFB 39 ;sloupcu na obr.

CODECR    DFB $0D ;Carriage Return
CODEFF    DFB $0C ;Form Feed
CODELF    DFB $0A ;Line Feed
```

* PROMENNE PROGRAMU

```
COUNTLIN DFB DUM ;pocitadlo zn. na r.
FORMFEED DFB DUM ;citac radku na form.
SCRROW DFB DUM ;cit.radek pro HARDC.
SCRCOL DFB DUM ;cit.sloupce pro HC.
```

* TABULKA RUTIN PRO PARALELNI TISK

```
PRINTTAB DFW OPEN-1 ;OPEN
          DFW WRITE-1 ;CLOSE
          DFW RETURN-1 ;GET
          DFW WRITE-1 ;PUT
          DFW RETURN-1 ;STATUS
          DFW RETURN-1 ;SPECIAL
          JMP OPEN ;INIT
          DFB 0 ;dummy
```

* RUTINE OTEVIRAJICI 6520 PORTA NA OUT

```
OPEN LDA #$38 ;pro definici I/O
      STA PACTL
      LDA #$FF ;vsechny bity output
      STA PORTA
      LDA #$3C ;uzavreni definice
      STA PACTL
      LDA #$80 ;^STROBE=1
      STA PORTA
```

```
RETURN LDY #1 ;vyzaduje system
        RTS
```

* RUTINA RIZENI TISKU
* I: ..<A> ASCII predava OS

```
WRITE CMP #$9B ;EOL
      BNE TISK
CARRET LDA LONGLINE
      STA COUNTLIN
      DEC FORMFEED
      BPL NOFORM
      LDA CODEFF
      JSR TISK
      INC COUNTLIN
      LDA KONSTFF
      STA FORMFEED
NOFORM INC COUNTLIN
      LDA CODECR
      JSR ONECHAR
      LDA CODELF
      JSR ONECHAR
TISK DEC COUNTLIN
     BEQ CARRET
     LDY #1
     RTS
```

* PODPROGRAM VYSTUPU JEDNOHO ZNAKU

```
ONECHAR LDY PTRIG1 ;BUSY=1
        BNE ONECHAR
        LDY #$80 ;BUSY=0 tisk.READY
        ORA #$80 ;^STROBE=1
        STA PORTA
        AND #$7F ;^STROBE=0
        STA PORTA ;DATA pripravena
        STY PORTA
        RTS
```

* INICIALIZACNI A INSTALACNI RUTINA
* provedena po nahrani, po RESET, po DOS

```
INIT LDA #PRINTTAB:L
      STA HATABS+1
      LDA #PRINTTAB:H
      STA HATABS+2
      LDA KONSTFF
      STA FORMFEED
      LDA LONGLINE
      STA COUNTLIN
      JSR COPYRIGH
      JMP OPEN
```

* PODPROGRAM VYPISU COPYRIGHT

```
COPYRIGH LDA AUX
          PHA
          LDA AUX+1
          PHA
          LDA #AUTHOR:L
          STA AUX
          LDA #AUTHOR:H
          STA AUX+1
          LDY #0
CLOOP LDA (AUX),Y
      STA (SAVMS),Y
      INY
      CPY #AUTHEND-AUTHOR
      BNE CLOOP
      BEQ AUTHEND
AUTHOR DFB $80,$C8,$00,$48,$80,$A3
       DFB $A5,$AE,$B4,$B2,$AF,$AE
       DFB $A9,$A3,$B3,$80,$80
VERSION DFB $B6,$90+VERSE,$8E,$90,$80
        DFB $C8,$00,$48,$80,$88,$E3
        DFB $89,$91,$99,$98,$97,$80
NAME DFB $A6,$A1,$AA,$B4,$A1,$80
AUTHEND PLA
        STA AUX+1
        PLA
        STA AUX
        RTS
```

```

PEND1 EQU * ;konec V1.x
* ***** *

* RUTINA ** HARDCOPY ** kopie obraz.

BEGHCOPY EQU (PEND1+255)/256*256

ORG BEGHCOPY,BEGHCOPY+FYZ

* BASIC: DOS or X=USR (HARDCOPY-1)

BASIC PLA ;vstup z BASIC
HARDCOPY LDA AUX
PHA
LDA AUX+1
PHA
LDA SAVMSC
STA AUX
LDA SAVMSC+1
STA AUX+1
LDA KONSTROW
STA SCROW
LDA KONSTCOL
STA SCRCOL
HC1 LDX #0
HC2 LDA (AUX,X)
AND #$7F ;INV na obyc.
CMP #$60
BCS HC3
ADC #$20 ;korekce na ASCII
HC3 JSR ONECHAR
INC AUX
BNE **4
INC AUX+1
DEC SCRCOL
BPL HC2
LDA CODECR LDA CODELF
JSR ONECHAR JSR ONECHAR
DEC SCROW
BPL HC1
PLA
STA AUX+1
PLA
STA AUX
LDY #1
RTS
PEND2 EQU * ;konec V2.x
* ***** *

* ZAPIS OBJECT CODE CENTRONICS V2.x

ORG $5000
FYZPOC EQU BEGIN+FYZ
FYZEND EQU PNDVER+FYZ
MPUTCH FYZPOC,FYZEND
BRK

```

* ** M A C R O M C I O **

* macroinstrukce pro operace s:
 * Central Input/Output
 * (c)1987 FAJTA Václav

* MACRO ** MOPEN **

* otevreni kan.,PR: MOPEN 1,8,128,"C:"
 * generuje premistitelny stroj.kod

```

MOPEN MACRO KANAL,POVEL,MOD,STRING
LDX #$10*KANAL
LDA #3 ;OPEN
STA $342,X
LDA #POVEL
STA $34A,X ;AUX1
LDA #MOD
STA $34B,X ;AUX2
LDA #ADRSTR$ [5-9]
STA $344,X
LDA #ADRSTR$/256
STA $345,X ;1.adr.buffer
BNE DAL$
BEQ DAL$
ADRSTR$ ASC STRING
DFB $9B ;EOL
DAL$ JSR $E456
BPL OK$
BRK ;error,navrat do MON
OK$ MEND ;navazuje

```

* ----- *

* MACRO *MOPER* R/W CHAR. or RECORD

* CMD: \$5=GET R., \$7=GET CH. (load)
 * \$9=PUT R., \$B=PUT CH. (save)
 * generuje premistitelny stroj.kod

```

MOPER MACRO KANAL,CMD,POCADR,DELKA
LDX #$10*KANAL
LDA #CMD ;IOCB COMMANDO
STA $342,X
LDA #POCADR
STA $344,X
LDA #POCADR/256
STA $345,X ;1.adr.buffer
LDA #DELKA
STA $348,X
LDA #DELKA/256
STA $349,X ;pocet byte
JSR $E456
BPL OK$
BRK ;err.,navrat do MON
OK$ MEND ;navazuje

```

* ----- *

```

* MACRO * MCLOSE *
* uzavira kanal, PR: MCLOSE 1
* generuje premistitelny stroj.kod

MCLOSE MACRO KANAL
LDX ##10*KANAL
LDA #0C ;CLOSE
STA #342,X
JSR #E456
BPL OK$
BRK ;err.navrat do MON
OK$ MEND
* ----- *
* MACRO * MPUTCH* SAVE CHAR. to "c:"
*
* EXTERNAL: MOPEN,MOPER,MCLOSE
* zrychlene nahrava,zaokrouhluje delku
* generuje premistitelny stroj.kod
* I: POC .. prvni adr.souboru
* KON .. posledni adr.soub.

MPUTCH MACRO POC,KON
LEN$ EQU (KON-POC+127)/128*128
MOPEN 1,8,128,"C:"
MOPER 1,8,POC,LEN$
MCLOSE 1
MEND
* ----- *
* MACRO * MGETCH *LOAD CHAR.from "C:"
*
* EXTERNAL: MOPEN,MOPER,MCLOSE
* zrychlene zavadi,zaokrouhluje delku
* generuje premistitelny stroj.kod
* I: POC .. poc.adresa zavedeni
* KON .. posledni adr.souboru

MGETCH MACRO POC,KON
LEN$ EQU (KON-POC+127)/128*128
MOPEN 1,4,128,"C:"
MOPER 1,7,POC,LEN$
MCLOSE 1
MEND
* ----- *
*** CENTRONICS PRINTER INTERFACE ***
* V2.1 (c)1987 by FAJTA Vaclav *
* ----- *
* V2.1 respektuje BASIC prikazy
* LPRINT: PRINT "P:": OPEN #x,8,0,"P:"
*
* umoznuje HARDCOPY obrazu prikazem:
* DOS nebo X=USR(BASIC) nebo >Select<*

* PORT1: pin 1-4 D1-D4
* pin 8 GND,D8
* PORT2: pin 1-3 D5-D7
* pin 4 ^STROBE
* pin 6 BUSY
* pin 8 GND
* ----- *

VERSE EPZ 2 ;1=normal 2=HardCopy
FYZ EQU $5000

VER2 EPZ VERSE-1
VER1 EPZ 1-VER2
BEGIN EQU $600
MEMLO EQU $2E7
DOSVEC EPZ $0A
PACTL EQU $D302
PORTA EQU $D300
PTRIG1 EQU $D011
CONSOL EQU $D01F
HATABS EQU $31A
SAVMSC EPZ $58
AUX EPZ $CC
DUM EPZ $0

ORG BEGIN,BEGIN+FYZ

* HLAVA SAMOZAVADECI CASTI

PNDVER EQU PEND1*VER1+PEND2*VER2
DOSVER EQU INIT*VER1+HARDCOPY*VER2

DFB 0
DFB (PNDVER-BEGIN+127)/128
DFW BEGIN ;zavadeci adr.
DFW INIT ;inic.adr.

* NAVAZUJICI SAMOSTARTUJICI KOD

LDA ##3C ;vyp.mgf.
STA PACTL
LDA #PNDVER:L
STA MEMLO
LDA #PNDVER:H
STA MEMLO+1
LDA #DOSVER:L
STA DOSVEC
LDA #DOSVER:H
STA DOSVEC+1
CLC
RTS

* OVLIVNITELNE KONSTANTY hodnoty+1

LONGLINE DFB $FF ;pocet zn. na radku
KONSTFF DFB 65 ;pocet radku na form.
KONSTROW DFB 23 ;radku na obrazovce
KONSTCOL DFB 39 ;sloupce na obr.

CODECR DFB $0D ;Carriage Return
CODEFF DFB $0C ;Form.Feed
COMASK DFB $02 ;Select
CODELF DFB $0A ;Line Feed

```

* PROMENNE PROGRAMU

```
COUNTLIN DFB DUM ;pocitadlo zn. na r.
FORMFEED DFB DUM ;citac radku na form.
SCRROW DFB DUM ;cit.radek pro HARDC.
SCRCOL DFB DUM ;cit.sloupce pro HC.
```

* TABULKA RUTIN PRO PARALELNI TISK

```
PRINTTAB DFW OPEN-1 ;OPEN
          DFW WRITE-1 ;CLOSE
          DFW RETURN-1 ;GET
          DFW WRITE-1 ;PUT
          DFW RETURN-1 ;STATUS
          DFW RETURN-1 ;SPECIAL
          JMP OPEN ;INIT
          DFB 0 ;dummy
```

* RUTINE OTEVIRAJICI 6520 PORTA NA OUT

```
OPEN LDA #38 ;pro definici I/O
      STA PACTL
      LDA #FF ;vsechny bity output
      STA PORTA
      LDA #3C ;uzavreni definice
      STA PACTL
      LDA #80 ;^STROBE=1
      STA PORTA
```

```
RETURN LDY #1 ;vyzaduje system
RTS
```

* RUTINA RIZENI TISKU

* I: ..<A> ASCII predava OS

```
WRITE CMP #9B ;EOL
      BNE TISK
CARRET LDA LONGLINE
      STA COUNTLIN
      DEC FORMFEED
      BPL NOFORM
      LDA CODEFF
      JSR TISK
      INC COUNTLIN
      LDA KONSTFF
      STA FORMFEED
NOFORM INC COUNTLIN
      LDA CODECR
      JSR ONACHAR
      LDA CODELF
      JSR ONECHAR
TISK DEC COUNTLIN
      BEQ CARRET
      LDY #1
      RTS
```

* PODPROGRAM VYSTUPU JEDNOHO ZNAKU

```
ONECHAR LDY PTRIG1 ;BUSY=1
        BNE ONECHAR
        LDY #80 ;BUSY=0 tisk.READY
        ORA #80 ;^STROBE=1
        STA PORTA
        AND #7F ;^STROBE=0
        STA PORTA ;DATA pripravana
        STY PORTA
        RTS
```

* INICIALIZACNI A INSTALACNI RUTINA

* provedena po nahrani, po RESET, po DOS

```
INIT LDA #7
      LDY #HCUSR20:L
      LDX #HCUSR20:H
      JSR #E45C ;set vector
```

```
NOSEL LDA #PRINTTAB:L
      STA HATABS+1
      LDA #PRINTTAB:H
      STA HATABS+2
      LDA KONSTFF
      STA FORMFEED
      LDA LONGLINE
      STA COUNTLIN
      JSR COPYRIGHT
      JMP OPEN
```

* PODPROGRAM VYPISU COPYRIGHT

```
COPYRIGHT LDA AUX
          PHA
          LDA AUX+1
          PHA
          LDA #AUTHOR:L
          STA AUX
          LDA #AUTHOR:H
          STA AUX+1
          LDY #0
CLOOP LDA (AUX),Y
      STA (SAVMS),Y
      INY
      CPY #AUTHEND-AUTHOR
      BNE CLOOP
      BEQ AUTHEND
AUTHOR DFB #80,#C8,#00,#48,#80,#A3
      DFB #A5,#AE,#B4,#B2,#AF,#AE
      DFB #A9,#A3,#B3,#80,#80
VERSION DFB #B6,#90+VERSE,#8E,#91,#80
      DFB #C8,#00,#48,#80,#88,#E3
      DFB #89,#91,#99,#98,#97,#80
NAME DFB #A6,#A1,#AA,#B4,#A1,#80
AUTHEND PLA
      STA AUX+1
      PLA
      STA AUX
      RTS
```

```

PEND1 EQU * ;konec v1.x
* *****
* RUTINA * HARD COPY * kopie obraz.
* BASIC: DOS nebo X=USR (BASIC)
* sllaceni: >Select<

BASIC PLA ;vstup z BASIC
HARDCOPY LDA AUX
PHA
LDA AUX+1
PHA
LDA SAVMSC
STA AUX
LDA SAVMSC+1
STA AUX+1
LDA KONSTROW
STA SCROW
HC1 LDA KONSTCOL
STA SCRCOL
LDX #0
HC2 LDA (AUX,X)
AND #$7F ;INV na obyc.
CMP #$60
BCS HC3
ADC #$20 ;korekce na ASCII
HC3 JSR ONECHAR
INC AUX
BNE **4
INC AUX+1
DEC SCRCOL
BPL HC2
LDA CODECR
JSR ONECHAR LDA CODELF
DEC SCROW JSR ONECHAR
BPL HC1
PLA
STA AUX+1
PLA
STA AUX
LDY #1
RTS

* RUTINA *VERTICAL BLANC INTERRUPT*
HCUSR20 LDA CONSOL
AND COMASK
BNE NOHC
JSR HARDCOPY
NOHC JMP #E462 ;EXITVBL

PEND2 EQU * ;konec V2.x
* ZAPIS OBJECT CODE CENTONICS V2.x

ORG #5000
FYZPOC EQU BEGIN+FYZ
FYZEND EQU PNDVER+FYZ
MPUTCH FYZPOC,FYZEND
BRK

```

PŘIPOJENÍ MAGNETOFONU A SYSTÉM TURBO 2000

Ing. Petr Havlíček

(pokračování z čísla 6/1987)

Úprava magnetofonu ATARI XC-12 na ochranu magnetofonového pásku

Cílem této úpravy je ochrana méně odolných magnetofonových pásků před vytlačněním „drop outů“ v místě, kde přítlačná kladka tiskne pásek k ose setrvačnicku.

Úprava spočívá v doplnění magnetofonu spínačem, kterým lze zvolit dva režimy provozu magnetofonu: původní „se zastavováním“ a nový režim „bez zastavování“ motoru. Dodejme ještě, že když nebude me nechávat stisknuté tlačítko PLAY v době, kdy se motor neotáčí, chráníme před deformací společně s páskem také pryžové kolečko přítlačné kladky.

Obr. 2 schematicky znázorňuje, jak je u magnetofonu ATARI XC-12 řešeno ovládání motoru. Je odvozeno jednak od stavu tlačítek magnetofonu a jednak od úrovně signálu MOTOR CTRL.

Signál MOTOR CTRL je přiváděn přes odpor 27k na bázi prvního tranzistoru, který společně s mechanickými spínači P2 a P3 tvoří vlastně trojici paralelně zapojených spínačů. Jednou z podmínek nutných ke spuštění chodu motoru je sepnutý alespoň jeden z nich. Spínače P2 a P3 jsou ovládány tlačítky klávesnice magnetofonu, kterými se volí funkce převíjení (REWIND<<F.FWD>>). Oba spínače jsou dobře patrné při pohledu pod desku plošných spojů ze strany od kláves. Jejich mechanické provedení je velmi jednoduché: spínač tvoří dvojice pružných kontaktů zapájených přímo do desky plošných spojů. (Rychlost a směr převíjení je dána mechanickým převodem.)

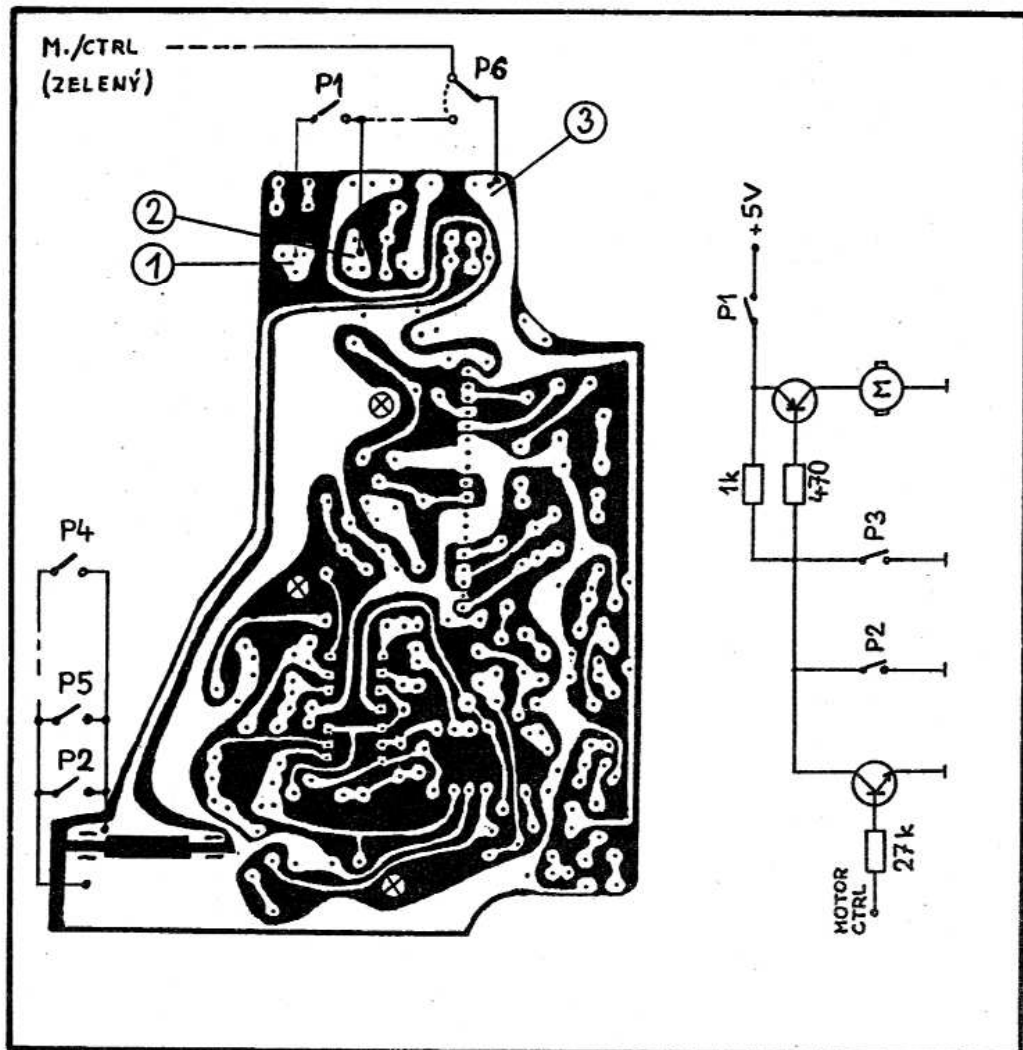
Magnetofon upravíme tak, že ke spínačům P2 a P3 zapojíme paralelně další spínač P4 (obr. 1) a upevníme jej do horní desky krytu magnetofonu. Režim „bez

zastavení" potom zvolíme sepnutím kontaktů spínače P4. Motor se bude otáčet, kdykoliv stiskneme tlačítko PLAY. O tom, zda přitom ze zvukového kanálu uslyšíme záznam na kazetě, rozhoduje nadále úroveň signálu MOTOR CTRL, který zajišťuje napájení snímacích zesilovačů.

Tato úprava také částečně řeší problém vyhledávání začátků programů, byť sice neumožňuje kontrolovat nahrávku poslechem, ale pomalý posun pásky usnadní přesné nastavení vestavěného počítače.

Obr. 1 Magnetofon ATARI XC-12 — deska plošných spojů

Obr. 2 Obvody řízení motoru u magnetofonu ATARI XC-12



Úprava magnetofonu ATARI XC-12 pro snazší vyhledávání začátků programů

V článku (2) je popsána tato úprava: mezi body 1 a 3 (obr. 1) zařadit spínač (dále spínač P5) a zdroj signálu MOTOR CTRL v počítači (zelený vodič) oddělit od bodu 3 diodou. (Anoda diody je spojena se zeleným vodičem, katoda je připájena na desku spojů v bodě 3.)

Tato úprava má dvě slabá místa, kterým se současně nelze vyhnout:

1. Je-li spínač P5 sepnut a je-li současně stlačena klávesa PLAY, otáčí se motor a snímací zesilovače magnetofonu jsou napájeny, takže ze zvukového kanálu slyšíme zvuk nahraný na pásku. Zesilovače i motor jsou však napájeny ze společného zdroje, takže rušení od motoru proniká k ostatním obvodům magnetofonu, o čemž svědčí zřetelně slyšitelné charakteristické bručení ve zvukovém kanálu. Magnetofon s takto napájenými obvody sice programy nahrává i snímá, ale pokud nám záleží na spolehlivosti systému, dáme vždy přednost kvalitnímu napájení a spínač P5 budeme používat pouze k vyhledávání začátku daného programu.

2. Jestliže bude tedy záznam a snímání probíhat při rozpojených kontaktech spínače P5, budou obvody magnetofonu napájeny ze signálu MOTOR CTRL, který sice není rušen motorem, ale úroveň jeho napětí jsme zmenšili o 0,6 V předřazením oddělovací diody. Úbytek napětí na diodě představuje více než 10 % napájecího napětí, což opět zhoršuje spolehlivost záznamu.

Problém je možno řešit tak, že místo spínače zapojíme přepínač P6 (obr. 1), který spojí bod 3 buďto s bodem 2 nebo se zeleným vodičem MOTOR CTRL od počítače. Tak dosáhneme stejného efektu jako v úpravě podle (2) s tím rozdílem, že dioda může být vynechána, protože nemůže dojít ke zkratu signálu READY/+5V a MOTOR CTRL.

Pro ty, kteří si upravili magnetofon XC-12 pro snazší vyhledávání začátků

programů podle (2) nebo podle předcházejícího návodu a současně chtějí doplnit magnetofon rozhraním pro systém TURBO 2000 dodejme, že signál MOTOR CTRL pro TURBO 2000 nutno brát přímo ze zeleného vodiče (od počítače) a ne z desky spojů, jak je naznačeno v (1).

Signál MOTOR CTRL jako adresa magnetofonu

Vyjma magnetofonu obsahují zařízení určená k připojení na sériovou sběrnici (tiskárna, disketová jednotka atd.) vlastní řadič, který zajišťuje komunikaci s počítačem. Řadič v podstatě analyzuje signál DATA OUT, rozliší, kterému zařízení je počítačem vysílána zpráva určena, a odpovídá po vodiči DATA IN pouze tehdy, je-li k tomu počítačem vyzván. Protože počítač vždy nejdřív ukončí přenos dat od jednoho zařízení a až potom zahájí přenos od jiného zařízení, nemůže na vodič DATA IN (od vnějších zařízení do počítače) dojít ke kolozí.

Aby ani magnetofon nemohl rušit přenos v době, kdy s počítačem komunikuje jiné zařízení, bylo nutné vyhradit magnetofonu na sběrnici zvláštní vodič (MOTOR CTRL) a jím vysílání magnetofonu zakazovat (nebo chcete-li povolovat). Obvodové řešení této funkce u standardního rozhraní firemního magnetofonu je nesmírně jednoduché a vychází z toho, že signál MOTOR CTRL zajišťuje také napájení obvodu magnetofonu. Je-li signál MOTOR CTRL ve stavu „vypnuto“, jsou obvody rozhraní odpojené od napájení a tím je automaticky zajištěno, že se nemůže otevřít budicí tranzistor připojený kolektorem na vodič DATA IN. Vodič DATA IN je potom v magnetofonu připojen už jen na odpor 68k (v kolektoru budicího tranzistoru). Hodnota odporu je dostatečně velká, takže signál DATA IN nemůže být podstatně ovlivněn, protože v klidovém stavu je na něm úroveň log. 1 zajištěna odporem 4k7 (v počítači) zapojeným mezi vodič DATA IN a +5V.

Princip buzení vodiče DATA IN u roz-

hraní TURBO 2000 je stejný, jako u standardního rozhraní, protože použitý obvod 7403 má výstupy hradel „s otevřenými kolektory“. Podíváme-li se nyní na zapojení přídavného rozhraní TURBO 2000, můžeme říci, že z hlediska napájení, významu signálu MOTOR CTRL i způsobu buzení signálu DATA IN není rozdíl mezi standardním a přídavným rozhraním. Z toho vyplývá několik zjednodušujících úprav zapojení přídavného rozhraní podle (1):

1. Lze vynechat vodič +5V, který je přídavné rozhraní připojeno k desce standardního rozhraní. Napájení potom vedeme od signálu MOTOR CTRL tak, že vodič na desce odpojíme z bodu 8 a připojíme do bodu 10 (viz (1), str. 23).
2. Dále můžeme vynechat tranzistor T4 a odpor 10k v jeho bázi, přičemž body na desce spojuj, ve kterých byl kolektor a emitor tranzistoru, propojíme drátovou propojkou nebo mezi ně zapojíme jednu z dvojice doporučených diod na výstupu operačního zesilovače (viz schéma zapojení v (1)).
3. Odpor 3k3 na výstupu log. integrovaného obvodu (připojen na vodič sběrnice DATA IN) musíme nahradit odporem 68k (100k). Hodnota tohoto odporu není kritická.

K úpravě magnetofonu na systém TURBO 2000

Při návrhu desky spojů přídavného rozhraní pro TURBO 2000 se zapomělo na

blokovací kondenzátor na napájení. Chybějící kondenzátor by se mohl uživateli vymstít zvláště tehdy, když použil nejrychlejší z doporučených logických obvodů 7403 S. K blokování stačí keramický kondenzátor s kapacitou 22n až 100n, který je možné připojit shora přímo na vývody 7 a 14 logického integrovaného obvodu 7403.

Kdo k nahrání programů používá běžný kazetový magnetofon, musel při úpravě na systém TURBO 2000 doplnit přídavným rozhraním již dříve amatérsky vyrobené rozhraní. Nejrozšířenější používané zapojení je shodné se zapojením obvodů ve firemním magnetofonu XC-12 (jeho schéma naleznete v (1)). V tomto zapojení předchází dvěma operačním zesilovačům, které jsou zapojeny jako pásmové propusti, operační zesilovač pracující jako tvarovač signálu. Tento vstupní operační zesilovač je buzen (přes vazební kondenzátor a odpor 1k) z kolektoru druhého tranzistoru snímáčního zesilovače, odkud je signál veden rovněž k operačnímu zesilovači v přídavném rozhraní systému TURBO 2000. Zde je důležité si všimnout, že vstupy obou rozhraní jsou buzeny ze stejného bodu, ale jsou vzájemně odděleny zmíněným odporem 1k, který zabraňuje tomu, aby zpětná vazba vstupního tvarovače signálu ovlivňovala vstupní signál pro TURBO 2000. Kdo tedy doplňuje svoje amatérsky vyrobené rozhraní na systém TURBO 2000, neměl by zapomenout obdobným způsobem oddělit vstupy obou rozhraní.

Literatura:

- (1) Richter, J.: TURBO 2000 — systém zrychleného přenosu dat z magnetofonu ATARI. Zpravodaj ATARI KLUBU Praha, příloha II, Praha, 1987.
- (2) Machulda, M.: Úprava magnetofonu ATARI XC-12 pro snadnější vyhledávání začátků programů. Zpravodaj ATARI KLUBU Praha, 1987, 2, s. 24—25.

Poznámka: V literatuře (1) na str. 19—24 byly názvy DATA IN a DATA OUT označeny opačně, než je tomu na zdiřce počítače, protože tam šlo o označení vstupu a výstupu z hlediska magnetofonu. Správné označení by tedy mělo být MGF DATA IN a MGT DATA OUT. Z hlediska počítače je bílý vodič (zdiřka 3) DATA IN, hnědý vodič (zdiřka 5) DATA OUT.

Prodloužení životnosti barvicí pásky pro tiskárny ATARI 1029

Imrich Martišek, Bratislava

Barvicí páska dodávaná k tiskárnám ATARI 1029, jak potvrdí mnozí uživatelé této tiskárny, nevydrží dlouhou životnost. Po spotřebování čtyř kusů jsem začal přemýšlet, jak si pomoci. Řešení je následující:

1. Sejmeme víko kazety, vyjmeme pásku a zkontrolujeme její pohon, který tvoří kolík s pryžovým kroužkem a přítlačnou planžetou. Kroužek bývá rozleptaný, zmačknutý nebo úplně deformovaný, což způsobuje navinutí pásky na pohon kolíku. Kroužek nahradíme novým, odříznutým z hadičky hustilky, resp. s čepičky automobilové vzdušnice.

2. Odstraníme přepážku (na opačné straně výstupu pásky z kazety) vybroušením nebo vyřezáním ostrým nožem, skalpelem apod. Páska bude méně zmačkaná a lépe se bude odvíjet.

3. Do prostoru před výstupem pásky z kazety vlepieme lepidlem Lepi-M, Polystyren cement aj. hranol z jemného filcu o rozměru 8 x 5 x 3 mm. Rozměr není kritický. Jde o to, aby se páska při výstupu z kazety otírala o filcový hranol.

4. Do víka kazety v místě nad filcovým hranolem vyvrtáme plnicí otvor na barvu o průměru cca 3 mm.

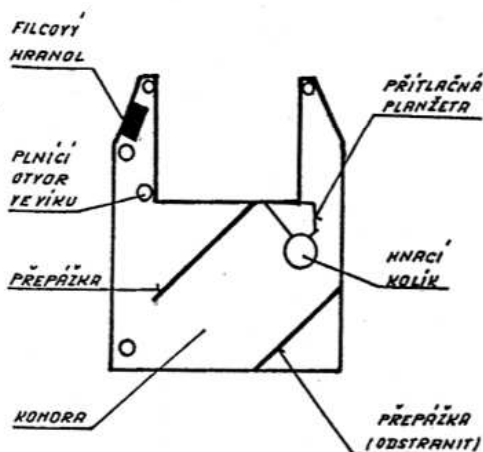
Kazetu složíme, pásku navineme otáčením hnacího kolíku rukou, přičemž kontrolujeme lehký posuv pásky. Páska má být po navinutí v místě dotyku s hlavou tiskárny mírně napnutá. Případné nedostatky lze napravit přihnutím přítlačné planžety, která zajišťuje přítlačení pásky k hnacímu kolíku (nastavení tahu), resp. vynechání opásání pásky v zadní části komory (uvolnění odvíjení).

Plnicím otvorem ve víku kazety kápneme (injekční stříkačkou, kapátkem) na filcový hranol 3—4 kapky „Barvy na číselničky“ (prodává se v maloobchodě za 4,20 Kčs), mírně ředěná lihem.

Přetočíme pásku rukou a vyzkoušíme kontrast tisku. Při silném nasycení filcu je písmo příliš černé a slité (nečitelné). Důvodem je

přesycení filcu barvivem. Mohou nastat i poruchy posuvu, například páska klouže na hnacím kolíku. V tomto případě pásku vyjme- me a vysušíme ji sacím papírem.

Takto upravenou kazetu s barvicí páskou používám skoro rok (asi 2—3 krát týdně), přičemž většinou tisknu grafiku. Při snížení kontrastu tisku přikápnu barvivo bez vyjmutí kazety z tiskárny. Životnost takto upravené kazety závisí zřejmě na mechanické životnosti nylonové pásky, na které dosud nepozoruji opotřebování.



Poznámka redakce: Na první pohled velmi vtipné řešení. Jak se ale vyjádřili odborní poradci, hrozí jedno velmi vážné nebezpečí. Není totiž zaručeno, že rozpouštědla obsažená v barvivu nemohou narušit speciální lepidlo, kterým jsou zalepena diamantová pouzdra jehel v tiskové hlavě. Proto pozor — prodloužení životnosti barvicí pásky — ale na vlastní nebezpečí.

Studený start pro počítače ATARI

ing. Karel Jiříčka, Praha

Jistě jste se všichni již mnohokrát dostali do situace, kdy po zmáčknutí tlačítka RESET se nahraný program znovu inicializuje a spouští od začátku, a žádný ovládací prvek počítače (kromě síťového vypínače) neumožňuje tento program opustit. To je způsobeno tím, že u počítače ATARI se hardwarový systémový reset (ovládaný tlačítkem RESET) využívá jak pro studený, tak i pro teplý start, a jejich rozlišení se provádí softwarově. Ukazuje-li proto návratová adresa uložená v registru DOSVEC na počátek programu a neumožňuje-li právě běžící program změnit hodnotu určitých registrů (jako např. BASIC příkazem POKE 830,0), pak jediná možnost jak smyčku opustit je vypnout počítač a počkat, až se vybijí paměťové kapacity v dynamické paměti RAM.

Časté zapínání a vypínání však počítači příliš nesvědčí, protože, jak je známo z dlouhodobých zkoušek spolehlivosti systémů, vzniká nejvíce poruch právě při zapínání proudovými a teplotními šoky. Proto byla snaha nalézt způsob, jak provádět studený start bez nutnosti vypínání celého počítače. V následujícím je uvedena jedna možnost řešení.

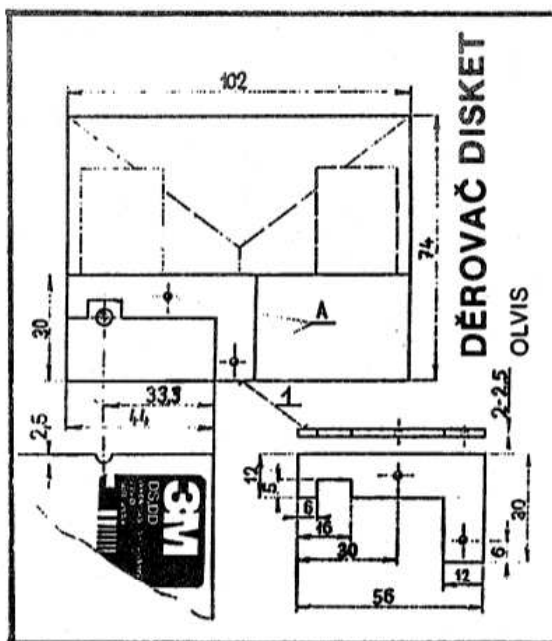
Vstup, který se u dřívějších verzí počítače využíval pro tlačítko TRIGGER u 4. joysticku a který je možno snímat přes adresu D013, je současně vyveden pod označením M5 na konektoru pro programový modul (CARTRIDGE). Pokud není modul zasunut je na tomto vývodu log. "0", při zasunutí modul na tento vstup přivádí log. "1", která odpojuje vnitřní paměť (RAM i ROM) na adresách A000-BFFF. Při každém zapnutí počítač se jme hodnotu M5 a uloží ji do paměťové buňky 3FA, a při každém zmáčknutí tlačítka RESET počítač nejprve testuje, zůstaly-li tyto dvě hodnoty stejné. Pokud ano, pak počítač provede teplý start, pokud ne (byl-li např. za

provozu vyjmut nebo zasunut paměťový modul), pak provede start studený. Tyto skutečnosti umožnily následující řešení studeného startu:

Mezi vývod M5 a +5V je přes ochranný odpor 680 Ω připojeno tlačítko TL, které umožňuje měnit za provozu logickou úroveň na vstupu M5 a tak přinutit počítač provádět namísto teplého startu start studený. Postup je pak následující:

1. Zmáčknout tlačítko TL, pak krátce zmáčknout tlačítko RESET. Tlačítko TL držet zmáčknuté až do doby, kdy se na obrazovce objeví známý nápis SELF TEST.
2. Poté máme podle potřeby dvě možnosti:
 - a) zmáčknutím RESET návrat do BASICu,
 - b) zmáčknutím RESET + START (+ OPTION) nahrát z kazety nový strojový program.

Ovládací tlačítko TL může být umístěno na na konektoru a zasunuto do patice pro CARTRIDGE nebo, což je výhodnější, umístěno přímo na základní desce počítače a vy-



vedeno vyvrtanou dírou na zadní straně počítače. Tuto úpravu však doporučuji přenechat odborníkovi.

Poznámky: 1. Pokud při nahrávání programu byla zmáčknutím OPTION odpojena vnitřní ROM a v oblasti RAM A000-BFFF je umístěna obrazová paměť, objeví se při zmáčknutí tlačítka RESET na obrazovce svislé pruhy (což není na závadu).

2. Pokud program využívá nebo je umístěn v oblasti A000-BFFF, je při zmáčknutí tlačítka TL obvykle nenávratně ztracen. To pak znamená mačkat tlačítko TL pouze pokud chceme program zničit úmyslně a provádět studený start.

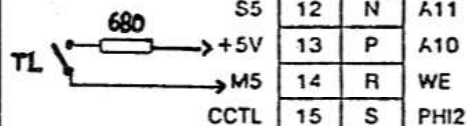
STUDENÝ START —

provádí se vždy při zapnutí. Systém je plně inicializován, paměť RAM vynulována.

TEPLÝ START —

provádí se při zmáčknutí tlačítka RESET. Systém je částečně inicializován, obsah paměti RAM (a tím i uložený program) zůstává zachován.

Obr. 1. Zapojení konektoru CARTRIDGE a připojení tlačítka TL



Mezi ataristy je již řada uživatelů disketové jednotky ATARI 1050. Tato jednotka zapisuje (a čte) pouze na tu stranu diskety, která je v zasunuté poloze směrem nahoru.

Chceme-li pro zápis využít i druhou stranu diskety, musíme v boční straně diskety (pouzdra) zhotovit výřez. Vzhledem k tomu, že výřez je kontrolován průchodem světelného paprsku, nezáleží na jeho tvaru, ale pouze na rozměrech a správném umístění. Speciální děrovače, které se vyrábí v zahraničí, se k nám nedovážejí, takže opět přichází na řadu šikovné české ručičky ...

Někteří kolegové vysekávají výřez speciálním nástrojem, jiní jej prostě vystřihují. Mě však napadlo řešení, které je finančně a výrobne nenáročné, přičemž se v praxi výborně osvědčuje.

K výrobě děrovače disket lze použít kancelářskou děrovačku papíru, která se dá zakoupit v kterémkoliv papírnictví. Vhodné

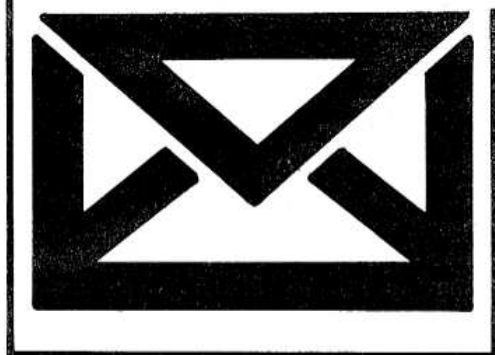
jsou typy tuzemské výroby KIN 640 nebo 641, které se dají jednoduchým způsobem rozebrat.

Po vyjmutí děrovacích nožů připevníme do děrovače podle obrázku zarážku (1), která přesně vymezuje umístění výřezu na disketě. Zarážku je třeba vyrobit z materiálu silného 2 - 2,5 mm, nejlépe z plechu, ale je možné použít i jiného materiálu, např. kuprexitu, plexiskla apod. V bodech A připevníme zarážku šroubkem nebo nýtem. Upevnění zarážky vylepšíme použitím vhodného lepidla (podle druhu použitého materiálu). Obdobným způsobem můžeme upravit obě strany děrovačky. Po provedení úpravy namontujeme zpět děrovací nože.

Z vlastní praxe mohu potvrdit, že po perforaci cca 200 ks disket nedošlo k podstatnému otupení nožů. Až k tomu dojde, výhodou mého návrhu je i to, že nože lze snadno vyměnit nebo nabrousit.

LISTárna

ZEPTALI JSME SE ZA VÁS ... JAK SPRÁVNĚ POSÍLAT KAZETY A DISKETY POŠTOU



Efektivní využívání výpočetní techniky předpokládá dostatečné softwarové vybavení uživatele. Ve značkových klubech výpočetní techniky se stala naprosto běžným jevem výměna programů nahraných na magnetických médiích. Jak ale správně postupovat při zasílání kazet či disket poštou? Smí se vůbec kazety a diskety poštou posílat? Jak takové zásilky zabalit a popsat? Kdy přebírá pošta odpovědnost za takovou nedoručenou zásilku? Odpovídá pošta i za případnou ztrátu dat (programů) nahraných na kazetě či disketě?

Právě na tyto a další otázky jsme se zeptali na Poště Praha 1 v Jindřišské ulici. Dostali jsme v podstatě velice jednoduchou a prostou informaci: „Je třeba postupovat podle Poštovního řádu“.

Ačkoliv Poštovní řád je veřejně přístupný a kterákoliv pošta je povinna dát na žádost uživatele poštovních služeb k nahlédnutí příslušná ustanovení Poštovního řádu a předpokládejme, že spolu s potřebným výkladem, nepovažovali jsme za účelné opsat a na stránkách Zpravodaje AK otisknout celý Poštovní řád.

Předně musíme poznamenat, že v žádném ustanovení Poštovního řádu se s pojmem „přepřava magnetických médií“ nesetkáme. Znamená to, že pošta např. nepřijímá odpovědnost za znehodnocení či úplnou ztrátu obsahu dat zaznamenaných na pásce nebo disketě, k čemuž může dojít např. tehdy, dostane-li se taková zásilka do blízkosti parazitního elektromagnetického záření. V tomto případě je nepodstatné, zda-li došlo k poškození obalu zásilky či nikoliv, zda-li bylo na zásilce uvedeno, že obsahuje magnetické médium apod. Jak bude uvedeno dále, riziko za obsah zásilky, resp. za „úbytek“, nese odesílatel.

Z praxe zkušených kolegů jsme k tomuto technickému problému poskládali mozaiku doporučení, která dále uvádíme. Podotýkáme, že vycházíme i z poznatků zahraničních kolegů (z SSSR). Jde o to, že některé zásilky přepravované v mezinárodním styku jsou celními orgány kontrolovány přístroji na bázi rentgenu. **Z technického hlediska** jsou tedy doporučována následující pravidla pro balení a zasílání magnetických médií:

- kazety nebo diskety vkládat zásadně do pevných obalů (např. kartonové či plastické, příp. i kovové krabičky),
- jednotlivé kazety nebo diskety, příp. pevný obal dokonale zabalit do kovové fólie (např. ALOBAL),
- kazety nebo diskety uvnitř pevných obalů zajistit proti pohybu (např. ústřížky molitanu, zmačkaného papíru atd.),

— pevný obal (krabičku) zabalit do papírového obalu a zásilku vybavit podle obecně známých a platných pravidel zakotvených v Poštovním řádu. Nedoporučuje se vkládat pevný obal (krabičku) volně do obyčejných nezpevněných obálek, protože hrozí protržení obálky (např. o roh krabičky, při manipulaci atd.).

Poznámka: Doporučení o zabalení magnetického média do kovové fólie s cílem ochrany před parazitním elektromagnetickým zářením není naší redakcí prověřeno. Rádi přijmeme poznatky těch, kteří takové zkoušky prakticky uskuteční.

Z hlediska ustanovení Poštovního řádu vyjímáme následující podstatné informace:

Obecně k zásilkám

Kazetu nebo disketu lze zaslat jako **listovní zásilku** (dopis) nebo jako **balík**.

Listovní zásilku lze podat doporučeně, balík s udanou cenou.

Zásilky dopravované poštou musí být zabaleny a uzavřeny. Volit vhodný obal a uzávěru je věcí odesílatele. Obal musí za všech okolností dostatečně chránit obsah zásilky před ztrátou, úbytkem nebo poškozením a musí být přiměřený způsobu a délce přepravy, hmotnosti, ceně a povaze obsahu. Obal musí chránit obsah zásilky tak, aby neutrpěl tlakem nebo zacházením za přepravy. Obal musí chránit obsah zásilky též proti změnám teploty a vlhkosti ovzduší.

Uzávěra musí být zařízena tak, aby nebylo možno se dostat k obsahu bez jejího poškození nebo uvolnění. U balíků, jejichž obal tvoří krabice s chlopněmi, musí být obsah zajištěn tak, aby vsunutím ruky pod chlopně nebyl dosažitelný.

Obal a uzávěra musí být upraveny tak, aby nemohly poranit osoby nebo poškodit věci a aby neztěžovaly zacházení se zásilkami.

Z důvodu zřetelnosti a čitelnosti adres, otisků poštovních denních razítek a záznamů pošty nesmějí být používány obálky nebo obaly zásilek tmavé barvy.

Ke tvaru, rozměrům a hmotnosti zásilky

Psaní se zasílají v uzavřených obálkách, které musí mít tvar ležatého obdélníku; není-li možno vložit jejich obsah do obálek, je věcí odesílatele, jak psaní zabalí a uzavře. Listovní zásilky neobvyklých tvarů, jako např. kulaté, oválné, trojhranné nebo ve tvaru koule apod., nejsou k dopravě poštou dovoleny.

Pro poštovní zásilky je dovolena nejvyšší hmotnost:

- a) pro psaní max. 1 kg,
- b) pro balíky max. 15 kg.

Pro **psaní** jsou stanoveny tyto rozměry:

- největší rozměr — součet délky, šířky a výšky nesmí být větší než 90 cm a žádný z rozměrů nesmí přesahovat 60 cm,
- nejmenší dovolený rozměr je 14 x 9 cm.

Pro **balíky** nejsou stanoveny největší dovolené rozměry; rozhodující je, aby balík mohl být od podací pošty přepraven používanými dopravními prostředky. Nejmenší dovolený rozměr je stejný jako u psaní. Pro neskladné křehké balíky je nejmenší stanovený rozměr — součet délky, šířky a výšky, nebo jde-li o svitek součet délky a dvojnásobného průměru — ne menší než 60 cm.

K adrese příjemce a odesílatele a dalším poznámkám

Na poštovních zásilkách musí být adresa příjemce a odesílatele uvedena čitelně a tak přesně, aby nebylo pochybností, kam se má zásilka dopravit, komu a kde se má dodat. Adresa příjemce zásilky musí být umístěna v pravé dolní čtvrtině adresní strany zásilky.

Adresa musí obsahovat tyto údaje: jméno a příjmení adresáta nebo nezkrácený název organizace; název ulice nebo náměstí a orientační číslo domu; nemá-li dům orientační číslo přidělené, uvádí se číslo popisné; u zásilek určených do více podlažních budov se doporučuje uvádět i číslo poschodí, popř. číslo bytu; je-li uživatelem bytu jiná osoba než adresát, je nutno doplnit adresu jménem a příjmením uživatele bytu, např. „u Jiřího Nováka“. Adresa musí dále obsahovat místo určení a u místa, ve kterém není dodávací pošta, i nezkrácený název dodávací pošty, která do tohoto místa zásilky dodává.

Nedílnou součástí názvu dodávací pošty je poštovní směrovací číslo. PSČ se píše nalevo od názvu dodávací pošty; u zásilek s předtištěnými rámečky pro PSČ se píše do těchto rámečků. Na zásilkách do velkých měst se u místa určení vyznačí ještě číslo městského obvodu a název části města.

Adresa a ostatní údaje adresní strany mohou být napsány, vytištěny nebo pořízeny rozmnožovacím způsobem. Při vyplňování adresy příjemce nesmí být použito červené nebo luminiscenční barvy. Adresu, údaj ceny dobírky, vyplacené částky a ostatní poznámky na zásilkách je dovoleno napsat jen inkoustem, kuličkovou tužkou, psacím strojem, tiskem, tiskárnou počítače nebo otiskem razítka. Pořizovat tyto údaje průpisem a průklepem psacích strojů není dovoleno.

Adresa odesílatele se uvádí na zásilkách, není-li Poštovním řádem stanoveno jinak, v levém horním rohu adresní strany zásilky, nebo na její zadní straně.

Na adresních stranách poštovních zásilek je kromě adresy příjemce a odesílatele možno uvádět jen předepsané označení a poznámky týkající se druhu zásilek, výplatného a zvláštního zacházení se zásilkou. Poznámky a označení mohou být napsány, vytištěny nebo pořízeny otiskem razítka. K jejich vyznačení nesmí být použito luminiscenční barvy. Není dovoleno uvádět na zásilkách jakékoliv podnikové nebo jiné znaky a nálepky, poznámky o vloženém obsahu apod.

K, možnostem přepravy zásilek

Odesílatel může žádat, aby pošta dodala jeho doporučenou zásilku, cenné psaní nebo balík jen pod podmínkou, zaplatí-li příjemce **dobírku**.

Odesílatel doporučené zásilky, cenného psaní nebo balíku může žádat, zaplatí-li stanovený poplatek, aby mu pošta opatřila potvrzení příjemce, že zásilku obdržel (žádat o **doručenku**). Žádá-li odesílatel o doručenkou, musí zásilku, u balíku též i příslušnou průvodku, označit na adresní straně nápadnou poznámkou „DORUČENKA“ a připojí doručenkou.

Zapsanou zásilku s doručenkou doručí pošta adresátovi, jeho manželu nebo jeho poštovnímu zmocněnci. Chce-li odesílatel, aby zapsaná zásilka s doručenkou byla doručena jen adresátovi a nikomu jinému, označí zásilku na adresní straně (u balíku i průvodku) poznámkou „DO VLASTNÍCH RUKOU“.

Odesílatel může při podání žádat, aby zásilka byla doručena spěšně, tj. co nejdříve po příchodu na dodací poštu. Jako spěšné zásilky nelze podat balíky vnitřního styku, letákové zásilky, nádražní psaní a všechny ostatní zásilky, které jsou adresovány do míst, v nichž není pošta. Pilné balíky se spěšně doručují i bez žádosti odesílatele, a to do všech míst, tedy i do těch, ve kterých není pošta.

Pošta odpovídá jen za zapsané zásilky, které byly podány k poštovní přepravě podle Poštovního řádu.

Odesílatel nebo adresát mohou žádat u kterékoliv pošty, aby se pátralo po zásilce, zda byla správně dodána. Jde-li o zapsanou zásilku, musí předložit podací stvrzenku. Žádat o pátrání lze do roka, počínaje od následujícího dne po dni podání zásilky k poštovní přepravě.

☆☆☆

Z uvedených všeobecných informací vyplývajících z Poštovního řádu je **pro zasílání kazet či disket nejdůležitější si zapamatovat:**

- kazety nebo diskety lze poslat jako listovní zásilku (dopis) nebo jako balík,
- vzhledem k tomu, že pošta odpovídá jen za zapsané zásilky, je vhodné kazety nebo diskety zabalené ve formě listovní zásilky zasílat doporučeně. V tomto případě má odesílatel - tak jako při podání balíku - stvrzenku od pošty o přijetí zásilky k poštovní přepravě a může tak tuto zásilku reklamovat v případě pochybností o správném doručení,
- je třeba dbát na dodržení předepsaných rozměrů (i minimálních) a hmotnosti zásilky, jakož i na úpravu adresní strany,
- v případě zasílání většího množství kazet nebo disket je vhodné takovou zásilku poslat jako balík s udanou cenou.

připravil Dr. Jan Hlaváček

Několik poznámek k šíření systému TURBO 2000

TURBO 2000

Dr. Jiří Tichota

Systém zrychleného přenosu dat - TURBO 2000, hardwarově i softwarově realizovaný Jiřím Richterem, umožnil převratné proměny nejen ve vlastní manipulaci s programy, ale významně ovlivnil i jejich zpřístupnění a šíření v celé členské základně ATARI KLUBU. Navíc, a to je důležitější, přivedl a stále přivádí nové uživatele k hlubšímu zájmu o strukturu upravovaných programů, takže se seznamují se základy strojového kódu, s prací s monitory, disasemblyery atd. Že takovéto hnutí má i některé stinné stránky je pochopitelné. Například na skutečnost, že se nyní občas setkáme i s chybnými verzemi upravených programů upozornil článek J. Pilného*

* / Pilný, J.: Nové výpisy programů pracujících v systému TURBO 2000. Zpravodaj ATARI KLUBU Praha 1987, 5, s. 30-37

Na rozdíl od autora tohoto článku si sice myslím, že pro začínajícího programátora má velký smysl pokoušet se i o vlastní úpravy již někým "zturbovaných" programů, protože jedině tak si mohou tito začátečníci v praxi ověřit svoji odbornou úroveň. Samozřejmě je nesporné, že vadné nebo neověřené programy by se dále (mezi další členy) neměly šířit. Háček je ovšem v tom, že chyba v upraveném programu se někdy projeví až hluboko ve hře, kam se často sám upravitel nepropracuje.

Proto byl také softwarovou skupinou výboru klubu založen archiv programů jak ve standardní (firemní) verzi, tak i v TURBO 2000. V archivu jsou ukládány jen ověřené verze.

Chci však podotknout, že za nesprávně upravené je nutné považovat i ty programy, které není možné nahrát jediným správným, tj. originálním loaderem TURBO 2000 vytvořeným J. Richterem. Autor systému TURBO 2000 J. Richter pečlivě zvážil jeho umístění. Zavádění neautorizovaných "hlaviček" pak působí řadu nesnází. Předně obecně je rozšíření pouze originální loader, kterým pak uživatelé nemohou neosetřené programy nahrát. Herní programy rozdělené do dvou částí volají při navázání právě originální loader a při jeho absenci se zhroutí (např. Interkarate, Mercenary, Airline aj.). Pokud není užití originálního loaderu ve složitě nahrávaných programech možné, je lepší speciální loader umístit v programu a spustit jej až po nahrání loaderem standardním.

Bezvýznamné nejsou dále chyby v pojme-

nování programů. I v citovaném článku J. Pilného jich je celá řada. Např. Liwewile - správně Livewire, Srankenstein - správně Schreckenstein, nebo Welcome to Firefllet - správně Firefllet, kde "Welcome to" je "vítejte do hry:". Mj. jde o to, že takto nesprávně označený program se ocitne na jiném místě seznamu programů a zájemce si marně láme hlavu, o jaký že program se to jedná. Podobně Sticks Gold - což je vlastně správně Captain S. G., a jiné. Koluje jakýsi "Renzirkus", což je ve skutečnosti Richard Pety's Talladega - Cosmi Curb.

K soupisu programů uvedených v článku J. Pilného mám ještě jednu podstatnou poznámku: Věřím, že oním "autorem a kol.", který hry skutečně upravil, je miněn autor systému Jiří Richter. Bylo by totiž do nebe volající, kdyby snad byl Jiří Richter zahrnut do označení "a kol.". Existuje totiž řada důkazů, že samotným Jiřím Richterem bylo upraveno dávno před vydáním článku J. Pilného a nakonec před vydáním přílohy II Zpravodaje AK několik desítek programů, které figurují v seznamu cit. článku. Totéž se týká jím zpracovaného loaderu pro snadnou úpravu "vykřičníkových" programů. Jiří Richter zůstal vždy poslední autoritou pro úpravu programů, na nichž si vylámali zuby i zkušení borci.

V souvislosti se zmíněným zpřístupněním většiny programů v TURBO-verzi se podle mne ještě s větší naléhavostí ozvala poptávka po návodech jak herních, tak užitkových pro-

gramů. A zde platí totéž, co bylo řečeno o programech samotných. Vedle řady kvalitních překladů originálních manuálů se objevily v poslední době i návody "náhradní", odvozené z uživateli zkušenosti. Ty někdy pouze uvádějí konečné řešení hry. To považuji za stejně nesmyslné, jako by byl návod k šachům, který by vedl tah po tahu až k matu. Některé z takových návodů jsou dokonce zcela zcestné a dokumentují jen nepochopení smyslu hry. Tak např. kolují tři velmi rozdílné verze návodu hry Chimera, nebo přehled vstupních kódů programu F 15 S.E. s šesti chybami, nefunkující povely pro hru Super Huey atd. Obtížné je i používání originálních návodů, pokud byly přeloženy někým, kdo nezná program z praxe, neovládá speciální počítačovou terminologii atd.

Obrácím se proto na členy ATARI KLUBU s následující výzvou:

Pokud vlastníte originální návody programů, které sami nemáte možnost přeložit, dejte je k dispozici softwarové skupině výboru klubu, která se postará o kvalitní překlad a zpřístupnění pro všechny. Pokud si pak návody sami rozepisujete, uvádějte vždy co nejpřesněji pramen, ze kterého popisujete, jinými slovy jde-li o přepis originálu překladu nebo o návod odvozený. Zabráňte tím řadě nedorozumění a zamezte roztrpčení těch, kterým se dostane do ruky váš opis.



Loader TURBO 2000, COPY 2000 a TURBO BASIC — XL 2.0 (TB)

Jiří Richter, Praha

Rovněž po uzávěrce jsme obdrželi návody k užívání programů nezbytných pro provoz systému TURBO 2000. Na četná přání členů klubu je proto aktuálně zveřejňujeme ve Zpravodaji AK.

Současně sdělujeme, že J. Richter dořešil nový systém zrychleného přenosu dat, který pracuje variabilně až do rychlosti přenosu větší jak 5000 Bd. Systém je slučitelný s původním systémem TURBO 2000. Od poloviny února t.r. je v klubu k dispozici pro zájemce kazeta s tímto programem, kterou si každý může zakoupit.

Poznámky k systému TURBO 2000

Pro nahrání programu v systému TURBO 2000 a jeho spuštění je nutný zaváděcí program **LOADER TURBO 2000** ve strojovém kódu.

Nahrávání Loadru:

1. Počítač ve stavu READY.
2. **BYE: (RETURN) : (START) + (OPTION) : (RESET)** — zazní tón.
3. Do magnetofonu vložit kazetu s programem Loader. — **(PLAY) : (RETURN)**
4. Po nahrání se objeví v nejvyšším řádku obrazovky nápis **TURBO 2000**.

Nahrávání programu upravených do TURBO 2000:

1. Pásku převinout několik málo otáček počítadla před hledaný program — **(RETURN) + (PLAY)**.
2. Počítač "hledá" hlavičku prvního programu od místa spuštění pásky (obrazovka je bez obrazu).
3. Po nalezení hlavičky se objeví širší vodorovné pruhy a hlavička se nahrává. Po nahrání hlavičky se zastaví chod magnetofonu, na obrazovce se objeví název vyhledaného programu a zazní tón.
4. Volba:
 - a) nahrát tento program — **(RETURN)**
 - b) nahrát jiný program — **(RESET)**, ohlásí se **TURBO 2000**, pokračovat ad 1, resp. 2.

Kopírovací program COPY 2000

kopíruje programy v systému TURBO 2000. Nahrávání tohoto programu ve strojovém

kódu je stejné, jako nahrávání programu **LOADER**. Postup kopírování:

1. Nahrát kopírovací program do počítače stejným postupem jako u programu **LOADER**. Po vypsání hlavičky vyhledaného programu možnost volby jako ad 4 u **LOADERu**, tj. **(RETURN)** nebo **(RESET)**.
2. Po nahrání programu do počítače zazní 2krát tón — počítač je připraven k přehrání programu.
3. Založit čistou kazetu do magnetofonu.
4. Volba:
 - a) **SAVE** na kazetu — **(RETURN)** lze libovolně -krát opakovat,
 - b) kopírování dalšího programu — **(RESET)**, tj. návrat ad 1).

TURBO BASIC — XL 2.0 (TB)

Jedná se o známý program **TURBO BASIC**, který byl rozšířen o systém nahrávání v **TURBO 2000**.

Způsob nahrávání TB:

BYE: (RETURN):(START)+(OPTION):(RESET)
Nový příkaz **DOS** (jiný význam příkazu **DOS**). Po provedení příkazu **DOS (DOS-(RETURN))** se na obrazovce vypíše menu programu **TURBO BASIC XL 2.0**.

LOAD SAVE QUIT VERIFY
Vysvětlení jednotlivých příkazů:

1. **LOAD** — nahrání programu do počítače
 2. **SAVE** — nahrání programu na kazetu
 3. **QUIT** — návrat do TB
 4. **VERIFY** — kontrola správnosti nahrávky
- ad 1) Stisknout **L** (1krát tón) — **(RETURN)** +

(PLAY). Počítač hledá hlavičku BASIC programu. Po nalezení hlavičky (na kazetě musí být program nahráný systémem TURBO příkazem SAVE v režimu DOS) se zastaví kazetový magnetofon. Nyní je možnost dvojí volby:

- a) nahrát program,
 - b) nahrát jiný program — 2krát (BREAK) a postup opakovat.
- ad 2) Stisknout S. Vypíše se PROGRAMM-NAME:. Napíšeme název programu (max. délka 10 znaků). Je-li první znak x, tak při LOAD se po nahrání sám spustí.

Po nahrání 1 kopie je možnost volby:

- a) další kopie (RETURN)
 - b) konec SAVE (BREAK).
- ad 4) Stisknout V. Přetočit pásku na začátek

programu — (RETURN)+(PLAY).

Po nahrání hlavičky — (RETURN).

Pokud je program v pořádku, vypíše se O.K., v opačném případě se objeví BOOT ERROR — znovu SAVE.

VERIFY je možné provést pouze ihned po SAVE.

Nové zařízení "T:Název souboru"

Univerzální zařízení — způsob použití obdobný jako u zařízení "D:" (pouze ze T musí být jediné !).

Příklady: LIST"T:Program.LST"; ENTER"T:x.LST" nebo "T:" nebo "T:Prx.x"; SAVE"T:Program.BAS"; LOAD"T:x.x"; RUN"T:Program". Další možnosti jako zařízení C: a D: — viz uživatelská příručka.

... a Jedna na Konec

HELE, TEN UŽ JE JAKO TY, ŘÍKÁ,
ABYCH MU DAL POKOJ, ŽE HO
BOLÍ HLAVA...



Zpravodaj ATARI KLUBU Praha — ročník I (1987) — seznam publikovaných článků

Ze života Svazarmu

ATARI KLUB bude zajišťovat srazy caravanů
red 4/87, s. 4—5

ATARI KLUB na ZENITU
J. Hlaváček 3/87, s. 1—2

AURORA hostila ATARI
HI 5/87, s. 2

Den Svazarmu v PKOJF
R. Hána 4/87, s. 4

Mezinárodní strojírenský veletrh v Brně (a ATARI)
HI 5/87, s. 2

Na mezinárodní soutěži
Michal Novák 5/87, s. 1

Na Slavnosti míru a Československo-sovětského přátelství
J. Hlaváček 3/87, s. 3

Pracovní návštěvy na ÚV Svazarmu
HI 2/87, s. 1—2

Vážení kolegové a kolegyně
výbor 1/87, s. 1—4

V den 70. výročí VŘSR
HI 6/87, s. 1—2

4. seminář k výpočetní technice v Brně
J. Hlaváček 4/87, s. 1—4

Zprávy výboru

Blahopřání do roku 1988
red 6/87, s. 8

Činnost klubu po prázdninách
výbor 4/87, s. 6

Dohoda s ATARI KLUBEM Timače
red 3/87, s. 4

Dopis výboru 487. ZO Svazarmu — ATARI KLUBU v Praze 4 všem členům ATARI KLUBU
výbor a redakční rada 2/87, s. 3—6

Evidenční číslo člena
red 3/87, s. 7

Fotografujete?
výbor 5/87, s. 6

Jak psát příspěvky do Zpravodaje AK
J. Hlaváček 6/87, s. 8—9

Konkurs
výbor 4/87, s. 7

KONTAKT
red 3/87, s. 8

KONTAKT — oprava
red 6/87, s. 9

Koupili jsme ATARI
výbor 4/87, s. 6

Kursy ve 2. pololetí 1987
red 3/87, s. 7

K vytváření pracovních kroužků 487. ZO Svazarmu — ATARI KLUBU
2/87, s. 6—8

Minianketa Zpravodaje ATARI KLUBU Praha
redakční rada 6/87, s. 10

Napsáno o ATARI
red 4/87, s. 6
5/87, s. 5

Nekvalitní obrazový signál počítačů ATARI 130 XE
red 1/87, s. 7

Objednávání literatury
red 4/87, s. 8

O činnosti kroužku v Hradci Králové
HI 5/87, s. 3

Odpověď redakce časopisu VTM
red 5/87, s. 3—4

Otevíráme „stop-stav“
red 3/87, s. 7

Placení členských příspěvků na rok 1988 pro členy 487. ZO Svazarmu — ATARI KLUBU v Praze 4 a pro hostující členy
výbor 5/87, s. 4—5

Pochvala a kritika
HI 5/87, s. 6

Poučení pro mimopražské členy
red 3/87, s. 6

Pozor na podvodníky!
HI 2/87, s. 9

Připomínáme
red 3/87, s. 4

Připravujeme dohodu s Bratislavou a Brnem
red 3/87, s. 6

Redakce ATARI KLUBU připravuje
red 3/87, s. 7

3/87, s. 7
5/87, s. 6
6/87, s. 9

Setkání se zástupcem ATARI KLUBU z Olomouce
red 3/87, s. 4—5

Určeno čtenářům Zpravodaje ATARI KLUBU
J. Hlaváček 1/87, s. 4—6

Uživatelé ATARI ST (PC?), ozvěte se
red 5/87, s. 5

Vytváří se pracovní kroužek ATARI KLUBU v Praze 8 - Čimicích

<i>red</i>	2/87, s. 8
Výzva k upřesnění edičního plánu ATARI KLUBU	
<i>HI</i>	2/87, s. 9
Vznikají další pracovní kroužky ATARI KLUBU	
<i>red</i>	3/87, s. 6
Základní údaje o naší ZO po jednom roce činnosti	
<i>výbor</i>	6/87, s. 3—7

Pro začátečníky

Aritmetické funkce na počítačích ATARI	
<i>V. Friedrich</i>	5/87, s. 11—12
BASIC nebo PASCAL	
<i>J. Rada</i>	6/87, s. 13—14
CASDUP 2.0 — kopírovací program kazeta - kazeta	
<i>V. Friedrich</i>	2/87, s. 11—13
Co je co?	2/87, s. 10—11
<i>J. Bok</i>	3/87, s. 9—10
Generování náhodných čísel	
<i>OLVIS</i>	3/87, s. 12
Komunikační jazyk "ENGLISH"	
<i>V. Friedrich</i>	5/87, s. 12—14
Kopírování programů z kazet — informace pro začátečníky	
<i>J. Jícha</i>	1/87, s. 8—12
Označování souborů na disketách	
<i>fis</i>	4/87, s. 13
Pole řetězců v jazyce ATARI BASIC	
<i>fis</i>	4/87, s. 9—10
Práce s řetězci v ATARI BASICU	
<i>J. Jícha</i>	4/87, s. 11—12
Právní ochrana výsledků tvůrčí činnosti v oblasti výpočetní techniky	
<i>E. Jenerál</i>	4/87, s. 14—16
	5/87, s. 7—10
	6/87, s. 11—13

Úvod k rubrice	
<i>red</i>	1/87, s. 7
Verifikace programů na kazetě	
<i>fis</i>	3/87, s. 11—12

Tipy a triky

ATARI BASIC — jeho verze a chyby

<i>P. J.</i>	1/87, s. 15
--------------	-------------

AUTORUN

<i>fis</i>	3/87, s. 16—17
Blikající kurzor	
<i>fis</i>	2/87, s. 19—20
Důležité podprogramy	
<i>fis</i>	3/87, s. 16
Jemné rolování v grafice 0	
<i>fis</i>	3/87, s. 16
Ochrana programu	
<i>fis</i>	2/87, s. 19
Operace s kanálem #0	
<i>fis</i>	3/87, s. 17
Ovládání disketové jednotky	
<i>fis</i>	4/87, s. 18
Pomůcky pro programování (1. část)	
<i>P. Jandík</i>	6/87, s. 15—16
Přesun znakové sady	
<i>fis</i>	2/87, s. 20
Přímý přístup do souboru DATA	
<i>fis</i>	3/87, s. 15
Rezervace paměti	
<i>fis</i>	2/87, s. 21
Rotace barev	
<i>fis</i>	2/87, s. 20
Rozšíření editoru	
<i>fis</i>	3/87, s. 15—16
Rychleji s ATARI 800 XL (130 XE)	
<i>Hej</i>	1/87, s. 17
Úvod k rubrice	
<i>red</i>	1/87, s. 15
Verifikace kazetové nahrávky na počítači ATARI	
<i>Hej</i>	1/87, s. 17—18
Vlastníte rovněž ATARI BASIC revize B?	
<i>Hej</i>	3/87, s. 17—18
Využití logických funkcí u jazyku TURBO BASIC XL 1.5	
<i>fis</i>	4/87, s. 19—20
3x disketová jednotka	
<i>fis</i>	5/87, s. 15—16

Otázky a odpovědi

Časový limit pro vložené hodnoty do počítače	
<i>vr</i>	3/87, s. 14
Jak lze vytvořit delší (než 3) programové řádky?	
<i>rd</i>	4/87, s. 17
Jak se dají využívat ovládací vstupy u počítačů ATARI 800 XL a 130 XE? Jakých typů	

ovladačů je možno použít?	
<i>rd</i>	2/87, s. 14—18
Jak se tvoří pohyblivé znaky?	
<i>rd</i>	4/87, s. 17
Jak vypočítat tangentu?	
<i>rd</i>	2/87, s. 14
Jakým způsobem lze zasáhnout do znakového souboru, který je zabudován v počítači?	
<i>vr</i>	5/87, s. 17—18
K opatření si literatury	
<i>rd</i>	1/87, s. 13
K použití běžného magnetofonu	
<i>rd</i>	1/87, s. 13
K testování paměti RAM pomocí samočinného testu SELF TEST	
<i>rd</i>	1/87, s. 14
Lze využít ATARI k programování hudby (MIDI) <i>vr</i>	3/87, s. 13—14
Proč nelze použít obyčejný kazetový magnetofon	
<i>rd</i>	1/87, s. 14
Úvod k rubrice	
<i>red</i>	1/87, s. 12
Změna nahrávací rychlosti a maximální rychlost nahrávání z magnetofonu	
<i>rd</i>	4/87, s. 17

Uživatelské programy

ATARI ARTIS — výstup na tiskárnu	
<i>JABS</i>	4/87, s. 24
DEBUG MONITOR — návod k programu	
<i>P. Vacek</i>	5/87, s. 23—29
DEBUG +	
<i>P. Jandík</i>	6/87, s. 21—25
Funkční test přídavné paměti RAM u ATARI 130 XE	
<i>O. Višek</i>	2/87, s. 22—23
Jak si uchovávat obrázky	
<i>fis</i>	4/87, s. 21—23
K.O.S. — Kazetový operační systém	
<i>J. Šetina</i>	6/87, s. 28—32
Nové výpisy programů pracujících v systému TURBO 2000	
<i>J. Pilný ml.</i>	5/87, s. 30—37
Popis hry SOLO FLIGHT	
<i>G. M.</i>	1/87, s. 19—22
Snaha o „normalizaci“ zápisů na kazety	
<i>HI</i>	1/87, s. 23—24
SORT v BASICu	
<i>lar</i>	5/87, s. 21—23

RAMDISK pro ATARI 800 XL	
<i>P. Strnad</i>	4/87, s. 24—25
ROCKFORD proslavil ATARI	
<i>fis</i>	5/87, s. 37—38
Turbo Basic Interpret pro ATARI 800 XL a 130 XE	
<i>J. Moučka</i>	3/87, s. 19—24, 32
Univerzální seznam — popis programu	
<i>J. Jirkovec</i>	6/87, s. 25—28
Úvod k rubrice	
<i>red</i>	1/87, s. 18—19
Zelená uživatelským programům	
<i>J. Hlaváček</i>	6/87, s. 33—35
Znakový editor ULTRAFONT	
<i>M. Vavřda</i>	5/87, s. 18—20

Koutek techniky

Aby oči nebolely	
<i>Hej</i>	3/87, s. 30
ATARI — SHILJALIS — VEF	
<i>Kamil Tesla</i>	4/87, s. 32—33
ATARI 600 XL s pamětí 64 kB	
<i>V. Adamovský</i>	4/87, s. 29—30
ATARI 800 XL s pamětí 320 kB	
<i>JABS</i>	6/87, s. 36—41
Centronics printer interface (1. část)	
<i>V. Fajta</i>	6/87, s. 46—49
Interfejs ke spojení tiskárny CONSUL 2111-03 s počítačem ATARI 800 XL (130 XE)	
<i>F. Vondruš</i>	6/87, s. 42—45
Interfejs pro ATARI 800 XL (130 XE) (pro připojení magnetofonu)	
<i>M. Bayer</i>	3/87, s. 25—28
Jednoduchý ovladač (joystick) pro ATARI 800 XL (130 XE), typ PR 1	
<i>P. R.</i>	3/87, s. 29—30
Návod na výrobu jednoduchého ovladače (joysticku) typ RH 1 a konektoru	
<i>R. Hána</i>	2/87, s. 28—30
Nová verze počítačů ATARI 800 XL	
<i>M. Machulda</i>	2/87, s. 24
Přehrávání programů bez počítače	
<i>O. Višek, K. Vlach</i>	5/87, s. 40—43
Připojení dálnopisu SIEMENS 100 k počítači ATARI 800 XL	
<i>S. Štulpa</i>	5/87, s. 43—47
Připojení magnetofonu a systém TURBO 2000 (1. část)	
<i>P. Havlíček</i>	6/87, s. 49—50

Stavíte rovněž ovladač s potenciometrem (PADDLE)?	
J. Bok	2/87, s. 25—28
Světelné pero	
JABS	4/87, s. 26—29
Úprava černobílého TV přijímače na monitor pro ATARI 800 XL, 130 XE	
ft dle návodu T. Naxery	3/87, s. 31—32
Úprava magnetofonu ATARI XC-12 pro snadnější vyhledávání začátků programů	
M. Machulda	2/87, s. 24—25
Úvod k rubrice	
ft	1/87, s. 24
Výběr součástek a nastavení interfejsu pro magnetofony	
V. Havelka	5/87, s. 39—40
Výroba konektoru pro ovladač (joystick)	
J. Štěrba	3/87, s. 28—29
Výzva konstruktérům	
red	6/87, s. 35
Zapojení ATARI 800 XL a 130 XE na video vstup TV přijímače	
M. Machulda	4/87, s. 31

Technické novinky

ATARI 800 XE a XL 12	
M. Machulda	6/87, s. 52
CCD TV kamera pro osvětlení od 2 luxů	
Bos	5/87, s. 48
Co vyplynulo z rozhovoru redaktora BAJTKu s Jackem Tramiellem	
J. Hlaváček, J. Tomášek	2/87, s. 31—32
DAT kazety od firem Fuji a Taiyo Yuden	
Bos	4/87, s. 34
Firma Atari na výstavě CeBIT	
L. Gál	5/87, s. 48
Kasparov využívá ATARI	
Bos	1/87, s. 24
Nové počítače ATARI — 260, 520 a 1040 ST a ATARI PC	
Bos	3/87, s. 34
Novinky osmibitové řady ATARI	
P. Jandík	6/87, s. 51—52
Popis ATARI 1040 STF	
Bos	3/87, s. 33
Premiera DAT magnetofonů Hitachi a Toshiba	
Bos	4/87, s. 34
Prodejnost ATARI ST v 1. pololetí 1986 v NSR	
Bos	3/87, s. 33

Scientific Computer Systems (SCS) — počítače kompatibilní se superpočítačem CRAY	
Bos	6/87, s. 52—53
SHARP PC 1600 — popis	
Bos	3/87, s. 35
Standard pro optický záznam	
red	6/87, s. 53—54
Still Video System — Canon	
Bos	5/87, s. 48—49

Listárna

ATOM bude psát o ATARI	
red	5/87, s. 50
Bez nadpisu k zamyšlení	
L. Horák, F. V.	6/87, s. 57—58
K zajištění kopírování programů a půjčování odborné literatury	
výbor a redakční rada	2/87, s. 33—34
PF 1988	6/87, s. 59
Po uzavěrcce	3/87, s. 35
Upřesnění KONTAKTu	
lar	5/87, s. 51
Zeptali jsme se za vás . . .	
. . . na GR PZO TUZEX	
J. Hlaváček	4/87, s. 35
. . . v k.p. ARITMA Praha	
J. Hlaváček	5/87, s. 51
. . . ve v.d. STYL	
J. Hlaváček	6/87, s. 54—56
. . . v GR OPZ	
Bos	6/87, s. 56
Železničáři hledají spolupráci	
red	5/87, s. 51
Oprava chyb ve Zpravodaji AK 1/87	
	2/87, s. 35
Oprava chyb ve Zpravodaji AK 2/87	
	3/87, s. 35
Oprava chyb ve Zpravodaji AK 3/87	
	4/87, s. 33

Přílohy Zpravodaje ATARI KLUBU Praha vydané v roce 1987

Bayer, M. a kol.: ATARI 800 XL — důležité adresy systému a jejich použití pro programování v BASICu	příloha I/87
Richter, J.: TURBO 2000 — systém zrychleného přenosu dat z magnetofonů ATARI	příloha II/87

Publikované zo súhlasom - vid' Prohlášení představitelů AK Praha.