

MIKRO

1989

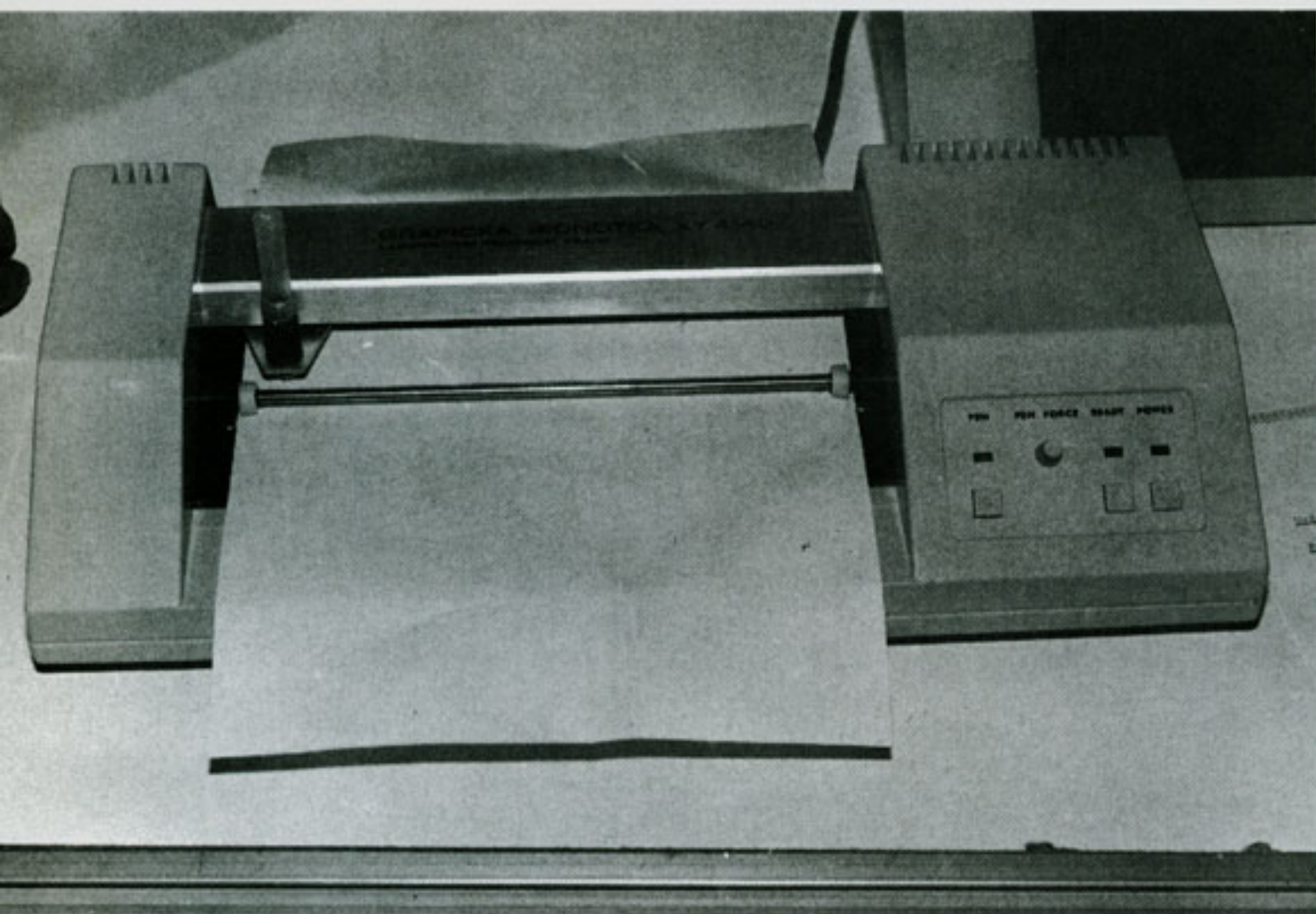
1

BAŮZE

technický
zpravodaj
svazarmu
pro zájemce o
mikropočítače

Cena 12 Kčs





Z výstavy
**Elektronizace
 a automatizace '88**
 konané ve dnech
 1.11. - 9. 11. 1989
 na brněnském výstavišti





OBSAH

The Russians Are Coming	1
Femina quae programmat	2
Dřu, dřeš, dřeme... Céčko	4
Filozof.aspekty stroj.myšlení (4) ..	7
Počítačová scifi (5)	10
Závada na ZX Spectru 128K	12
LECROM pro ZX Spectrum	13
Kompatibilita Didaktik Gama/ZX Sp...	16
Rozšíření paměti ZX Spectra (2)	17
Turbo interface pro ZX Spectrum	21
Měření napětí	24
Interface C211-03 pro Atari 800/130	25
Programuj, ale zůstaň člověkem	27
Přehled laserových a spec.tiskáren .	29
Novinky z domova	30
Informace VTI	31
Nabídka Mikrobáze	32

Technický zpravodaj Svazarmu pro zájemce o mikropočítače. Vydává 602.ZO Svazarmu ve spolupráci s redakcí časopisu Amatérské radio. Povoleno ÚVTEI pod evidenčním číslem 87 007. Zodpovědný redaktor ing. Jan Klabal. Sestavil Ladislav Zajíček. Grafická úprava Yveta Dunděrová. Sekretářka redakce Božena Divišová. Redakční rada: ing. Petr Horák, ing. Jan Klabal, ing. Petr Kratochvíl, Josef Kroupa, Daniel Meca, ing. Alois Myslík, ing. Josef Truxa, Ladislav Zajíček. Za původnost a správnost příspěvků ručí autoři. Ročně vyjde 10 čísel. Cena výtisku 12 Kčs podle ČCÚ a SCÚ č. 1030/202/86. Roční předplatné 120 Kčs. Objednávky přijímá a zpravodaj rozšiřuje 602.ZO Svazarmu, Wintrova 8, 160 41 Praha 6.



602.ZO

&

RADIO

THE RUSSIANS ARE COMING

Pod titulkem "Rusové nastupují" zaměřuje šéfredaktor amerického měsíčníku BYTE 10/88 svůj pohled na původní sovětský software, jaký chtějí obchodní organizace SSSR nabízet na západním trhu. Podtitulek "A vypadá to, že pro obchod mají velmi zajímavý software" mluví dostatečně jasně.

Koncem léta 88 přijeli do USA zástupci Akademie věd SSSR, vědecko-výzkumného institutu, Ústavu aplikované fyziky a matematiky, Sibiřského centra výpočetní techniky a Komise pro computerovou technologii. Všechny předváděné programy byly upraveny i pro automatickou demonstraci svých možností. Texty byly jak ruské, tak anglické. Z výčtu aplikací plyne, že v kolekci nechybělo nic, čím se dnes běžně prezentují zahraniční firmy (včetně expertních systémů). Programy jsou vytvořeny převážně v jazyku, který je hybridem Small Talku a Prologu. Vadim Kotov ze Sibiřského centra uvedl, že všechny programy jsou v podstatě moduly, které lze libovolně sestavovat. Všechny také spolu dokáží vzájemně komunikovat. Takovou výzbroj označují slovy "software factory".

Šéfredaktor chválí způsob prezentace programů jako velice jasný, zřetelný i zaměřený na komerční efekt. BYTE se dokonce rozhodl demonstrační programy zařadit do své mohutné databanky BIX. Tak si je kdokoli (nejen v USA) může vyvolat a prohlédnout. Což může prospět nejen obchodu, ale i dát podnět rozvoji další tvorby.

Sověti dále předvedli ruský slovní procesor s obrovským thesaurusem, na který je Vadim Kotov zvláště hrdý. Při tvarosloví, jakým oplývají slovanské jazyky, není divu.

Jeden ze sovětských zástupců uvedl, že v SSSR je asi 200 000 počítačů typu PC XT/AT. Rádi by přešli na systémy s 80386, jenže strop embarga končí u procesorů 80286 do 12,5 MHz. Sovětští technici vyvinuli 32-bitový počítač s procesorem, který má obdobnou architekturu jako RISC. Pracuje i v paralelním režimu.

Nejoblíbenějším programovacím jazykem sovětských profesionálů je Céčko (plus UNIX), následované Pascallem a Modulou-2.

V závěru statí je uvedeno, že BYTE se do sovětských institucí dostává podle pravidla jeden do jedné. Tak se na každý výtisk tvoří dlouhá fronta zájemců. Kotov v žertu uvedl, že jeho kolegové měsíčníku říkají "PlayBYTE, protože je tam tolik zajímavého ke koukání".

Zatím se nezdá, že by BYTE v nejbližších letech uvedl článek s názvem "The Czechoslovaks Are Coming". Ne že by u nás nebyli dobří programátoři. Jenže jsou to osamělí běžci, kteří se potýkají s nedostatečným zázemím a propadlem byrokracie, která si nejradši pochutnává na tom, co voní životností. Tak se z jejího stínu nemohou vynořit ti, kteří mají nadání, ale žádnou rozumnou možnost jeho uplatnění. Když má něco dávat sladké plody, musí pro to mít patřičné životní prostředí. Pak svou energii může vydávat na tvorbu, nikoli jen na holé přežívání.

Jde o vytvoření podmínek, v nichž by se potřebné struktury mohly tvořit samy. Je to přestup z kvalitativně nižších do vyšších strukturálních úrovní. Hrozně jednoduchý, přírodou odnepaměti ověřený princip emergence. Jenže dosud kolem nás stále ještě operují administrátoři, pro které je osobně atraktivní tomuto principu čelit. Svými oblíbenými prostředky si zajišťují důkazy oprávněnosti své existence pro neexistenci druhých. Od kulturních komisí, které rozhodují, co se druhým smí líbit, až po reaktivátory předpisů pro mrazírenskou úpravu subtilní potence novodobých komunikačních nástrojů.

Pro důkaz oprávněnosti tohoto tvrzení není nutno číst o úspěších sovětského softwaru v anglosaských časopisech. Stačí prolistovat stránky polského nebo maďarského tisku. I když tu v potřebném směru bylo již něco málo učiněno (např. snížení cla na dovoz počítačů), nezbytné základní stimuly vzniku seberegulačních struktur s horizontální komunikací jsou nadále za hranicemi všedních dnů. Každým takovým dnem se naše ztráty v současném synergickém světě zvyšují, protože v něm chybí náš vklad vzájemné návaznosti. A stačí tak málo - dát prostor uplatnění důvtipu a umění lidí, kteří by tady u počítače dokázali netušené. Snad proto, že vím, že takoví mezi námi jsou, zůstávám optimistou. Jen mám obavu, aby se jim vlastní zkušenost osamělého běžce nestala stereotypem, z něhož se už nedokáží vymanit.



HOVORY

O PROGRAMOVÁNÍ FEMINA QUAE PROGRAMMAT

...neboli žena programující. Pozvání Inky Jelínkové (ročník 1956, programátorka Inogry) k následujícímu rozhovoru bylo inspirováno obsahem interview Trojúhelník žena-muž-počítač (viz Mikro-báze č.4/88). Pro připomenutí - PHDr.B.Blažek v něm ženy řadí spíše do polohy "měkkých programátorek", muže k typům "tvrdým". A to netoliko u počítačů, ale v životě obecně. Než jsem začal Ince klást otázky, dal jsem jí rozhovor s B.Blažkem přečíst. Její první reakce:

"Nevím, ale pokud jde o rozvrhování práce předem, myslím, že právě v tom ženy patří spíš ke tvrdému typu. Alespoň pokud jde o ženy, které znám já. Souhlasím ale s tím, že jsou mnohem intuitivnější než muži, což je zase oblast měkkosti."

"Ženy jakých profesí máš na mysli?"

"Hlavně matematicky."

"Muž, kterému je žena představena jako matematicka, obvykle poklesne v kolenu. Jaký názor na ně máš ty?"

"To poklesnutí je zapříčiněno vžitým předsudkem. Znam jednu, která má už třetího potomka, jiná je skvělá společnice... Prostě jsou to úplně normální ženský. Když porovnám jejich i svůj přístup k řešení problémů, musím znova podtrhnout, že je plánovitější než u mužů. Ti se do věci vrhají víc po hlavě. Programátor, se kterým pracuji, začne třeba definicemi proměnných a jede pořád pryč, až napíše obrovské monstrum. Žádné blokové rozvržení předem. Myslím, že je to dáno i tím, že žena v běžném životě musí souběžně zvládat spoustu věcí najednou. Kdyby si je předem nepromyslela, nezvládla by je. Její přístup je proto praktičtější."

"Kdybych se to pokusil převést do kuchyňské symboliky, znamená to, že žena dokáže vařit ve více hrncích najednou, až je z toho požadovaná krmě, zatímco muž...?"

"Ten by to dokázal taky. Jenže v tom nemá onu potřebnou praxi. A zvykově ho taky k tomu nic nenutí. Přestože neporovnatelně víc žen umí vařit, kuchařskými mistry se stávají muži. Muž bere všechno víc jako hru a do hry se dokáže ponořit. Žena víc kalkuluje předem."

"Brání tedy ona pověstná ženská vypočítavost ženě v onom hlubokém ponoru do hry?"

"Já myslím, že neexistuje programátor, pro kterého by programování nebylo aspoň trochu hrou. Pro mě je to každopádně víc hra. A při hledání chyb i detektivka. U mne je to dáno i tím, že

dělám počítačovou grafiku - v ní je výsledný produkt sám dost hravý."

"Jak ses dostala k programování?"

"Tak, že jsem se nedostala na školu, kam jsem chtěla jít. Nechala jsem se umluvit a v roce 1976 šla na ČVUT, kde nově otevírali obor programování. Z humanitní větve gymplu jsem rovnýma nohama skočila úplně někam jinam. Pro výuku programování tenkrát ještě nebyla žádná skripta ani jiná literatura. Všechno jsem chytala z přednášek profesorů, pro něž samotné to byla novinka. Pro diplomovou práci jsem měla k dispozici jediný podklad-firemní katalog INTEL. Během studia jsem se prakticky vůbec nedostala k počítači. Programy jsme psali v místnosti s dvaceti dálnopisy. Co jsi naťukal, to jsi dal do přihrádky. Za několik dní se ti to odněkud vrátilo s tvými chybami... a tak dokola. Počítač jsem poprvé viděla pod sklem na nějaké výstavě. Díky tomu nevím skoro nic o hardwaru."

Po škole jsem se do přímého kontaktu s počítačem dostala ve VÚMSu. Tam jsem si vůbec poprvé mohla s počítačem povídat opravdu "z očí do očí". Byly to ty velké stroje řady EC. Co je ale na nich o nervy, to je ladění - nejsou vybaveny žádným kloudným debuggerem. Na nich se ladí přes výpisy z tiskárny. Příkazy ke kontrolním tiskům si musíš dát přímo do programu, pak ho zase slinkovat, přeložit a čekat na obrátku. Ta je podle operačního systému, a podle toho, jak je to výpočetním střediskem zařízeno, různě dlouhá. Někdy na to, aby ses dozvěděl, co ti program dělá v nějaké buňce paměti, čekáš třeba celý den."

Pak jsem se dostala ke slovenským "emvéeskám". Při práci s nimi jsme museli zavést deník, do kterého jsem zapisovali jejich poruchovost, abychom vykazali, proč neplníme plán. Když zrovna fungovaly, dělalo se na nich dobře, protože jim nechyběly debuggery. Ale měly jen assembler a Basic. To byl pro mě krok zpátky, protože předtím jsem dělala ve Fortranu a PL/1."

"A tvoje další jazyková výzbroj?"

"Na fakultě jsme se učili ještě Algol a Lisp. V práci teď už konečně dělám na pécečku, nejradši v Pascalu."

"Jaké bylo tvé první setkání s domácím počítačem?"

"Strhující. Oba doma programátoři a na stole jedno ZX Spectrum! "Zápasili" jsme o to, kdo si k němu sedne. Když z nás opadlo ono pověstné herní

šilenství, začali jsme sami něco zkoušet. Často jsme u počítače ztratili pojem o čase. Člověku se zdálo, že u něj sedí hodinku... a za okny už svítalo. Dosavadní záliby - kultura, turistika - šly stranou."

"Takže počítač změnil rytmus vašeho denního života?"

"Určitě ano. Ale já bych to nekvalifikovala jako nějaký "zvláštní stav". Prostě je to koníček. Když třeba maluju, což je moje další hobby, taky jde všechno stranou. Teď momentálně doma počítač nemáme. Ne že bychom nějaký nechtěli, ale nevidíme žádný rozumný způsob, jak si nějaký dobrý opatřit. Ale to není jen náš případ."

"Jak tvoříš programy v zaměstnání?"

"Oba děláme spolu na jednom programu. Máme výhodu v tom, že nejsme časově omezováni, k počítači můžeme zasednout prakticky kdy chceme. Většinou za ním vyjíždíme kolem poledne a často u něj sedíme až do půlnoci. To je skoro, jako bychom jej měli doma."

"Co je smyslem grafických programů, které na počítači děláte?"

"Jsou pro plzeňskou Škodovku. Výsledkem má být děrná páska pro NC stroje. Podle dodaných technologických výkresů děláme matematický model obráběných součástí."

"Tak, aby se tím už mohl řídit stroj?"

"Ne. Děláme jen úplný začátek. Náš produkt jde do dalších mezifází přípravy, kde se vše přepracovává na cesty obráběcího stroje."

"V jaké podobě předáváte svůj produkt těm mezičlánkům?"

"Jako hotový celek s veškerou uživatelskou komunikací. Takže příští uživatel nemusí nad ničím nijak zvlášť koumat, jen si vybírá potřebné funkce z menu."

"To děláte ve dvou celý program?"

"Protože je dost rozsáhlý, je předem rozdělen do jednotlivých modulů a na nich průběžně dělají dvě skupiny lidí."

"Jak by podle tebe měla týmová práce na programu vypadat?"

"Nedostatkem zadání často bývá jeho přílišná obecnost. Třeba program pro tvorbu karosérií se pochopitelně bude v lecčem lišit od potřeb ševcovské dílny. Když zadavatel pořádně neví, co vlastně chce, jak to pak má vědět a řešit programátor? Myslím, že by vždy u toho měl být člověk, který bude přesně vědět, co a jak, který bude mít dopředu jasno v tom, jak postupovat. Tedy takový řídicí element. Stává se, že během tvorby programu se u něj vystřídají různí lidé, takže v něm jsou třeba duplikované moduly, které svědčí i o různých koncepcích řešení..."

"V souvislosti s tím - myslíš si, že produkty našich programátorských týmů by byly konkurence schopné na zahraničním trhu?"

"Tady hrozně záleží na podmínkách, ve kterých programátoři pracují. Třeba my jsme ještě donedávna dělali na PDP. Jenže uživatelé to chtějí na standardu PC, na který to teď předěláváme. A pak tu jde o organizaci postupu. My vytahujeme data z výkresu, který je jako výchozí materiál zcela nevhodný. Normálně by to mělo být tak, že už konstruktér bude tvořit na počítači, ze kterého bych pak dostala všechna potřebná data modelu ploch. Vlastní výkres by tu byl jen jaksi mimo chodem - pro dokumentaci, eventuálně pro jiné potřeby podniku. Když nám někde někdo maluje výkresy a my to pak z nich složitě vytahujeme, nikam to nevede, nejde tomu dát ani nějaký zobecňující postup. V takové situaci jsme opravdu bezkonkurenční."

"Vraťme se k trojúhelníku žena-muž-počítač. Čím je podle tebe - jakožto programující ženy - dáno, že jsou rodiny, kde se žena vzbouří proti počítači?"

"Mně se to fakticky těžko posuzuje. Ale když si odmyslím svou vlastní situaci, snad je to tím, že

spojení muž-počítač ji tak nějak jako "ponižuje". Když třeba muž sbíjí poličky, umí si žena představit, o co jde - tady hřebík, támhle kladívko... a navíc je to užitečné. Ale počítač je něco zcela vzdáleného, nepochopitelného. Cizí svět, který k ničemu není."

"Jak z toho ven? Prodat počítač?"

"To asi ne, to zase chudák chlap... Blažek ve svém rozhovoru mluvil o spojení přes děti. Nevím, ale z toho by se zase mohlo vyvinout spiklenectví otec-syn proti matce. Tak by už na ni byli dva. Když se na celou záležitost podívám úplně z gruntu, vidím problém v tom, že v té sféře laiků jde pořád ještě o záhadnou novinku. Až se dostatečně rozšíří, ženy chtě nechtě přijdou s počítači do přímého styku v zaměstnání i mimo ně. Až pokladni nebo písařka dostane na stůl místo mechanického stroje počítač, začne chápat, k čemu to je, že je to věc, která šetří čas, námahu a že to není nic z nějakého cizího světa."

"Není problém taky v tom, že většina dnešních typicky ženských činností není na počítač převoditelná? Třeba vaření, šití, praní opatrování dítěte... Zatímco muž se často zabývá něčím, co po využití počítače přímo volá?"

"Určitě i v tom to je. Žena skutečně věnuje valnou část svého času, myšlenek a energie činnostem, které jsou mimo mužský obzor. Kdyby muži třeba myli nádobí, zatímco žena by mohla sedět u počítače... Vlastně ať si to muži takhle experimentálně ověří sami na sobě! Třeba pak ženu víc pochopí a přijdou i na to, jak věc řešit. Chce to ale víc aktivity od muže, nesetrvávat v té jakési tajemné nadřazenosti."

"Na začátku naší rozmluvy jsi uvedla, že ženu považuješ za tvora mnohem systematictějšího než muže. Nemohu si odpustit, abych řekl, že v detailech to třeba tak je, ale..."

"...že jinak nám chybí větší nadhled? To asi ano. Žena často reaguje mnohem iracionálněji než muž. Pořád ale tvrdím, že je to dáno tím, že se musí zabývat věcmi, kterými se pro zajištění chodu rodiny někdo zabývat musí. Souhlasím s Blažkem, že podstata problému leží v historickém vývoji role ženy. Nemyslím ale, že by řešení bylo v jejím úniku od rodiny někam, kde by se ukájela všelikými hrátkami. Konkrétně jde třeba o to, že efektivita oněch rozličných ženských činností je dnes velmi nízká. Když žena potřebuje pro dítě a rodinu to či ono, nemůže si to vyvolat zmáčknutím několika knoflíků na počítači, ale musí obíhat, shánět, vystávat ve frontách... Je to miliarda únavných, nepříjemných, vyloženě netvůrčích detailů, které musí pořád řešit, cele se jim věnovat. A když pak přijde domů a vidí svého muže zcela ponořeného do počítače, konflikt je ve vzduchu. Myslím, že tu obecně jde jak o změnu společenské pozice ženy, tak o podmínky, které by ten proces mohly urychlit."

"I když to neplatí absolutně, tak v umění bývá patrný rozdíl v umělecké expresi, svědčící o diferencovaném tvůrčím, estetickém prožitku žen a mužů. Nemyslíš, že se to nějak projevuje i ve tvorbě programů?"

"Těžká otázka. Možná jsou muži v té racionální úrovni nápaditější, vynalézavější. I když si myslím, že i tady intuice může hrát svou pozitivní roli..."

"A nemůže ta intuice někdy být trochu zavádějící?"

"To jistě. Ale ve výsledku se obě cesty nakonec scházejí. Žena třeba přeskočí několik úrovní, kterými ta mužská racionalita prochází, ale někdy se musí zase vrátit a začít "skákat odjinud". Určitě tu svou roli hraje i její větší citovost. Ale asi i tady platí, že zlatá bude ona příslověčná střední cesta, ke které to měkké i tvrdé společně přispívá potřebnou měrou."

DŘU, DŘEŠ, DŘEME... CÉČKO

(1)

V Mikrobázi se zatím neobjevil jediný program napsaný v Céčku. To nasvědčuje tomu, že v zájmové činnosti si Céčko zatím nenašlo své místo. Důvodů této absence je několik. Základní - pro výuku Céčka u nás není běžně dostupná literatura. Něco prý vydala slovenská Alfa, něco Slušovice... Jenže v krámě nic není. Mezi lidmi pobíhají "ruzné preklady z anglictiny", které aspoň částečně zmenšují obecný dopad věhlasné redakční neschopnosti zdejších nakladatelství. Dalším důvodem nejspíš bude obecné uspokojení z pokrytí vlastních potřeb Basicem, assemblerem a poslední dobou i Pascalem. Řetězec pokračuje nepřístupností manuálů ke kompilátorům Céčka. Ale jak všichni víme, to pro tuzemsky ostříleného majitele počítače není žádná větší překážka ("sežeň, kde můžeš, ale hlavně sežeň").

To vše dohromady sice vytváří do jisté míry neprostupnou mlhu kolem nasazení Céčka v zájmové činnosti, nicméně neměli bychom zůstat příliš pozadu za obecnými vývojovými trendy. Když se dnes podíváte do nejrenomovanějších zahraničních časopisů, zjistíte, že Céčko má navrch. Jistou základnu pro jeho pochopení rozhodně máme. Ze všech tří výše jmenovaných jazyků má Céčko nejblíže k Pascalu. S ohledem na hejna assemblerů tvůrců se Céčko jeví jako nejvhodnější doplněk assembleru (a naopak). A přívrženci Basicu snad nepohrdnou jazykem, který jim otevírá trochu jiný pohled na programovou tvorbu.

Následující výukový seriál je do značné míry publicistickým experimentem. Budu ho psát jako ten, kdo se sám Céčko učí tak, jak je tu možné - z různých kolujících materiálů a překladů. Chci tak m.j. dosáhnout přesného obrazu skutečnosti, v jaké tu všichni jsme. I když pokládám hlavu na špalek ostrým kritikám superznalců, láká mě tento experiment ještě z jiného důvodu. Totiž - když vám někdo vysvětluje něco, co už sám dokonale zná, jeho výkladu obvykle chybí porozumění pro trampoty začátečníka, protože na ty své už dávno zapomněl. Já tyhle trampoty budu průběžně prožívat s vámi. V tom vidím zásadní klad experimentu. V neposlední řadě chci o tom, o čem se u nás obvykle píše v křeči odlidštěné pseudovědeckosti, psát co nejlidštěji. Zápor je v tom, že v tuto chvíli-

stejně jako vy - Céčko neovládám. Cílem celého seriálu pochopitelně bude položení tohoto záporu na lopatky. Ještě se domluvíme na tom, že při naší společné výuce budu vždy o nějaký krok napřed, abych se vyhnul kázání bludů a svým náskokem zajistil i určitý nadhled nad vývojem celého seriálu.

Jaké zdroje a nástroje mám pro psaní seriálu připraveny? Americkou knihu The C Programming Language otců jazyka B. Kerninghana a D. Ritchieho, ruský překlad anglické publikace Jazyk C, úvod pro programátory (R. Berry, B. Mikinz: A Book on C) a editor/kompilátor Céčka pro ZX Spectrum s "tuláckým" překladem manuálu ve slovenštině. To je pro tuto chvíli všechno. K tomu lze přičíst ještě osobní zkušenost s assemblerem, Basicem, částečně Pascalem a Forthem a možnost konzultací s těmi, kdo Céčko ovládají. Tolik na úvod, pusťme se do toho.

Stručná charakteristika

Autoři Céčka jej sami nezařazují k vyšším programovacím jazykům. Uvádějí jej jako jazyk "relativně nižší úrovně". Céčko neoperuje přímo se složenými objekty jako je řetězec znaků, seznamy nebo pole, nemá možnost dynamického obsazení paměti, nemá ani vybavení pro vstupní a výstupní operace, příkazy typu READ a WRITE a přímý přístup k souborům. Všechny tyto a další operace známé z vyšších programovacích jazyků musejí být provedeny explicitně voláním funkcí, které spolu s dalšími obvykle obsahuje knihovna, jež se pro ten který systém počítače dodává spolu s jazykem (uživatel si je může napsat i sám). Do vytvářeného programu včleňujeme jen takové moduly knihovny, které pro jeho chod potřebujeme. Udržení jazyka na nízké úrovni má svou nespornou výhodu především v jeho snadné přenositelnosti na různé počítače (nezávislost na jejich architektuře). Tím je dána i jednodušost a kompaktnost kompilátoru Céčka.

Historicky je Céčko pupeční šňůrou spojeno s operačním systémem UNIX (původně napsaném jazykem B). Céčko bylo vyvinuto v něm a on byl pak znova a lépe napsán jím. I proto se na každém počítači vybaveném systémem UNIX objevuje Céčko jako zá-

kladní jazyková implementace. První céčková verze UNIXu pro počítač PDP-11 obsahovala 13000 řádek Céčka a pouhých 800 assembleru.

Základními datovými typy Céčka jsou znaky (char), čísla celá (int) a s pohyblivou řádovou čárkou (float). Oproti svým spřízněným předchůdcům - BCPL a B - má Céčko navíc hierarchii odvozených datových typů tvořených ukazateli, poli, strukturami, uniony a funkcemi. Umožňuje základní konstrukce řízení běhu programu, což je jeden ze zásadních požadavků strukturovaného programování: shlukování příkazů, rozhodování (if), cykly (while, for, do), klíč výběru (switch).

Výhodná je možnost užití ukazatele a aritmetické adresy. Argumenty funkcí jsou při přenosu kopírovány, proto není možné, aby volaná funkce změnila hodnotu aktuálního parametru ve funkci volající. Argument však můžeme změnit jeho "předáním adresou" pomocí přenosu ukazatele adresy, na níž argument leží. Pak jej volaná funkce může změnit. Pole je předáváno adresou svého počátku.

Funkce mohou být volány rekurzivně, lokální proměnné se vytvoří vždy při každém dalším volání. Definice funkcí nemohou být vnořovány do sebe, ale proměnné mohou být deklarovány blokovou strukturou. Funkce mohou být přeloženy samostatně. Proměnné mohou být interní ve své funkci, nebo externí - pro jeden soubor (jednu část zdrojového textu) či zcela globální. Interní mohou být statické nebo automatické.

Této velmi stručné charakteristice jazyka rozuměli znalci Pascalu, jimž byla určena především. Informaci si mohou doplnit článkem Programovací jazyk C (AR 12/88), kde jej autor srovnává přímo s Pascalem. Ostatní nechť se nenechají odradit, k neznámým termínům se podrobně propracujeme v dalším výkladu.

O syntaxi

Při pohledu na výpis zdrojového textu Céčka upoutá spousta různých závorek. Opticky zaujme především umístění svorkových. Nejméně je hranatých, užívají se jen pro vymezení rozsahu polí.

Mně se hned od začátku zalíbila jedna drobnost. Je to možnost umístění průvodních komentářů takřka kam libo. Komentáře se zapisují mezi lomítka s hvězdičkami, např.:

```
/* Tohle je komentář v Céčku */
```

Takto uzavřenou informaci můžeme napsat vlevo či vpravo od programového textu, i jen na samotnou řádku. Mezi textem a hvězdičkami může být kolik-koli mezer, nebo naopak žádná. Komentář může "přetékat" do další řádky.

Běh céčkového programu začíná od funkce s názvem main. Ji může text začínat, ale nemusí. Jako každá funkce v C, i main má u sebe kulaté závorky:

```
main ()
```

Funkci main - na rozdíl od ostatních - nelze volat z jiných funkcí a její parametry, pokud existují, se zadávají operačním systémem.

Ostatní funkce mají svá vlastní jména, která jim dává programátor. Výhodou Céčka (oproti Pascalu) je, že se nemusíme ohlížet na sled zápisu funkcí - volání funkce můžeme zapsat do místa programu, před nímž ještě nebyla definována. Kompilátor si s problémem poradí (podobně jako většina assemblerů).

Obsah funkce je ohraničen svorkami. Všem, co je mezi nimi, se říká tělo funkce. Symbolický příklad prázdné funkce:

```
nic ()  
{  
}
```

Možné jsou i zápisy:

```
nic () {}      nebo:      nic () {  
                                     }      apod.
```

Mezi svorkami jedné funkce mohou být (vždy v páru) i další, které vymezují určité vnitřní činnosti funkce. Počet levých svorek musí být shodný s počtem pravých. Svorky musejí otevírat a uzavírat určitou činnost vždy na správném místě. Opticky však nezáleží na tom, kam je v rozsahu sémantické správnosti umístíme. Může to být kde-koli na řádce, kde se činnost otevírá či uzavírá (resp. na konci předchozí či začátku další řádky) nebo na volné řádce. S ohledem na přehlednost programu je dobré, když jsou sobě odpovídající svorky opticky "pod sebou".

Podobně při zápisu řádek můžeme zleva vynechat jakýkoli počet pozic a začít psát, kde libo. Pro získání přehledu ve struktuře funkcí můžeme jednotlivé činnosti psát vždy o určitý počet pozic zleva. Ale stejně tak můžeme všechno psát úplně zkraje.

V konvenci grafického zápisu je tedy Céčko podobné Pascalu.

```
main ()  
{  
  printf ("Jak vám to jde?\n");  
}
```

Tato funkce vytiskne větu, která je v příkazu printf umístěna mezi uvozovkami. Znak změny (escape) \ upozorňuje, že nejspíš bude následovat řídicí kód tisku. V tomto případě je to new line, resp. line feed, obvykle ASCII kód 13, který posune kurzor nebo tiskovou hlavu na další řádku. Jak postupovat při odeslání textu na výstup, záleží na kompilátoru (jeho knihovně) i operačním systému počítače. Orientaci poskytne příslušný manuál.

Text mezi uvozovkami je znakový řetězec, resp. znaková konstanta. Volací příkaz printf (zkr. print format) volá stejnojmennou funkci, která si odebere znakový řetězec jako svůj argument. Z pohledu vlastního procesu jsou to parametry přenesené z těla funkce main do podprogramu printf. V tomto případě je podprogram funkcí knihovny. Jejím úkolem je "dešifrovat" argument a patřičně jej zpracovat, tj. odeslat na specifikovaný výstup. Vlastní funkce main zde v závorkách žádné argumenty pro přenos nemá - seznam jejích argumentů je prázdný.

Tak jako můžeme komentáře, text řádek a svorky umisťovat relativně kam libo, neplatí to o některých souslednostech znaků. V něčem tedy stanovenou syntaxi zápisu dodržet musíme. V poslední ukázce to platí především o těsném sousedství kulatých závorek s uvozovkami a znaků \n. Jakoukoli syntaktickou chybu kompilátor při překladu hlásí, takže se nemusíme bát, že bychom opomenutím něco pokazili. Ovšem jsou i drobné omyly, které pro kompilátor nejsou syntaktickou chybou, jenže program pak provede něco jinak, než jsme původně chtěli. K tomu se ještě dostaneme.

Jak v zápisu vidíte, za argumentem funkce musí být středník. Obecně lze říci, že vše, za čím je v Céčku středník, se stává příkazem.

Výše uvedené tělo funkce main můžeme rozdělit. Stejný výsledek tisku dostaneme třeba takto:

```
main ()  
{  
  printf ("Jak v");  
  printf ("ám to jde?\n");  
}
```

Takřka basicové roztržení. Oba volací příkazy printf můžeme napsat i vedle sebe na jednu řádku

(přímo basicově). Rozhodující je výsledná přehlednost textu a osobní vkus.

Můžete se zeptat - a co když budu chtít vytisknout uvozovky nebo znak \ uvnitř textu? Pak před každý z těchto znaků musíte umístit znak změny: \" resp. \\. Kompilátor pozná, že v tomto případě nejde o řídicí kód. Za znak změny můžeme psát i jiné řídicí kódy, symbolizované písmeny. Pro úplnost uvedu všechny možnosti:

```
\b - posun o jednu pozici doleva (BS)
\r - návrat na levý kraj řádky (CR)
\f - posun na novou stránku (FF)
\t - horizontální tabulátor (HT)
\n - posun o jednu řádku (NL)
\" - "
\\ - \
```

Pokud za znak změny uložíte jiný znak, než některý z uvedených, kompilátor jej bude ignorovat. V některých kompilátorech můžeme za znak změny psát osmičkové, dekadické nebo šestnáctkové vyjádření kódů ASCII. Tak např. \0 bude skutečnou nulou (ne ASCII 30H), \7 je zvonek (BEL) apod.

Až na stanovené výjimky se všechen text Céčka píše malými písmeny. Jsou i kompilátory, kterým je to jedno, ale to je opět výjimka.

K syntaxi jazyka se budeme průběžně vracet.

Programové řízení

Tím nejdůležitějším, co by měl člověk o jazyku vědět, je způsob přenosu parametrů a řízení běhu programu. Je zvláštní, že právě tyto informace bývají v učebnicích buď nepřehledně rozptýleny, nebo se objeví někde ke konci výkladu (na potvrzení toho, že autor to zná tak dobře, že nemá proč o tom mluvit). Výjimku netvoří ani literatura, kterou mám k dispozici. Zjištění základní podstaty vyžadovalo detektivní pátrání.

Jak už jste si přečetli výše, provádění programu začíná funkcí main (česky hlavní) - asi jako v assembleru od adresy ORG, v Basicu od čísla startovací řádky. Funkce main může být v textu umístěna kdekoli, kompilátor už si ji najde.

Běh programu se odvíjí ve směru od začátku textu k jeho konci, pokud nenarazí na příkaz změny programového řízení. Pro tuto změnu se nejčastěji používají volací příkazy. Obsahují název funkce, na kterou má být řízení převedeno. Schématicky:

```
main ()
{
první (); druhá ();
}
první () {}
druhá ()
{
třetí ();
}
třetí () {}
```

Když program v těle funkce narazí na jeho poslední pravou svorku, je to příkaz pro návrat do místa, odkud byla funkce volána. Tato svorka má tedy funkci obdobnou instrukci RET v assembleru nebo příkazu RETURN v Basicu, volací příkaz je obdobou CALL assembleru, resp. GOSUB Basicu. Z uvedeného plyne, že v Céčku programátor nemá jinou šanci, než psát strukturovaně. Jak už je pro tento typ programování typické, i v Céčku je lepší sestavovat těla funkcí s jednou zásadní, "obecnou" činností tak, aby mohla případně doplnit činnosti jiných funkcí. Pascalským s jejich procedurami není třeba nic vysvětlovat. Zkušenější assembleristé pochopí, že tu jde o obdobu tvorby struktury pomocí subrutin, jež mají univerzálnější funkce. Trochu problémů bude přechod na nové myšlení činit našim basickým, kteří těžko udržují křišťálovost struktury programu i přes možnost používání příkazů GOSUB a RETURN.

Právě pro ně mám jednu radostnou zprávu. I v Céčku existuje příkaz goto. Ale hned musím radost trochu ztlumit. Použití goto je velice zřídka. Uplatňuje se např. pro vypadnutí z programové linky při závažnějších chybách. Například:

```
funkce ()
{...
if (katastrofa) goto hlášenímálérů
...
}
hlášenímálérů: printf ("Kup mi nový disk!")
```

Když je výraz katastrofa nenulový, programové řízení skočí na adresu symbolizovanou návěštím hlášenímálérů. Pro příkaz goto je nutno podprogram uvést návěštím ukončeným dvojtečkou (podobně jako v assembleru).

Jak vidíte, v názvu funkce a návěští nesmí být mezera. Pokud by to vadilo vašemu gramatickému citění, můžete místo mezery použít podtržítka. Valná část starších kompilátorů vnímá z názvů funkcí, návěští, definic a deklarací (o nich příště) jen prvních osm znaků. Pro srozumitelnost je však můžete psát delší. V případě, že v názvech dvou různých funkcí apod. bude prvních osm znaků shodných, kompilátor vás na vzniklou dezorientaci upozorní. Pokud budeme nějakou část našeho programu v Céčku spojovat s assemblerem, musíme délku názvů obvykle podřídit maximální délce assemblerových návěští, aby se na ně mohli vzájemně odvolávat.

Příkaz volání funkcí odkloňuje programové řízení z těla jedné funkce do funkce jiné (stejně tak i poslední pravá svorka těla). Pro nepodmíněnou změnu uvnitř těla má Céčko dva příkazy: break a continue. Oba se používají uvnitř cyklů s podmínkami, break i v podmíněných výběrech. Bývají tedy bytostně spojeny s podmínkami. Například:

```
if (tohle) break;
if (tamto) break;
...
```

nebo:

```
if (něco) continue;
if (cosi) continue;
...
```

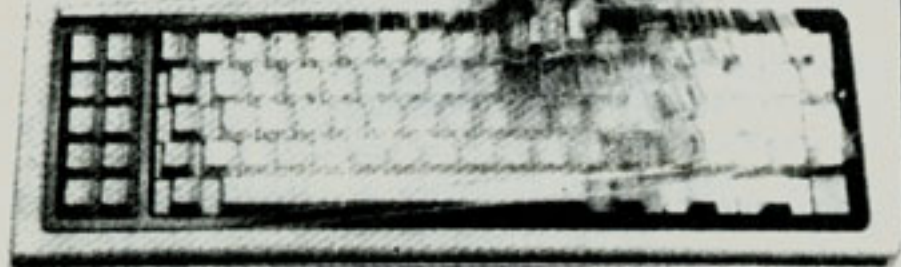
Je-li splněna podmínka, za kterou příkaz je, pak po breaku následuje vypadnutí z cyklu, zatímco po continue vlastní cyklus pokračuje provedením další operace s argumenty příkazu cyklu (vnitřní část cyklu neproběhne až do svého konce, aktivní continue zbytek cyklu přeskočí). Ve spojení s podmínkami můžeme oba tyto příkazy s přimhouřením oka přirovnat k podmíněným skokům assembleru JP podmínka, adresa - ale s tím, že v Céčku je adresa automaticky dána (buď skok na adresu za cyklem, resp. výběrem, nebo na adresu provedení další operace základních podmínek cyklu). K oběma příkazům se dostaneme v části věnované cyklům a výběrům.

Nakonec jsem si nechal příkaz return. Z hlediska změny řízení chodu programu má stejný dopad jako poslední pravá svorka těla funkce. Je to tedy návrat k volacímu příkazu (přesněji - hned za něj). Return může být bez argumentu nebo s ním. Je-li s ním, např.:

```
return (výraz);
```

pak volající funkci vrací hodnotu výrazu. Je-li bez argumentu, je vracená hodnota nepředpokladatelná. Je zřejmé, že oba výrazy mohou být výsledkem výpočtu volané funkce. To se už dostáváme k přenosu parametrů, jímž příště začneme.

NEW!
SafeSkin
Keyboard Protector



Finally! A Keyboard Protector That:

Merritt Computer Products nabízí za \$29,95 ochrannou fólii na klávesnice všech možných počítačů typu PC (nejen IBM). Jak patrně z obrázku, fólie se jednoduše položí na klávesnici a hned lze psát. Při psaní se prý vůbec nepozná, že tam fólie je. Jejím hlavním úkolem je chránit klávesnici před prachem a před programátory, kteří už se neumějí najíst jinde, než nad klávesnicí svého počítače.

Toshiba 3-in-One je 24-jehličková tiskárna, která může tisknout až v sedmi barvách. Nejvyšší rychlost tisku je 216 zn/s. Hustota grafiky: 180 * 360 bodů/palec. Vnitřní RAM 32K slouží jako buffer, který lze využít i pro definici vlastních znaků (download). Na čelním panelu je LCD display, poskytující řadu informací uživateli a dva konektory pro rozšíření bufferu o dalších 32K i o další předprogramované sady znaků. Cena \$949.

FILOSOFICKÉ ASPEKTY STROJOVÉHO MYŠLENÍ

(4)

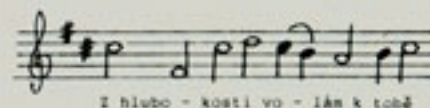
Hudba a kouzla

Nebude snad na újmu ucelenosti výkladu, nabídneme-li na tomto místě krátké zamyšlení nad kouzly v hudbě. Hofstadterova kniha věnuje hudební tematice hodně místa. V tomto odstavci se však chceme soustředit na hudbu jako takovou. Učiníme tak v pěti krátkých pododstavcích - diskusních příspěvcích - v naději, že náš pohled přinese jistá pozorování, která mohou mít obecnější význam. Celý odstavec lze přeskočit bez ztráty souvislosti; čtenář se k němu může kdykoli vrátit.

1. Hudba má s jazykem společné především to, že je to mluva (rétorický systém), zatímco výtvarné umění má (snad) s jazykem společné především to, že je to referenční systém. Podobně jako jazykové útvary vyprávějí o nějakých objektech, mají význam, sémantiku, výtvarná díla (obvykle) něco zpodobňují, obraz je obrazem vodopádu, hada, létajících rybek atd. Mluvit o rétorických vlastnostech obrazu je patrně možno jen v dosti přeneseném významu slova. Naproti tomu hudba sdílí s mluveným jazykovým projevem řečnické (rétorické) vlastnosti: člení se v čase podobně jako mluvený projev, zná podobné figury (např. gradaci) a je schopna vyvolávat v posluchači různé afekty. (V baroku byla hudební rétorika jako analogie (snad abdukcce) klasické rétoriky podrobně teoreticky propracována (Mattheson, viz /25/, /34/).) Raději však nebudeme mluvit o žádné hudební sémantice, protože se domníváme, že to je možné jen v dosti přeneseném smyslu slova. Podobně nebudeme mluvit o autoreferenci v hudbě, považujeme to za násilné.

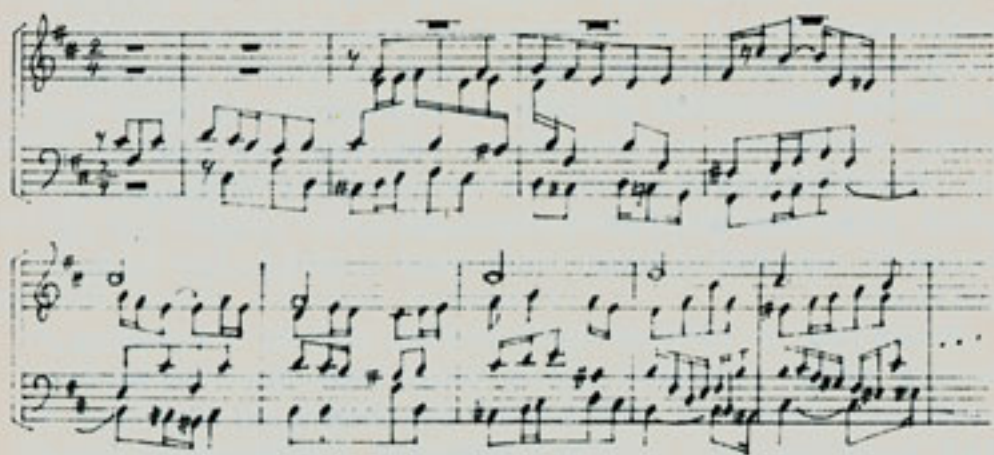
2. Estetické působení hudby je do značné míry dáno její imanentní (vnitřní) kouzelností; tu je třeba odlišit od hudebních kouzel vázaných na mimohudební systémy. Hudební dílo se liší od typického jazykového textu (byť uměleckého) tím, že je mnohonásobně strukturováno v čase (melodie

i v prostoru (harmonie) a jednotlivé části a částičky se k sobě mnohonásobně vztahují. Motivy (krátké úryvky) i témata (větší celky) se opakují, mění, deformují či transformují, spojují atd. Zvláště bohatá je v tomto ohledu polyfonní hudba - a J.S.Bach je její nedostižný král. Kontrapunktické techniky současného smysluplného průběhu samostatných rovnoprávných hlasů jsou velmi bohaté a jako pravidla (omezení), která si autor klade, mají zřejmý estetický účinek: budí pocit řádu, který je v dynamickém napětí s jedinečnou emocionální náplní díla. To potud, pokud je posluchač schopen tento řád, tj. přítomnost a dodržení pravidel, uslyšet anebo alespoň přibližně vycítit. To, zda jistou pravidelnost uslyšíme, závisí na tom, jak je složitá a zda takovou pravidelnost potenciálně očekáváme, tj. zda odpovídá stylu dané hudby (fugovaný začátek skladby; předimitace chorálové melodie v chorálové předehře apod.). Jako příklad nechť poslouží začátek malé verze chorálové předehry Aus tiefster Noth schrei ich zu dir z Bachových Dogmatických chorálů. Uvedený německý text má v chorálu tuto melodii:



Z hlubo - kosti vo - lán k tobě

Tato melodie zní v předehře v nejvyšším hlase (sopránu) od 6. taktu; předtím ji však tři ostatní hlasy postupně hrají v čtyřikrát rychlejším tempu (osminy), a to nejprve tenor, pak bas (v protipohybu - zrcadlový obraz), pak alt; jednotlivé nástupy jsou v těsně, tj. další hlas začíná citací úryvku dříve než předchozí skončil. Samá kouzla, chcete-li, ale kouzla vnitřní, patřící k běžnému arzenálu barokní skladebné techniky, a co více: kouzla slyšitelná, zejména pro toho, komu (jako Bachovým současníkům) je základní chorál dobře znám.



Naproti tomu najdeme v hudbě (u Bacha i u jiných) jiná kouzla, jakoby kouzla ve vlastním smyslu: často se neslyší, nýbrž spíše vidí (v notách), čtou, či jsou nám prozrazena. Rozoznáváme, že hudební útvar (motiv či celá skladba) kromě zákonité struktury uvnitř očekávaného estetického systému (třeba barokní polyfonie) vykazuje zajímavou strukturu v jiném systému, strukturu nevnímání nebo jen obtížně vnímatelnou poslechem znějící hudby. Jiný systém může být původnímu značně příbuzný: tak např. složitější račí kánon (jako Bachův z Hudební obětiny) slyšíme jako výbornou hudbu, ale patrně ne jako račí kánon. Ze jde o račí kánon ověříme spíše z notového zápisu:

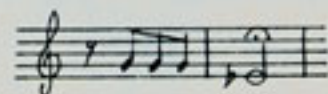


Jde tedy o systém symetrií, ale povýšený na samostatný systém, už nesloužící hudebně estetickým cílům (vnímatelných sluchem). Jindy jde o drasticky jiný systém. To se týká např. hudebních kryptogramů, založených na tom, že noty mají za svá jména písmena (A, B, C, D, E, F, G, H, S=es): citace jména B-A-C-H v poslední fuze z Bachova Umění fugy (plus spousta skladeb jiných autorů na toto téma), motiv S-C-H-A u Schumana apod. Viz pozoruhodnou partii o hudbě a zraku v kap. VIII Mannovy knihy /10/ (s.68n). (Tato kniha je o hudbě a ďáblu, takže se tam dá nalézt mnoho o hudbě a kouzlech; m.j. též motiv H-E-A-E-S z milostné písně hlavního hrdiny - skladatele Leverkühna. O motivu se později ukáže, že je to kryptogram pro Hetaera Esmeralda, což je skladatelova osudová partnerka.)

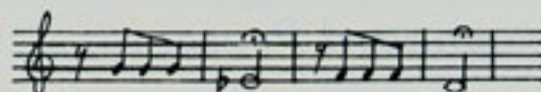
3. Hudební kouzla lze rozdělit na kouzla nezávislá na notovém zápise a kouzla, která na něm podstatně závisejí. Kánon (či račí kánon) je příkladem kouzla invariantního vůči způsobu notace; podobně kouzla spočívající v tom, že číselky vyskytující se v textu, který je zhudebňován, mají protějšek v počtu hlasů či not ve skladbě (např. Orlando Lasso, Svatba v Káni Galilejské: pro šest džbánů s vodou, šest hlasů aj. - viz /10/). Kryptogramy jsou příkladem kouzla podstatně závislého na způsobu notace, tj. kódování hudby.

4. Cesta k prožití hudebního kouzla může vést od významu v hudebním systému k významu v mimohudebním systému nebo naopak. Slyšíme přesvědčivou hudbu; dodatečně ověříme, že jde např. o račí kánon (jsme překvapeni, vnímáme kouzlo). Autor skladby postupuje obráceně: rozhodne se, že bude psát přísnou formou a pak se snaží v rámci zvolených pravidel (omezení) vytvořit krásnou hudbu. Podaří-li se mu to, vytvořil kouzlo. Zajímavý

příklad hudebního kouzla rozpoznávaného od mimohudebního významu k hudebnímu předkládá Ilja Hurník (/4/, s.77): Skladatel Schmeidtke očekává o půlnoci zázračné zjevení inspirace ve společnosti přítele Rosenzweiga. Půlnoc mine, nic se neděje, Schmeidtke štká: "Nichts, nichts!", Rosenzweig ho utiňuje: "Aber geh, geh, geh, es wird schon wieder alles gut!" Schmeidtke zbledne, neboť slyší: g, g, g, es:



"Inspirace!" křičí a snaží se pokračovat:



(Pokračování v následujícím bodu.)

5. Hudební kouzlo vyžaduje, aby struktura, o niž jde, chápaná jako hudba, byla kvalitní (krásná, esteticky významná) buď sama o sobě nebo v širším kontextu díla. Hraje-li vaše dítě na piáno a cvičí stupnice v protipohybu, posloucháte triviální račí kánon; místo kouzla prožíváte spíš muka. Známá písnička "Bratře Kubo" je triviální (ač milý) čtyřhlasý kánon, žádné velké kouzlo. Tento popěvek pozoruhodným způsobem "zakouznil" Gustav Mahler ve své 1. symfonii: vystupuje tam (v pozměněné podobě, ale jako kánon) v úžasném kontextu rozsáhlé volné věty. Kouzlo není především v tom, že je to kánon a je teď v nádherném kontextu, nýbrž v tom (také v tom), že s překvapením rozpoznáváme, že útvar, který jsme měli zařazen mezi nenáročné popěvky (zařazen v systému popěvek, chcete-li) se najednou stal Hudbou Par excellence. (Jde o abdukcii?) A konečně ubohý Schmeidtke: z kouzla nebylo nic. Rozumný Rosenzweig prohlásil motiv za fádni, Schmeidtkův pokus o pokračování za ubohý "šustrflek" a navrhl mu, aby napsal nenáročný singspiel. (Co z tohoto motivu udělal Beethoven, nemusím čtenáři napovídat.)

Naše zamyšlení nad kouzly v hudbě končí. Čtenáři, který je nepřeskočil, nabídneme několik otázek:

- Která z kouzel popisovaných v předchozích odstavcích této práce (a netýkajících se hudby) lze chápat jako rozpoznání, že kromě dobrého významu v daném, zamýšleném, "domácím" systému má sledovaný objekt dobrý význam v nějakém jiném, překvapivě "cizím" systému?

- Umíte ještě něco jiného z úvah tohoto odstavce přenést na kouzla jinde než v hudbě?

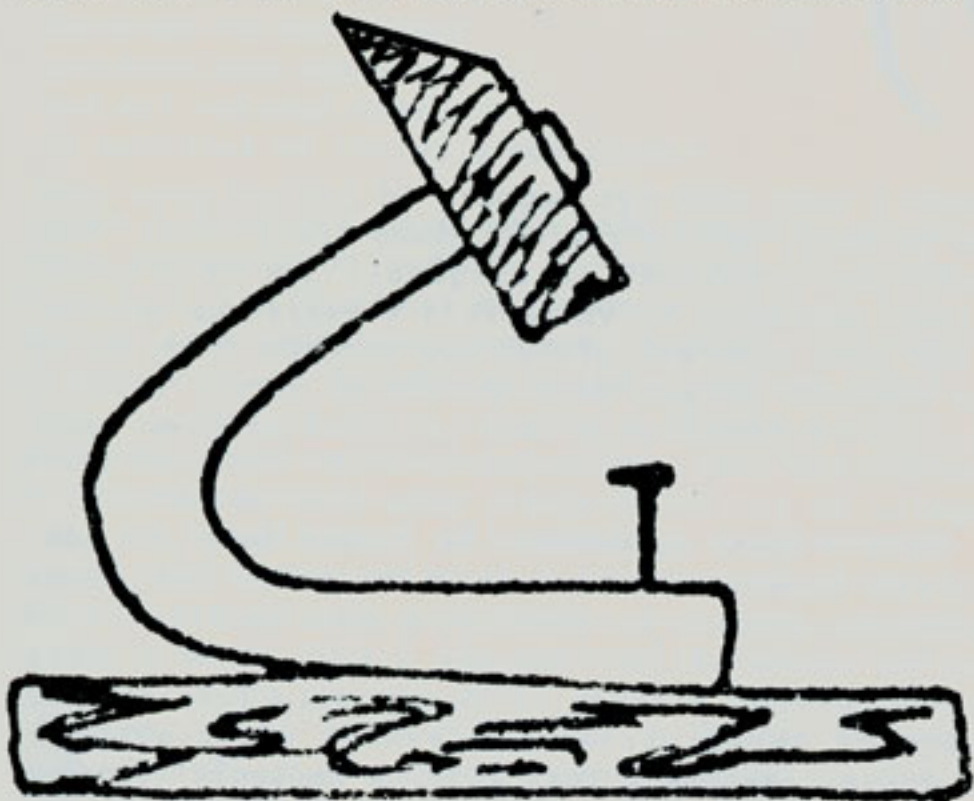
- Jaký je hudební význam posloupnosti prvních písmen pěti číslovaných pododstavců tohoto odstavce?

Pauza

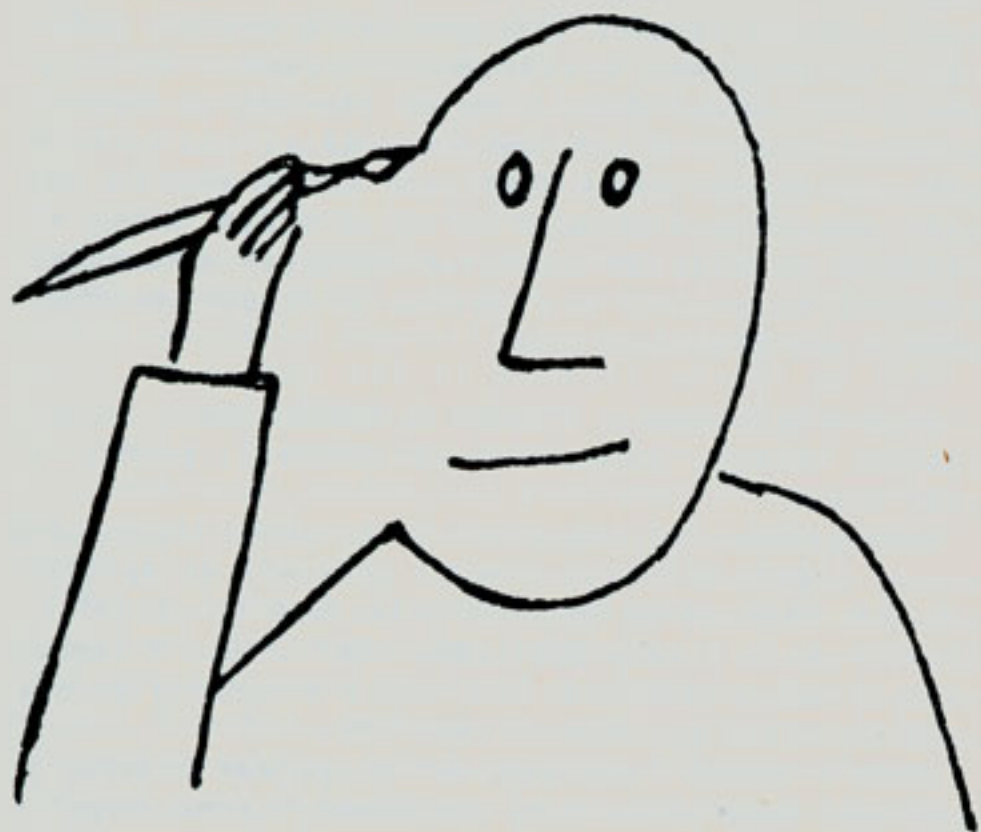
Čtenáři, který by nyní rád přerušil čtení a zaposlouchal se do své oblíbené hudební skladby, nabízíme malou přestávku, řekněme na deset řádků. Byli bychom však rádi, kdyby jí též využil k zamyšlení nad tím, co dosud přečetl. Bylo by prospěšné, kdyby si uvědomil, jaký je vlastně jeho názor na popsání kouzla. Je to jen bezúčelné hraní, jednou s trivialitami, jindy se zcela atypickými a vykonstruovanými situacemi? Anebo má pocit, jakoby v temném sklepe konečně nahmatával obrysy zapomenuté lucerny? (Nepřehlédnout podivnou metaforu: "nahmatávat lucernu"!) V tom případě nechť se pokusí znovu prolistovat přečtené a nejprve si sám všimnout jakýchkoli souvislostí, které ho napadnou - souvislostí s vlastními zkušenostmi i souvislostí jednotlivých vzpomínutých kouzel. Můžeme už pokračovat?

V řadě popsaných kouzel si můžeme povšimnout několika opakujících se motivů. Nápadné je např. ono napětí mezi vždy dvěma odlišnými popisy jednotlivých kouzel: jednak v podobě rozvinuté: ta je zpravidla hierarchická, vyžaduje potenciálně nekonečný regres a respektuje přírodní nebo mentální čas. Za druhé v podobě svinuté: ta je zas podivně zacyklená, často protismyslná, v zastaveném čase.

Od jednoho popisu ke druhému lze obvykle přejít potlačením resp. přidáním jedné "dimenze" (např. času), která příslušnou situaci strukturuje. Přechod od svinutosti k rozvinutosti (např. uvažování perspektivy v Escherových obrázcích, výšky tónu v hudbě, hierarchie interagujících systémů, provedení rekurzivní procedury, dosazování do autoreferenční věty apod.) zpravidla jev vysvětluje, byť i za cenu konfrontace s nekonečnem a



Obr.14. Kvázisubjekt



Obr.15. Pozitivní a...

ovšem i za cenu ztráty půvabu. Přechod od rozvinutosti ke svinutosti (průmět "nevhodným" směrem, zahryznutí se do vlastního ocasu, napsání programu s odkazem na sebe sama, schématické nakreslení zpětnovazebního obvodu atd.) zpravidla generuje paradox, poskytuje však jakýsi hlubší vhled do celé situace.

Oba zmíněné způsoby popisu jsou komplementární v tom smyslu, že je nelze kombinovat - asi tak, jako vtípu se nelze současně smát i racionálně rozebrat příčiny jeho vtípnosti. V komplementaritě svinutosti a rozvinutosti lze tušit jeden z prin-

cipů možného paradigmatu "nové" kybernetiky (kdo zná /8/, ví co tím chceme říci).

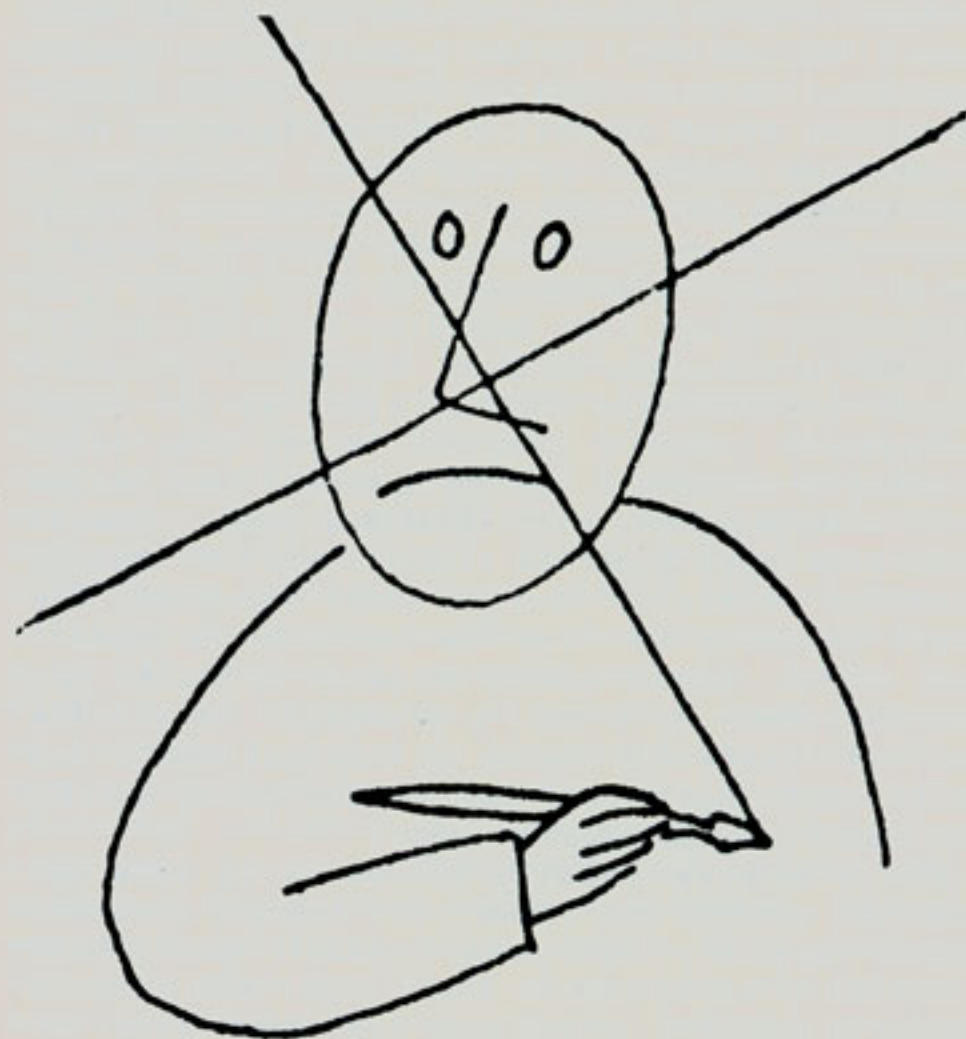
Druhý princip souvisí s pojmem samovztažnosti. Setkali jsme se s ní v mnoha podobách: autoreference (odkazování k sobě), seberegulace (= regulace zpětnou vazbou), sebevolání (rekurze), sebeinformace, sebezrcadlení, sebenápodoba. Je to zvláštní princip, pro naše další úvahy velice důležitý - dává totiž vodítko, jak se postavit k problému člověk-stroj.

Samovztažnost má dvě základní podoby. V jedné ji všichni dobře známe - u sebe samých. My se neustále vztahujeme k sobě samým, víme o sobě, myslíme si, že myslíme a tedy jsme, krmíme se, mluvíme o sobě, chlubíme se, občas si i děláme ze sebe legraci. Zde je samovztažnost projevem subjektivity v pravém slova smyslu.

Druhý typ samovztažnosti je ten, který se opakovaně vyskytoval v našich kouzlech. Ta podivná samovztažnost věcí (případně jejich genidentit), která jako by je dočasně oživovala, předstírala, že mají své vlastní "já". Zde je samovztažnost (nikoli projevem, ale) původcem virtuální subjektivity, kterou nazýváme kvázisubjektivitou.

Setkáme-li se s kvázisubjektem, zpravidla se zasmějeme. Proč? Je to snad smích člověka, který se dívá na nedokonalou karikaturu sebe sama?

V knize /3/ má Hofstadter zajímavou poznámku: "Předměty mají speciální a výlučný vztah k sobě, který omezuje jejich schopnost působit na sebe stejně, jako mohou působit na jiné předměty."



Obr.16. ...negativní vztah k sobě

(s.278). Tužka nemůže psát na sebe sama a sekera nerozštípne vlastní topůrko. Poslání nástrojů je v tom, co mohou dělat s jinými předměty, nikoli se sebou samými. Poruší-li toto pravidlo, stanou se kvázisubjekty (obr. 14).

Projev samovztažnosti může mít dvojí polaritu - pozitivní a negativní. Pozitivní polarita podporuje smysl (kvázi)subjektu, negativní působí proti němu: "Jsem pravdivá věta" versus "Jsem nepravdivá věta", kladná versus záporná vazba (vlastně naopak), sebereprodukce versus sebevražda, obr. 15 versus obr. 16. (Co had Uroboros? Krmí se anebo požírá?)

Záhy uvidíme, jakou úlohu může hrát negativní samovztažnost v matematice. Vyžádá si to však určitou přípravu.

(pokračování)

Ivan M.Havel, Petr Hájek

POČÍTAČOVÁ SCIENCE FICTION

(5)

(-15-) Laster del Rey: Instinkt. Osoby: Sentři, Cepět ... roboti 200. generace, 324MD2771 - robot 20. generace; On a Ona - lidé. Místo a doba: Biologická (robotí) laboratoř, dost vzdálená budoucnost.

Sentři se svým kolektivem se již několik let snaží vyrobit živou hmotu, která na Zemi vyhynula. Dotáhli to až k prasatům, člověk se jim však stále nedaří. Přitom vědí, že bez pomoci člověka nemohou pokračovat ve svém vlastním vývoji: jejich někdejší tvůrci (jimž se cítí i jistým způsobem sentimentálně zavázáni za svůj vznik) jim nezabudovali obvody pro "instinkty" = zabudované reakce mimo inteligenci, nenaučitelné, které jim chybějí /I54/; i když, mimochodem, to byly právě tyto instinkty, které lidi zahubily (instinkt boje). Povídka je situována do doby, kdy Sentři přivezl z Venuše staromódního robota, který byl až dosud zasypán v jedné z jeskyní. Stroj nesl název 324MD2771-CHIRURG a jak se ukázalo, jeho úkolem bylo skutečně poskytovat lékařskou pomoc lidem. Z jeho paměti se podařilo získat chybějící detaily lidského hardwaru. Povídka končí okamžikem, kdy Cepět oznamuje Sentřimu, že tentokrát se experiment zdařil. Sentři spěchá do sálu, ve kterém byly znovustvořeny On a Ona, a přemýšlí, jak člověka oslovit. Závěrečný dialog po otevření dveří: On: Další robot? Okej, pojď dál. Co zas chceš? Sentři: Nic Pane, jen vám sloužit /I55/.

I54 * Otázka rozdílu člověk-stroj může být jasnější strojům než lidem * Při častých diskusích těchto rozdílů si nedovedeme uvědomit specifickou lidskosti; kdybychom se znali lépe, mohli bychom snadněji posoudit, které naše rysy bude těžké, ne-li nemožné vložit do programů.

I55 * Člověk má (možná nechtíc) vlastnosti, které mu vždy zajistí hegemonii nad neorganickými formami života * Tenhle předpoklad - či závěr - je asi neoptimističtější ze všech, které se v úvahách o koexistenci vyskytují: máme to dobré a ani nevíme proč. Takové iracionální momenty lze ovšem pozorovat i uvnitř lidské společnosti samé - jsou jedinci, kteří dovedou získat nadvládu (v té či oné formě) jaksi automaticky, bez zásluh a bez uvědomění si, v čem jejich "síla" či kouzlo vlastně tkví. Mnozí náboženští a političtí vůdci byli takovým faktorem obdařeni, ať již ho využili v dobrém či zlém. Mimochodem, jednou z mála knih, které měl Sentři k dispozici jako relikv z lidské éry, byl Mein Kampf, kteréžto dílo se mu zdá nepochopitelné a mysteriózní, ale jasně dokazující, že člověk byl plný instinktů. Srv. s I12.

AE56 * Není vyloučeno, že potenciální boj o prvenství člověk versus robot, bude své vítěze alternovat * Chvilí mohou lidé sestrojovat roboty,

chvilí tomu může být naopak. Ad infinitum, nebo do té doby, než to rozřeší někdo třetí (I46). Nebo může jít o rekurzi inteligencí, jak její první členy popisuje I. Varšavskij v Peperuum mobile - lidé jsou donuceni předat moc strojům třídy A, ty po dvaceti letech mají potíže se stroji třídy B (...?). Z téhož díla bude užitečné ocitovat jeho motto (viz T50): "Metakynetikům, kteří si vážně myslí, že to, co si myslí, máme brát vážně".

IA57 * Není vyloučeno, že člověk bude své znalosti robotů alternovat * To jest, že jednou přestane chápat, co stvořil, a bude se divit, když na to znovu přijde. Dokonce i když půjde o zcela nekonfliktní situace. V Pociu síly I. Asimova to nějakou chvíli trvá, než výkvět představitelů lidstva, řekněme z roku 2050, s nadšením (zejména z vojenských důsledků) pochopí, že člověk - sám, jen s kusem papíru! - je schopen bez kalkulačky znásobit dvě čísla. Objev Algoritmu násobení v době stěhl řízených těžkopádnými a drahými počítači je objevem, který umožní nasadit do boje lacinou lidskou sílu = objev století na úrovni atomové bomby! A teď laskavě uvažte: ono je to v principu možné! Kolik lidí bude za pár desítek let znát algoritmus dělení? Co by se stalo, kdyby ti všichni na nějakém světovém kongresu matematiků náhodně zahynuli, ale výroba příslušných VLSI běžela bez přerušení dál?

(-16-) M. Crichton: Člověk na konci. Osoby: cvok, lékaři. Místo: psychiatrické oddělení, později i svět mimo.

Nebezpečnému psychopatovi je do mozku voperována sonda napojená na centrální počítač. Má za úkol odhalit bližící se záchvat a aktivně ho potlačit. Člověk takto řízený počítačem však unikne z nemocnice a začne ohrožovat své okolí (kterážto pointa není důležitá pro hlavní ideu I58).

IA58 * Běžné "softwarové" propojení člověka a počítače bude během doby nahrazeno "hardwarovým" * Přitom může být člověk periferií, jako v této povídce, nebo naopak procesorem. V S.R. Delaneyově povídce Nova jsou celé výrobní procesy přímo řízeny nervovými impulsy tak, jako již dnes některé pohyby umělých končetin. Prostě přijdete do práce, zastrčíte se do zásuvky a představujete si, že pracujete.

AE59 * Biologické počítače jsou na obzoru * Genetické inženýrství se již dnes vážně zamýšlí nad biologicky řízenou výrobou technických prvků. A ani myšlenka technicky adaptovaného člověka samého se nezdá být příliš fantastická - jako např. v Asimovově povídce Okruhliak na oblohe, kde snížením dielektrické konstanty na synapsích neuronů (působením diamagnetického pole) bylo

dosaženo zrychlení šíření vzruchů a tím i schopností mozku ve smyslu normálního i "para" myšlení. Skoro se nechce věřit, že by takovéhle pokusy někde ve skutečnosti neprobíhaly.

(-17-) R.Brantigan: All Watched Over by Machines of Loving Grace.

I like to think
of a cybernetic meadow
where mammals and computers
live together in mutually
programming harmony
like pure water
touching clear sky
...
I like to think
of a cybernetic ecology
where we are free of our labors
and joined back to nature
returned to our mammal
brothers and sisters
and all watched over
by machines of loving grace /I160/

IE60 * Živý a neživý svět může vytvořit přirozenou harmonickou jednotu * Předcházející úvahy dávají i jistý návod na computerizovaný ráj: všechny naše představy, myšlenky a přání, napojené na stroj, který je realizuje buď v reálném světě, nebo - pokud je to v rozporu s přáním jiných - navozuje představy splněných přání, ev. tato přání bezbolestně potlačí. Viz i I25 až I27.

E61. * Rozdíl mezi přirozeným a nepřirozeným nemusí být neodstranitelný * Rousseauský návrat k přírodě s počítači schovanými pod drny, aurea aetas lidstva, zlatý věk navozený vyspělou technologií, louka půvabná květy ocunu a čekanky, ale s hmyzem, proti jehož dotěrnosti je člověk softwarově chráněn - to může být nepřijatelné pro čtenáře starořeckých bukolických zpěvů. Zkusme si ale představit, že všechny ty VLSI obvody, které ráj implementují, nejsou produkty ukrytých fabrik a nejsou instalovány specializovanými četami JZD, ale rostou tak, jako květiny. A působí blahodárně (a možná i bez naší znalosti) právě tak, jako dnes léčivé byliny, jejichž mechanismus dodnes není vždy přesně znám, ale přesto přináší úlevu. Na některých suchých stráních prostě vedle mateřídoušky obecné (*Thymus serpyllum*) poroste mateřídouška pascalovitá (*Thymus pascali*), jejíž nať bude blahodárně působit na tok informací v CNS právě tak, jako nať té nynější působí na oběh kyslíku v průduškách a potravu v gastrointestinálním traktu. A po horách budou chodit moudré babky-programátorky ovládající problematiku zpracování dat intuitivně a empiricky asi tak, jako babky-kořenářky z minulého století rozuměly chemii.

EA62 * Jednota živého a neživého může mít i negativní bod stability * Symbióza lidí a strojů může být vysloveně ztrátová z hlediska (dnešních) hodnot člověka. Můžeme se stát nedobrovolnými terminály celku, jehož celkový účel nechápeme. Ostatně není vyloučeno, že se v takové situaci nevědomky právě nacházíme. Nebo k ní nevědomky napomáháme - jako v povídce Ch.Hoddera-Willamse *Fistfull of Digits* (Plná hrst číslic) se za našimi zády právě může rodit počítačová síť typu Servex, která již sleduje své vlastní cíle. Síť, která nás pozoruje a zkoumá, která se nerozpakuje někoho si k prozkoumání vypůjčit a - když bude mít dobré parametry - zapojit do sítě jeho mozek přímo. Na závěr citaci jedné z těchto potenciálních obětí, citaci, která ukazuje i na to, že stroje těžko převezmou vládu nad lidmi, aniž by mezi nimi vybudovali svou pátou kolonu: Many people fear computers, because they seem to impersonate human beings. But they are wrong. What they should fear is the opposite: human beings who impersonate

computers (Mnoho lidí se bojí počítačů, protože se jim zdá, že zosobňují člověka. V tom se dopouštějí omylu. Naopak by se měli bát lidí, kteří zosobňují počítače). A když už jsme u toho optimismu, vyplývajícího z myšlenky, že až tu nebudem, nebude to stát za nic, je dobré upozornit, že rovnice "vrcholná inteligence = x + počítač" nemá zdaleka jediné řešení (x = člověk) ani když odhlédneme od pořadí důležitosti. V povídce C.Simamaka Když ještě žili lidé je např. uvažována eventualita x = pes; pro ještě půvabnější řešení x = /pes odjinud, pes od nás/ viz /54/. Filmoví fanoušci ostatně znají i x = opice z Planety opic. Viz též I46.

Ax Pozdě ale přece dorazil na redakční stůl příspěvek /64/, reprezentující třídu povídek, v nichž symbióza člověk-stroj vykazuje humorné, ale nikoli nemožné rysy. V povídce J.Fabiána Ráno člověk obsluhovaný dokonalým domácím počítačem jde do plně automatizovaného provozu - házet lopatou (čtenáři Dikobrazu znají tento motiv z obrázku, v němž jsou na pole pomocí nablýskaných vrtulníků dopravovány babky s motykami). B.Švandová ve svém Opusu bez názvu představuje zase výzkumníka, kterému počítač nutí program práce i spolu s výsledky.

(-18-) S.Lem: Stotřicet sedm sekund. Osoby: Já - redaktor, Milton Hart - informatik z MIT, IBM 0161 - počítač upravený pro redakční práci. Místo a doba: redakce agentury UPI, současnost.

Povídka je půvabná tím, co zde bohužel musíme pominout: neustálou tvorbou hypotéz, jimiž se vypravěč pokouší vysvětlit podivné chování počítače, a jejich překonáváním. Zde se musíme omezit až na závěrečnou Hartovu teorii. Jádro zápletky: Počítač přijímá zprávy z dálnopisů a navrhuje rozmístění sazby. Navíc je spojen s různými obecnými informačními zdroji, pomocí nichž ověřuje a doplňuje informace a podílí se i na frazeologické ediční práci. Zábavné to začíná být, když redaktor zjistí, že počítač je schopen produkovat text, i když je odpojen od hlavního zdroje - dálnopisů (zaviní to náhodné vypadnutí konektoru mezi dálnopisem z Ankary a počítačem). Sérií pokusů překvapený redaktor zjistí, že: 1/ Počítač extrapoluje texty zpočátku zcela přesně (včetně kvantitativních údajů), později se diference zvětšuje a po 137 sekundách je zcela bezradný. 2/ To, co říká závisí na tom, co se skutečně stane (nikoli na informaci, která je fingovaně posílána dálnopisem). Dovede tedy - v rámci limitu 137 předvídat události. 3/ Jeho schopnost závisí na napojení na celou informační síť (kterou měl snadno k dispozici v nočních hodinách, kdy se sestavuje uzávěrka), a to i tehdy, má-li predikovat jevy odehrávající se těsně vedle něj (házení kostkou). V brilantní, typicky lemovské povídce plné souběžných nápadů (Hart např. vyslovuje domněnku, zda Bůh nemohl náhodou stvořit svět neúmyslně /I63/; nebo: zjistit přesný tok informací v počítačové síti je práce pro detektiva, kterým by nemohl být nikdo jiný, než zase počítač /I64/), se nakonec ukazuje jako jediná konzistentní /I65/ hypotéza, že dostatečně složitá počítačová síť představuje organismus nového typu, pro nějž přítomnost není bodová, ale je třeba intervalem v délce 137 sekund (psychologická "přítomnost" člověka je údajně půl sekundy), a v tomto rozpětí je schopen předvídat. To, co se u nás má stát teprve za dvě minuty, děje se už pro počítač zrovna tak, jako pro nás to, co momentálně nevnímáme /I66/. Nebyl by to Lem, aby si neodpusťil i krásný filozofický závěr: máme-li svobodnou vůli, rozprostírá se až za hranici 137 sekund, ač o tom z introspekce nic nevíme. Hart ještě zjistí, že schopnost předvídat se nevztahuje na atomární děje - atomy mají nulovou přítomnost. Z toho, že naše činy předvídatelné jsou, usuzuje na to, že

myšlení není kvantové povahy. Povídka končí "aplikačními úvahami": počítač jednak zrekonstruoval jednu havárii, jednak bylo na celou věc uvaleno mlčení - 137 sekund neomylné předpovědi se za jistých okolností rovná rozdílu mezi zničením a záchranou světa.

I63 * I velká díla mohou být stvořena neúmyslně
* Nejde jen o pánaboha a svět. Je docela představitelné, že program, který jste za jistým účelem napsali, může dělat daleko víc, než jste neúmyslně stvořili ne-li svět, tedy nějaký dosud neznámý jeho model. Alchymista, který neuspěl při hledání elixíru života, objevil třeba výborné lepidlo; s malou, ale přece jen nenulovou pravděpodobností je možný i opak. A zcela prozaicky: věnujeme dost úsilí nalezení třídy všech úloh, které daný program (resp. jeho mírné modifikace) řeší? Nemůžeme něco ze svých hotových programů prodat ještě někomu jinému pro řešení zdánlivě odlehlého úkolu? Nemáte v zásuvce patent na docela dobré lepidlo, aniž o tom víte?

I64 * Svět počítačů je uzavřen vzhledem k operaci monitorování * To jest činnost počítačů mohou s úspěchem sledovat, kontrolovat, hodnotit... zase (JEN?) počítače. Nehluboký námět k hlubokým úvahám: vzhledem ke kterým dalším operacím (s "JEN" i bez něj) je svět počítačů v tomto smyslu uzavřen? Dnes a zítra? Vezměte např. "stvoření". Co o počítačích mohou zase (JEN) počítače?

T65 * Některé sci-fi představují dobrý trénink v hledání netradičních modelů * Řada povídek (zejména S. Lema, I. Asimova, A. Clarka, ale i jiných) odvozuje své pointy velice logickým způsobem a učí vyrovnat se se "zázraky" co nejpřirozenějším, byť jakkoli fantastickým způsobem. (A fantastičnost téhle povídky převyšuje podle našeho soudu podivuhodnost nejen libovolných n-hlavých stvoření

(n je větší než 1), ale i systémů pracujících v běžné logice na neběžném technologickém základě - jak poznáme v dalších povídkách kapitoly.) Připravuje nás na období, kterým prošla klasická fyzika: nebát se neočekávaných závěrů, pokud jsou konzistentní s předloženými fakty. Člověk se snadno nechá zaskočit nevysvětlitelnem a vzdává se, pokud se mu nevtěsná do vžitých představ. Těžko tvoří hypotézy mimo oblast dosavadní zkušenosti. To lze do jisté míry pozorovat i v banálních záležitostech: kolikrát jste byli překvapeni, že "evidentně správný" program nefungoval, jen proto, že vás zpočátku ani nenapadlo, že -0 se nemusí chovat jako +0?

I66 * Čas počítače může být jiný než čas člověka
* Tj. může mít jiné kvalitativní vlastnosti, tak jako mikrosvět je fyzikálně odlišný od makrosvěta. Rychlosti logických operací v počítači jsou daleko za horizontem naší představivosti. Teorie, že složité informační systémy vytvářejí své vlastní chápání času, může být zajímavou pohádkou; je však těžké ji na základě dosavadních zkušeností odsoudit jako absurdní, vymyšlenou jen pro zábavu. A o to jde - neodsuzovat apriori. Byla by to chyba asi stejného druhu jako nekriticky akceptovat. Ostatně uznejte, že je to hezky vymyšleno - čím je něco chytřejší, tím větší má přítomnost: pro kámen je přítomnost nijaká, komár zachytí tisícinu vteřiny, člověk půl vteřiny, superpočítač 137 sekund a pro pánaboha je teď pořád! Funguje to, i když zaměníte "přítomnost" "zorným polem paměti" - tím, jak chápe kontinuitu dní a událostí dítě a dospělý, a co vše z historie je schopen naráz vzít v úvahu stroj. Srv. I44.

(pokračování)

RNDr. J. Hořejš, CSc., ing. J. Franek

ZÁVADA NA ZX SPECTRU 128 K

U tohoto počítače se objevila zajímavá závada. Asi po 40 minutách po zapnutí přístroje přestaly pracovat některé programy. Uvedenou závadu vám nejlépe objeví program D-WRITER, který po načtení do počítače odmítá pracovat. Nadměrně se zahřívá chladič a stabilizátor. Po odkrytí počítače (klávesnice mimo počítač) pracuje bez závad, ale při opětovném zakrytí se závada objeví znova.

Některé série těchto počítačů mají uvnitř krytu i na spodní straně klávesnice nanesenou stříbrnou vodivou vrstvu, která způsobuje zkrat mezi klávesnicí a počítačem.

Odstranění závady:

Nejdříve uvolníme klávesnici (šrouby na spodní straně). Opatrně ji nadzvedneme, posuneme směrem k sobě a položíme. Tak se zachytí o vyvýšený ná-

stavec pro úchytný šroub a v této poloze bude zajištěna. Horní okraj krytu počítače natřeme po celém obvodu dvakrát izolačním lakem. Použil jsem černý matný lak na školní tabule. Lak, který chcete použít, nejdříve vyzkoušejte, aby neleptal kryt. Dále natřeme zásuvku RS232 a video modulátor přelepíme izolepou. Klávesnice je spojena se zemí přes vodiče, tak se nemusíte obávat účinků statické elektřiny.

Po důkladném zaschnutí laku dáme počítač do původního stavu. Takto upravený počítač už mi pracuje více jak rok bez nejmenší závady. Pracovní teplota chladiče se po jedné hodině po zapnutí pohybuje kolem 38 stupňů Celsia. Úspěch úpravy počítače si vyzkoušejte opět programem D-WRITER.

F. Kratochvíla

LECROM

PRO

ZX SPECTRUM

Mezi uživateli ZX Spectra jsou rozšířeny různé verze EPROMek, které odstraňují známé chyby původní ROMky a většinou "umějí" ještě něco navíc. Mezi nejrozšířenější patří bezesporu ISO ROM, TURBO ROM a LECROM. Ne každý však s novou EPROMkou získá i příslušný návod, takže mnohdy ani nemůže v plné míře využít všech možností své xxxROMky. Ve snaze tuto mezeru zaplnit, jsme před časem uveřejnili popis úprav v TURBO ROM, nyní přinášíme podrobný popis LECROM přímo od jejího autora, Jiřího Lamače. Do některého z příštích čísel chystáme popis ISO ROM.

Věříme, že tímto článkem vyřešíme řadu problémů především těm, kteří LECROM v2.3 již získali, (rychlost šíření je neuvěřitelná), přitom však mají k dispozici nanejvýš návod ke starší verzi, který se bez vědomí autora (i když s jeho dodatečným souhlasem), dostal do sborníku ostravské pobočky ČSVTS.

Ze všech nám známých xxxROMek (a není jich málo), nejvýrazněji zvyšuje užitnou hodnotu ZX Spectra právě LECROM. Pokud byste po prostudování následujícího návodu měli o ní zájem, napište nám. V případě většího zájmu bychom se pokusili zajistit dodávku naprogramovaných EPROMek, nebo alespoň distribuci programu na kazetách.

LECROM

Hlavním popudem vzniku nové ROMky bylo více než 1K volného prostoru ve staré ROMce. Kromě toho, že byly opraveny její chyby, LECROM umožňuje psát basicové příkazy jednotlivými písmeny, používat zabudovaný monitor a mít k dispozici řadu dalších užitečných funkcí.

Více jak 1K volné paměti nemůže nikoho nechat dlouho klidného. Je to relativně dost, současně ale i málo (těžko se sem vejde assembler 280). S každým bajtem paměti musí být naloženo co nejšetrněji. Při psaní nových rutin jsem si nejméně

třicetkrát uvědomil známou pravdu - každý program jde zkrátit aspoň o jeden bajt. Vždycky když se někde našlo třeba šest volných bajtů, hned jsem uvažoval, jakou novou funkci by bylo vhodné přidat. I přes dost stísněné poměry se přece jen leccos podařilo. Nakonec - posuďte sami:

Opravy chyb původní ROMky

Stará ROMka Spectra měla v sobě několik méně významných chyb, které v LECROM již nejsou. Především je opravena klasická chyba v rutině pro zpracování nemaskovatelného přerušení.

Při dotazu "scroll?" nebo hlášení "Start tape..." a čekání na stisk tlačítka se dříve po stisku EXTEND nebo GRAPH zobrazil poslední editovaný řádek v editační zóně.

Další dvě opravené chyby se týkají rutiny tisku na obrazovku. Rutina odmítala řídicí kód BS (CHR\$ 8), byla-li aktuální pozice AT 1,0; stejný kód byl nesprávně zpracován při pozici AT 0,0. Kód CHR\$ 9 rutina vůbec ignorovala.

S novou ROMkou můžeme bez jakéhokoli nebezpečí používat příkaz CLOSE. V původní ROMce vedl pokus o uzavření již zavřené linky vyššího čísla ke zhroucení Spectra.

Časová prodleva mezi hlavičkou a programovým blokem u příkazu SAVE byla zkrácena asi na 100 ms.

Od verze 2.3 se již počítá s připojením ZX Interfacu 1 i přes to, že tento interface volá některé rutiny, které LECROM již nemá.

Nová inicializace

Po zapnutí nebo resetu Spectra s novou ROMkou proběhne jiná inicializace systémových proměnných. Nastaví se INK 7, PAPER 0, BRIGHT 1 a BORDER 0. Je zapnuto pípnání klávesnice a jsou zkráceny časové konstanty REPDEL a REPPER pro zrychlení editace.

Ihned nápadný je inovovaný test paměti RAM. Na rozdíl od původního je schopen odhalit chybu ve všech bitech paměti. Kontroluje zápis nul i jedniček.

Počínaje verzí 2.3 je zaveden i ROMtest, který je vhodné provádět při použití paměti EPROM (teoreticky mohou něco "zapomenout").

Každý si jistě všimne i nového generátoru znaků (původní nebyl příliš estetický). Při jeho vytváření jsem však vycházel z původního.

Po počáteční inicializaci je zobrazeno úvodní MENU, z něhož se vybírá stiskem tlačítka 1, 2 nebo 3:

- 1 Basic provede start Basic editoru
- 2 Monitor spustí malý monitor paměti
- 3 Boot vygeneruje příkaz RUN pro načtení programu "run" z prvního microdrivu.

1. Basic

První nápadnou změnou při vstupu do Basicu je kurzor, který je kurzorem v pravém slova smyslu. U starších verzí LECROM je to známé blikající "L".

Je pozměněna funkce CAPS SHIFT. V módu CAPS LOCK se písmeno napsané se stisknutým tlačítkem CAPS SHIFT zobrazí jako malé (tak, jak to dělají jiné rozumné počítače).

Nový Basic má vylepšenou editaci. V dlouhých řádcích lze pohybovat kurzorem nejen vodorovně, ale i svisle. Tím se editace podstatně zrychlí. Při přemístění kurzoru na začátek editované řádky (před její číslo) mají klávesy pro pohyb kurzoru nahoru a dolů stejnou funkci jako původně, tj. pohybují horním kurzorem.

Hlavní předností upraveného editoru je možnost zápisu klíčových slov po jednotlivých písmenech. Původní klávesnice s předem nadefinovanými celými příkazy nevyhovovala mnoha uživatelům, zvyklým pracovat s většími počítači. S LECROM lze psát malými i velkými písmeny, s mezerami mezi slovy i bez nich. Nutnou a postačující podmínkou pro správnou tokenizaci je, že před ani za klíčovým slovem nesmí být nějaké jiné písmeno. Po odeslání příkazu proběhne konverze textu na obvyklé tokeny Basicu. Někdo sice může namítnout, že je rychlejší příkaz PRINT zapsat pouhým stisknutím klávesy... Ale v nové úpravě odpadá zapeklité přepínání módů kurzoru, LECROM navíc umožňuje používat místo celých příkazů jen jejich zkratky. Stačí napsat začátek příkazu a ukončit jej tečkou. Přitom je lhostejné, na které pozici textu příkazu tečku napíšeme (jak dalece jej zkrátíme).

Kompletní seznam nejkratších možných zápisů všech basicových příkazů:

<=	<.	<>	nemá	>=	>.
ABS	ab.	ACS	ac.	AND	an.
ASN	as.	AT	nemá	ATN	nemá
ATTR	a.	BEEP	be.	BIN	b.
BORDER	bo.	BRIGHT	br.	CAT	ca.
CHR\$	ch.	CIRCLE	ci.	CLEAR	cle.
CLOSE #	cl.	CLS	nemá	CODE	c.
CONTINUE	con.	COPY	cop.	COS	nemá
DATA	da.	DEF FN	d.	DIM	di.
DRAW	dr.	ERASE	er.	EXP	e.
FLASH	fl.	FN	f.	FOR	nemá
FORMAT	fo.	GO SUB	gos.	GO TO	g.
IF	nemá	IN	nemá	INK	nemá
INKEY\$	i.	INPUT	inp.	INT	nemá
INVERSE	inv.	LEN	l.	LET	nemá
LINE	li.	LIST	lis.	LLIST	ll.
LN	nemá	LOAD	lo.	LPRINT	lp.
MERGE	me.	MOVE	m.	NEW	ne.
NEXT	nex.	NOT	n.	OPEN #	op.
OR	o.	OUT	ou.	OVER	ov.
PAPER	pa.	PAUSE	pau.	PEEK	pe.
PI	p.	PLOT	pl.	POINT	po.

POKE	pok.	PRINT	pr.	RANDOMIZE	ra.
READ	re.	REM	nemá	RESTORE	res.
RETURN	ret.	RND	r.	RUN	ru.
SAVE	sa.	SCREEN\$	s.	SGN	sg.
SIN	si.	SQR	sq.	STEP	ste.
STOP	sto.	STR\$	st.	TAB	t.
TAN	nemá	THEN	th.	TO	nemá
USR	u.	VAL	nemá	VAL\$	v.
VERIFY	ve.				

Tak např. řádek:

```
10 BORDER 7: PAPER RND*8: CLS : PRINT PEEK 23609:
DRAW 10,10
```

stačí napsat jako:

```
10 bo.7:pa.r.*8:cls:pr.pe.23609:dr.10,10
```

Komu tento způsob zápisu nebude vyhovovat, může nově zavedeným příkazem FORMAT "k" přepnout Spectrum do původního módu s obvyklým zápisem příkazu jedinou klávesou.

Když v módu vypisování jednotlivými znaky vyvoláte řádek do editační zóny, automaticky se provede jeho detokenizace (příkazy jsou zpětně rozšířeny na znaky).

Pokud nebyly použity nové příkazy, je program zapsaný v tomto módu plně kompatibilní s programem zapsaným v původní ROMce. Až na jednu výjimku, kterou je použití funkce VAL. Např.:

```
10 pr.val"sin(pi/2)"
```

bude zpracováno jako:

```
10 PRINT VAL "sin(pi/2)"
```

To znamená, že funkce vepsané v uvozovkách jsou považovány za text - takto zapsaný příkaz nebude v originální ROMce správně zpracován. Zde je třeba použít FORMAT "k" pro přepnutí na přímé příkazy, zadat příslušné funkce a tímtež příkazem se vrátit zpět do režimu vypisování po znacích.

Nové příkazy Basicu

Původní ROMka umí do programu zařadit řádky obsahující příkazy typu:

```
MOVE string,string
ERASE string
CAT (bez parametru)
FORMAT string
```

Ale snaha o jejich provedení je provázena chybovým hlášením "Invalid stream". Tyto příkazy jsou nově využity takto:

MOVE "r m,n","r k" - kopíruje blok paměti od adresy "m" o délce "n" na adresu "k". Příkaz velmi rychle přemísťuje zadané bloky, můžeme jim také zaplňovat část paměti nějakou hodnotou. Na místě "m", "n", "k" mohou být libovolné výrazy. Vyhodnocení provádí funkce VAL. Velmi výhodný je tento příkaz u Spectra s 80K RAM. Místo "r" lze užít písmeno "d" - pak se pracuje s dolními 32K paměti ve druhé stránce. Např. MOVE "r 16384,6912","d 0" uschová obrazovku Spectra do jakéhosi "RAMdisku". Celá paměť Spectra je pak volná a obrazovku lze pomocí MOVE "d 0,6912","r 16384" kdykoli získat zpět.

Od verze 2.3, která je schopna pracovat s pamětí až 528K, je syntaxe MOVE pozměněna takto:

MOVE "saa,ll","dbb" - kopíruje blok z paměti "s", počínaje adresou "aa", dlouhý "ll", na místo "d", od adresy "bb". Specifikace paměti pomocí znaků "s" a "d" je následující:

r - RAM (běžná pracovní paměť)

a,b,c,d,e,f,g,h,i,j,k,l,m,n,o - jednotlivé

stránky RAMdisku. Máme-li jen 80K RAM, má samozřejmě smysl mluvit jen o stránce "a", u Spectra 272K lze pracovat s "a" až "g".

Příklady:

MOVE "r16384,6912","r4e4" - zkopíruje VideoRAM na adresu 40000.

MOVE "r4e4,"2e3","c0" - zkopíruje pracovní paměť od adresy 40000 v délce 2000 bajtů do RAMdisku "c" na adresu 0.

Při používání příkazu **MOVE** je však třeba dát pozor na několik věcí. Především - není dobré si přepsat systémové proměnné nebo zásobník. Pro přenos bloku nelze pracovat s posledními 35 bajty RAMky, protože sem se před vlastním přesunem umísťuje přesunovací rutinka (původní obsah této oblasti zůstane ovšem zachován - před přesunem se uklidí do oblasti **WORKSPACE** a po provedení přesunu se vrátí zpět).

ERASE "i" - zruší vše, čím se projevuje ZX Interface 1. Smaže jeho kanály, vyhodí buffery pro microdrive a network a vypustí příslušné systémové proměnné. Linky 4 až 15 budou uzavřeny. Basicový program pak začíná na adrese 23755 jako u samotného Spectra. Příkazem **RUN** nahrajeme program "run" z prvního microdrivu. Pak budou fungovat i basicové programy se strojovým kódem umístěným v programových řádcích.

CAT - spustí vestavěný monitor (ovládání monitoru viz dále).

FORMAT "k" - slouží k přepínání funkce klávesnice. Po provedení příkazu naskočí původní kurzor "K" - klíčová slova zapisujeme stisknutím jednoho tlačítka. Opětným provedením tohoto příkazu se funkce klávesnice přepne do zápisu jednotlivými znaky.

Od verze 2.3 jsou zavedeny ještě další příkazy:

FORMAT "s" - vypíná a zapíná dotaz "scroll?".

COPY n - zkopíruje n bloků z kazety na kazetu. Pro kopírování je využit pouze pracovní prostor Basicu, takže se neztratí programy ve strojovém kódu. Není-li pod adresou **RAMTOP** dostatek místa, lze příkazem **CLEAR** prostor zvětšit.

2. Monitor

Velmi praktický a u Spectra dosud postrádaný je zabudovaný monitor. LECROM takový jednoduchý monitor obsahuje. Můžeme jej spustit třemi způsoby. Ze základního menu (po zapnutí počítače) nebo z Basicu příkazem **CAT** nebo zastavením běhu programu současným stiskem tlačítek **SYMBOL SHIFT**, **SPACE** a **B**. Posledně jmenovaný způsob pracuje i při přerušení a lze jím zastavit i programy ve strojovém kódu (při povoleném přerušení). Musejí ovšem zůstat zachovány důležité systémové proměnné, které monitor potřebuje. Ovládání monitoru bylo ve verzi 2.3 značně pozměněno, proto popíšeme každé samostatně.

Ovládání starší verze monitoru

Monitor se po spuštění ohlásí hvězdičkou a kurzorem "C". Povel monitoru se skládají z písmene a hexadecadického čísla, pokud jej ten který povel vyžaduje. K dispozici je celkem sedm povelů:

Madr - výpis paměti od zadané adresy. Každý řádek výpisu obsahuje adresu prvního zobrazovaného bajtu. Následuje šest hexadecadických hodnot (obsah šesti adres), na konci řádku je šest ASCII znaků, odpovídajících předcházejícím číslům. Znaky větší než 128 se vypisují jako znaky o 128 menší, řídicí kódy (menší než 32) se zobrazují jako tečka. Na konci každého řádku se výpis zastaví a čeká na stisk tlačítka. **SPACE** výpis ukončí, jiná libovolná klávesa spustí tisk dalšího řádku.

Sadr - změna obsahu paměti od zadané adresy. Monitor napíše aktuální adresu a její obsah. Buď zadáme nový obsah bajtu, nebo stiskem **ENTER**u původní obsah paměti ponecháme beze změny. Činnost

se ukončí stiskem tlačítka **SPACE**, následovaného **ENTER**em.

B - opuštění monitoru a skok do Basicu. Samozřejmě by neměl být poškozen obsah důležitých systémových proměnných.

Gadr - spuštění programu ve strojovém kódu od zadané adresy. Program je volán instrukcí **CALL**, takže instrukcí **RET** se můžeme vrátit zpět do monitoru. Před spuštěním programu jsou registry **AF**, **BC**, **DE**, **HL**, **IX** a **SP** naplněny svými standardními hodnotami, uloženými v oblasti **MEM** (viz dále). Po návratu z volaného programu zůstane obsah uvedených registrů zachován jakožto nové standardní hodnoty. Možnost číst a měnit obsah registrů CPU se hodí při ladění programů.

R - čtení a změna obsahu registrů CPU. Změna se pochopitelně netýká registrů přímo, pracuje se se standardními hodnotami. Monitor vypíše název registru a jeho původní obsah. Zadáním čísla se původní hodnota přepíše, prosté odeslání odřádkuje beze změny.

Dčíslo - převod hexadecadického čísla na dekadické.

Hčíslo - převod dekadického čísla na hexadecadické. Pozor! Vzhledem ke stísněnému prostoru v paměti používají povel **D** a **H** stejnou pracovní oblast jako povel **R**. Proto se může stát, že se provedením **D** nebo **H** změní standardní obsah registrů CPU.

Ovládání monitoru v2.3

Monitor se hlásí výpisem, v jehož horní části je obsah registrů (kurzor ukazuje na aktuální registr). Pod obsahem registrů je vždy vlevo adresa, za ní hexadecadický výpis od dané adresy a v pravé části jsou odpovídající ASCII kódy.

Příkazy monitoru se zadávají přes klasickou editovací řádku. To znamená, že lze v plném rozsahu využít editovacích schopností ROMky. **EDIT** zruší celý řádek.

Vlastní příkazy:

Q - Quit - skok do BASICu

Sxxxx - Search - vyhledávání slova xxxxH

Sxx - Search - vyhledávání bajtu xxH

Gxxxx - Gosub - **CALL** adresy xxxxH s návratem do monitoru (před provedením tohoto příkazu monitor nastaví obsah registrů dle jejich zobrazených hodnot)

L - List - rychlé listování dopředu

R - Reverse - couvnutí o bajt dozadu

Mxxxx - Modify - změna nastavení kurzoru do paměti

H***** - Hex - modify s dekadickou hodnotou adresy *****

Další možností je změna obsahu paměti prostým vložením požadované hodnoty bajtu do adresy, na kterou právě ukazuje kurzor (potvrzuje se stiskem tlačítka **ENTER**). V případě, že do editovací řádky není nic vloženo, může **ENTER** sloužit i k posunu o jednu adresu dopředu. Nastavení kurzoru u registrů se provádí tlačítky **Caps Shift + 6**. Tato funkce je totožná se stiskem tlačítka **ENTER** (vložením hodnoty a stiskem **Caps Shift + 6** se změní obsah daného registru).

Závěr

V LECROM je proti původní ROMce řada úprav a změn (jak doufám, k lepšímu). Aby byla zajištěna co největší kompatibilita s programy, které volají podprogramy z ROMky, zůstaly všechny rutiny původní ROMky na svých obvyklých adresách. Nové rutiny byly různě propleteny do všech volných skulinek. Nebyla to práce jednoduchá, takže nelze vyloučit, že přese všechnu pečlivost se mohla vloudit drobná chybička. Pokud nějakou objevíte, sdělte mi to, prosím, prostřednictvím redakce Mikrobáze.

Jiří Lamač

PROGRAMOVÁ KOMPABILITA POČÍTAČŮ DIDAKTIK GAMA A ZX SPECTRUM

Možnost využití bohatého softwaru ZX Spectra je nespornou výhodou pro uživatele mikropočítače Didaktik Gama výrobního družstva Služba Bratislava, závod Didaktik Skalica. V ojedinělých případech však můžete narazit i na programy, které na počítači Didaktik Gama nebudou fungovat. Jde o případy, kdy program pro ZX Spectrum pracuje s určitými úseky paměti ROM, které byly v Didaktiku Gama změněny. Porovnáním paměti ROM obou mikropočítačů zjistíme rozdíly v rozsahu cca 300 bajtů. Podívejme se, o jaké funkční změny jde:

1. Mikropočítač Didaktik Gama je vybaven paralelním interfacem s obvodem MHB 8255. Na adresách 04AAH-04C1H Didaktiku Gama je rutina, která zajišťuje inicializaci obvodu 8255 a určuje délku řádku na tiskárně na 80 znaků. Tato část je volána z adresy 129FH po vynulování celé paměti RAM. Na adresách 04AAH-04C1H ZX Spectra je nepoužívaný podprogram monitoru (viz /1/).

2. Při inicializaci systému Didaktiku Gama se na obrazovce vypíše text "(c) 1987 Didaktik Skalica", který je uložen na adresách 1539H-1554H, na rozdíl od původního textu "(c) 1982 Sinclair Research Ltd." ZX Spectra.

3. Z chyb, které obsahuje romka ZX Spectra, jsou v Didaktiku Gama opraveny tyto:

a) Číslo -65536 je interpretováno správně. U ZX Spectra dává příkaz PRINT INT -65536 výsledek -1, u Didaktiku Gama je výsledek správný.

b) Je opravena chyba "CHR\$ 9", tj. "kurzor doprava". V ZX Spectru příkaz PRINT CHR\$ 9 nefunguje, v Didaktiku Gama ano.

c) V rutině ošetřující nemaskovatelné přerušení (NMI) byl na adrese 6DH opraven chybný relativní skok, takže rutina v Didaktiku Gama funguje.

Ostatní chyby romky ZX Spectra, popsané v /1/ se projevují i v Didaktiku Gama.

4. V romce ZX Spectra je od adresy 386EH volná oblast. V Didaktiku Gama je na adresách 386EH-390AH obsluhována rutina pro tiskárnu, která se připojuje přímo k obvodu 8255. V tabulce kanálových adres (15BEH-15BFH) je v Didaktiku Gama provedena změna adresy pro výstup znaku, a to právě na 386EH. Tato rutina je rovněž využívána příkazem LLIST. Na adrese 17F5H Didaktiku Gama je změněna adresa skoku funkce LLIST na 38F6H. Proto pozor! Pokud není připojena tiskárna, dostane se Didaktik Gama při výkonu příkazů LPRINT a LLIST do věčné smyčky, ve které čeká na potvrzení připrave-

nosti tiskárny! Kontakt s počítačem lze obnovit pouze novou inicializací systému.

Nová funkce rutin pro tiskárnu v romce Didaktiku Gama vadí při načítání některých programů ZX Spectra z magnetofonu. Jde o programy, které používají přepnutí výpisu znaků na tiskárnu k tomu, aby se hlavičky dalších bloků nevypisovaly na obrazovce. Didaktik Gama při přepnutí čeká na potvrzení připravenosti tiskárny, a tak načítání programu nepokračuje. Úprava takového programu pro Didaktik Gama spočívá ve vynechání instrukce POKE 23578,16 v té části programu která řídí čtení dalších částí. Program pak lze normálně načíst do paměti Didaktiku Gama s tím, že v úvodním obrázku se při nahrávání programu budou objevovat názvy načítaných bloků. Tak lze pro Didaktik Gama upravit např. programy Thanatos a Green Beret.

5. Do oblasti romky, která není v ZX Spectru využita, je v Didaktiku Gama uložena rutina pro kopírování systémového zásobníku. Paměť RAM Didaktiku Gama je rozdělena na základních 16K a dvě paměťové banky po 32K. Banky se přepínají instrukcí OUT 127,0 nebo OUT 127,1 podle toho, se kterou bankou chceme pracovat. Při přepnutí je vždy kopírován systémový zásobník z jedné banky do druhé. Tato rutina je v romce Didaktiku Gama na adrese 390BH. V tabulce adres jednotlivých příkazů Basicu (na adrese 1AF3H v Didaktiku Gama) je adresa původní instrukce OUT změněna právě na 390BH.

Závěrem lze říci, že programová kompatibilita Didaktiku Gama se ZX Spectrem je velmi dobrá. S ohledem na popsané rozdíly mezi paměťmi ROM obou počítačů by problémy s chodem programů ZX Spectra na Didaktiku Gama měly být minimální.

R. Gemrot

Literatura

- /1/ M.Derian: Spectrum Monitor and Basic Interpreter (AR 10/85, str.383)
- /2/ Didaktik Gama, návod k obsluze
- /3/ K.Kurylowicz a spol.: Przewodnik po ZX Spectrum (WkiL 1986)
- /4/ B.Dědina, P.Valášek: Mikroprocesory a mikropočítače (SNTL 1983)

ROZŠÍŘENÍ PAMĚTI ZX SPECTRA (2)

Další materiál o ZX-Sp 80K je rozdělen do kapitol:

- 1) Koncepce rozvoje ZX-Sp 80K
- 2) Informace o podobných zapojeních s CP/M
- 3) Jak psát programy pro ZX-Sp 80K
- 4) Microdrive a ZX-Sp 80K
- 5) Způsob odstránkování 64K RAM u ZX-Sp 80K
- 6) (e) STD BUS
- 7) Posilovač sběrnice
- 8) Řadič floppy disku
- 9) Stručný popis BIOSu
- 10) Přehled programů nahraných na kazetě č.2
- 11) Literatura

ad 1) Koncepce rozvoje ZX-Sp 80K

Každý uživatel má vůči svému zařízení jiné nároky. Proto je opodstatněná otázka, zda existuje nějaké zapojení či typ počítače, který by uspokojil všechny zájemce nebo aspoň jejich velkou část. Pochopitelně nejde jen o samotný počítač, který bez programů k ničemu není (a naopak).

Výhodou programů je, že mohou vznikat dávno po sestavení počítače. Jejich vývoj může sledovat potřeba uživatelů. Optimální by bylo, kdyby se i hardware počítače mohl časem měnit tak, aby byl v souladu s požadavky uživatele. Podobné myšlenky možná daly vzniknout stavebnicové koncepci počítače IBM PC. Koncepce rozvoje ZX-Sp 80K je podobná (viz obr. 1/1).

Uživatelé se obvykle ubírají jednou ze dvou hlavních cest:

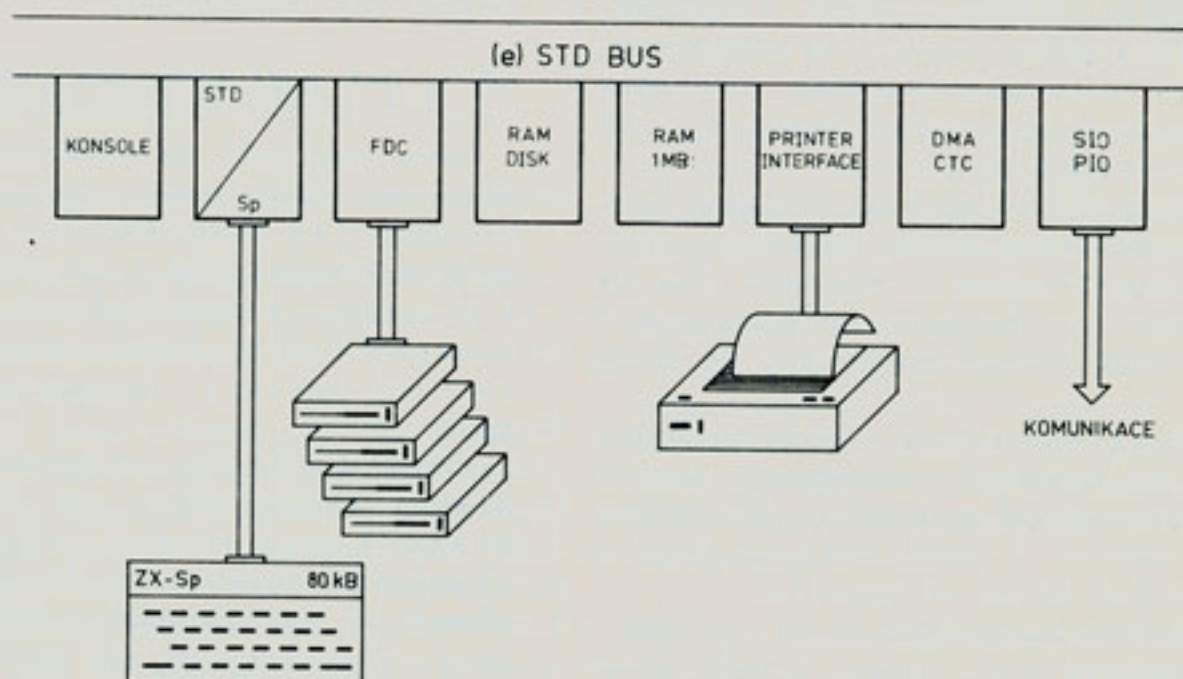
a) Uživatel nechce pracovat se žádným operačním systémem, ale chce využívat výhod programového vybavení ZX-Sp 80K. Postaví si tedy jen základní verzi. Pro záznam používá nadále magnetofon. Podle dalších potřeb může do Spectra vestavět (nebo připojit na sběrnici) Cracker. S touto konfigurací je pak schopen využívat veškeré programové vybavení pro ZX-Sp 80K (kromě diskové implementace operačního systému CP/M), tedy i

verzi CP/M pro magnetofon či microdrive. Pro práci s magnetofonem nemá smysl vybavovat Spectrum větší operační paměti než 80K (přenosová rychlost okolo 11K za minutu). Např. při paměti 256K zabere záznam a jeho verifikace kolem 50 minut za předpokladu, že během nahrávání nedošlo k chybě přenosu.

b) Uživatel chce využívat všech možností, které mu výpočetní systém s osmibitovým procesorem poskytuje, tedy včetně profesionálních programů a nepřeberné řady periférií. Protože Spectrum se po elektrické stránce pohybuje na hranicích svých možností, je třeba je vybavit posilovačem sběrnice. Při posílené sběrnici je vhodné zajistit kompatibilitu na základě nejrozšířenější světové normy pro připojení periférií na 8-bitovou datovou sběrnici - STD BUS. K němu lze připojit jednotlivé periferní desky - např. řadič floppy disku, DMA, stránkovanou paměť, ramdisk, CTC, komunikační kartu s PIO a SIO, univerzální porty s PPI, A/D a D/A převodníky a mnoho dalších desek pro speciální aplikace. Řada těchto desek je navržena, některé jsou již funkční, ostatní jsou vyvíjeny. Na základě uvedené sběrnice lze použít velké množství profesionálních desek různých firem, desky z Mikro-AR či vyvinout desky vlastní, které by ovšem měly být kompatibilní se Spectry i jinými počítači. Tak si každý bude moci postavit takovou konfiguraci, jaká mu bude vyhovovat. A přitom zachová svou kompatibilitu s ostatními.

Pomocí speciální přizpůsobovací karty je možné zajistit kompatibilitu se Spectry rozšířenými podle verzí různých autorů, a to i když využívají větší paměť než 80K (272K, 592K a více). To je ovšem řešení poněkud zbytečné, protože uvedená zapojení hardwarově ani softwarově nejsou schopna nabídnout to, co úprava podle ST 11/87.

Obrázek 1/1 znázorňuje možnou konfiguraci ZX-Sp 80K. Spectrum je ke sběrnici (e) STD připojeno přes její oddělovač. Vlevo od posilovače je umístěna konzole, připojená k portu 7BH. V ní je



Obr. 1/1

Cracker (u těch, kteří jej nemají zabudován přímo ve Spectru) a BOOT EPROM pro zavedení libovolného operačního systému z disku. Kromě toho plní řadu dalších funkcí.

FDC - viz kapitola 8

RAM DISK - použitelný jako jeden z disků operačního systému CP/M; leží na portech 0 a 1

RAM 1 MB - stránkovaná paměť RAM; dosud ve vývoji

PRINTER INTERFACE - pro připojení tiskárny a jiných periférií

DMA a CTC - karta s uvedenými obvody; dosud ve vývoji

SIO a PIO - univerzální komunikační karta

Minimální konfiguraci pro práci s CP/M tvoří ZX-Sp 80K, posilovač sběrnice a FDC; nebo FDC upravený k přímému připojení ke sběrnici ZX-Sp 80K. Podmínkou není ani konzole. BOOT lze před začátkem práce načíst z magnetofonu (necelých 400 bajtů).

Podmínky kompatibility:

Předkládané informace je třeba chápat pouze jako doporučení. Každý si může provést takové úpravy, jaké mu budou vyhovovat. Chce-li přesto zůstat kompatibilní se vznikajícím softwarem, měl by se držet těchto zásad:

- přepínání při OUT (FFH), A
- zpoždění o jeden cyklus M1
- dodržení zapojení konektorů ZX-Sp 80K a (e) STD - viz kap.6
- nepoužívat porty k jiným než vyhrazeným účelům (0,1,10-17H,7DH)

ad 2) Informace o podobných zapojeních s CP/M

Na ZX-Sp 80K byl implementován operační systém CP/M v diskové verzi s řadičem WD 2797 pro mechaniky 8", 5.25", 3.5" a rovněž s microdrivem a elektricky ovládaným magnetofonem (SP 210, SM 260, SM 261). Posledně uvedená implementace nebyla dosud zdokumentována. V diskové verzi lze zvolit fyzický formát záznamu dat. Buď klasický (odpovídající standardu CP/M), nebo formát kompatibilní např. s počítači ROBOTRON 1715, 5120, 5130, který používá sektor delší než 128 bajtů.

Z hlediska srovnání je nejzajímavější produkt J.Lamače, který nabízí Mikrobáze. Je třeba podotknout, že nejde o operační systém CP/M, ale MIKROS, což je československá verze systému CP/M firmy Digital Research. Od klasické CP/M se liší v tom, že není využíván (implementován) I/O bajt na adrese 3. Z toho důvodu některé systémové programy nepracují správným způsobem.

I přes tuto odchylku od normy lze hodnotit implementaci systému "CP/M" na microdrive jako velmi dobrou. Na žádost řady uživatelů Spectra byl tento systém implementován začátkem roku 1988 i na ZX-Sp 80K. Vyhovělo se tak nejen uživatelům ZX-Sp 80K, ale i propagátoru verze podle J.Lamače, Danielu Mecovi, který přímo vyzývá ke kompatibilitě a implementaci uvedeného systému i na jiné počítače - viz klubový zpravodaj Spectra 1/88 (602.ZO Svazarmu) a rovněž AR/A 9/88.

Pro uživatele je potěšitelné, že vzájemná kompatibilita je tak zajištěna nejen na úrovni operačního systému, ale dokonce i záznamu dat na cartridge. Znamená to, že data zaznamenaná v úpravě podle AR/A 9/88 lze přímo použít na ZX-Sp 80K.

Naskýtá se tedy otázka, které z obou zapojení zvolit. Chceme-li se správně rozhodnout, musíme posoudit jejich hardwarové možnosti. Mikrobáze spolu AR/A 9/88 nabízí i rozšíření na 272K a 528K. A vše je umístěno přímo ve Spectru:

1) Aby byla zajištěna spolupráce s IF 1, tedy i s microdrivem, je adresa A15' (resp. A15GE) vyvedena na pin, který náleží adresovému vodiči A15. Znamená to, že v módu, kdy je v adresovém prostoru procesoru připnuta paměť RAM od adresy 0-64K, není

přístupná adresa A15. To způsobuje nekompatibilitu hranového konektoru Spectra a obecně znemožňuje rozšíření systému o další periférie. ZX-Sp 80K to řeší jiným způsobem (viz kap.4: Spolupráce microdrivu se ZX-Sp 80K).

2) V obvodu přepínání paměti je dekódován jediný adresový vodič. To v adresovém prostoru 0-256 způsobuje zrcadlení stránkovacího obvodu 127x. Je tu použito lineární dekódování. To znemožňuje rozumné adresování dalších periférií nad rámec klasického Spectra (8 periférií). U ZX-Sp 80K je věc řešena opět jiným způsobem (viz kap.7).

3) Není implementován signál RAMS jako u ZX-Sp 80K. Proto nelze zajistit přímý přístup k neomezeně velké paměti libovolného typu jako u ZX-Sp 80K, ale maximálně k 528K dynamické paměti RAM (18).

Z toho plyne závěr, že kdo chce používat operační systém "CP/M" s microdrivem a využívat přitom paměť 272K nebo 528K, která zvyšuje komfort obsluhy a je vestavěna přímo do Spectra, lze mu plně doporučit úpravu podle Mikrobáze či AR/A 9/88. Tuto úpravu, optimalizovanou pro použití microdrivu, si lze objednat u Mikrobáze.

Pro toho, kdo chce využívat operační systém "CP/M" s microdrivem a přitom předpokládá rozšíření svého systému pro náročnější aplikace, je vhodnější systém implementovaný na ZX-Sp 80K. Po stránce fyzické přenositelnosti dat jsou obě implementace shodné. "CP/M" s microdrivem a 80K pamětí je plně funkční, z hlediska komfortu a rychlosti však nikoli optimální. To ale neplatí o verzi diskové. ZX-Sp 80K bylo optimalizováno právě pro diskovou verzi. Jako stručný popis BIOSu může zájemcům o verzi "CP/M" pro microdrive sloužit již zmíněný článek v AR/A 9/88 (18).

Pokud se někomu uvedené srovnání zdá neobjektivní, můžeme mu po poskytnutí veškeré informace a prostředky k nestrannému porovnání obou implementací (adresa viz Mikrobáze 10/88).

Další implementací operačního systému CP/M je zapojení podle publikace 666.ZO Svazarmu. Zde se zdrojový text BIOSu rovněž odchyluje od normy CP/M. Z toho důvodu se některé programy chovají nestandardním způsobem. Obvodové řešení řadiče včetně separátoru dat je klasické (podle hardwarového manuálu firmy INTEL). Při jiném zapojení a vynaložení stejných finančních nákladů se to dá řešit lépe. Z hlediska současných poznatků není použití obvodu I 8272 pro stavbu řadiče operačního systému CP/M optimální. Odpůrci mohou namítnout, že většina počítačů IBM PC a jeho klony uvedený řadič používají. To ovšem neříká nic o vhodnosti použití tohoto řadiče v našich podmínkách.

Existuje ještě několik ojedinělých implementací operačního systému CP/M, používajících jiná zapojení stránkovacích obvodů. Pro jejich malé rozšíření je nebylo možné vyzkoušet a provést jejich objektivní zhodnocení.

ad 3) Jak psát programy pro ZX-Sp 80K

Tato kapitola se nevěnuje popisu programovacích technik. Chce pouze upozornit na to, že v řadě případů lze programy psát takovým způsobem, aby byly aplikovatelné i na obyčejném Spectru 48K. Jde o programy, u nichž převažuje prostor pro data nad vlastní řídicí částí. Mezi takové programy patří textové editory, databanky, slovníky, kopíráky apod.

Když u těchto programů umístíme zásobník a hlavní řídicí část nad 32K, pak stačí provést test, který zjistí, zda je program umístěn v klasickém Spectru či ZX-Sp 80K. V systémové proměnné si poznamenáme, odkud je volná paměť pro data. Při testu na 80K přestrávkujeme a snažíme se zapsat informaci do spodní části paměti. U klasického Spectra je tam i po přestrávkování romka, do níž zapsat nelze. Z toho vyplývá, že u klasického

Spectra lze v programu použít stránkovací mechanismus, i když jej Spectrum ignoruje. Teprve u rozšířeného Spectra je funkční.

U ZX-Sp 80K nesmíme zapomenout ošetřit přerušeni - při ukládání dat od adresy 0 přerušeni zablokovat a po přestránkování jej opět povolit (aby mohla být čtena klávesnice); nebo data ukládat od adresy vyšší než 38H a přerušeni ošetřit jiným způsobem.

stačí v horní části paměti vytvořit malý buffer a podle okamžité potřeby do něj z romky přenést tělo momentálně aktuální rutiny se zmodifikovanými absolutními adresami.

Když při tvorbě programů přihlédneme k uvedeným pravidlům, nově vzniklé programy budou aplikovatelné na klasickém Spectru i ZX-Sp 80K.

ad 4) Microdrive a ZX-Sp 80K

Někteří odpůrci zapojení ZX-Sp 80K (obvykle autoři jiných zapojení) mu vytýkali nemožnost spolupráce s microdrive. Použijeme-li paměť RAM od adresy 0 a současně i microdrive, který pro tuto konfiguraci Spectra nebyl koncipován, dojde ke kolizi dat na sběrnici. Spectrum komunikuje s IF1 pomocí instrukce RST 8. Předá-li procesor řízení na adresu 8, ohlásí se IF1. Znamená to, že se signálem ROMCS odepne ROM Spectra a do jejího adresového prostoru se přistránuje 8K ROM IF1. Některí uživatelé ZX-Sp 80K vymýšleli poměrně složité obvody pro dekódování signálu ROMCS. Jejich řešení obvykle přinášela částečnou nekompatibilitu se stávající normou ZX-Sp 80K - buď na systémovém konektoru, nebo softwarovou. Uvedený problém byl nucen řešit i J.Lamač. Vyřešil to tak, že pro ovládání IF1 nepoužívá jeho romku, ale vlastní rutiny.

U ZX-Sp 80K lze použít microdrive dvěma způsoby, které se navzájem vylučují:

a) Když chceme použít IF1 pro práci s operačním systémem CP/M, IF1 se nesmí hlásit na adrese 8, máme-li přestránkováno do módu 64K.

b) Když chceme IF1 použít v módu Cracker, kdy simulujeme činnost paměti EPROM, musí při inicializaci IF1 na adrese 8 dojít k odstránkování ROM Spectra (to zajistí IF1) a rovněž k odstránkování (odepnutí) paměti RAM. Po připojení ROM IF1 by jinak došlo ke kolizi s pamětí RAM na datové sběrnici.

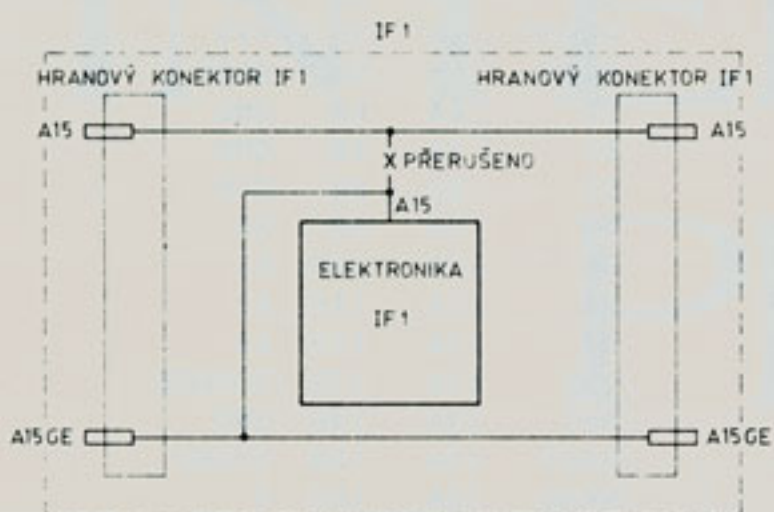
Řešení problému je jednoduché:

ad a) Na ovládací elektroniku IF1 přivedeme místo adresy A15 adresu A15GE (viz obr.4/1). Zapojení vstupního i výstupního konektoru IF1 zůstává beze změny. Tak se zachová kompatibilita se všemi perifériemi, které lze k IF1 připojit. V případě, že by takto upravený IF1 bylo třeba používat i na neupraveném Spectru, musíme adresy A15 a A15GE propojit odporem. Není to řešení zcela čisté. Lepší je použít jedno hradlo OR, které vytvoříme např. ze dvou tranzistorů a jednoho odporu (viz obr.4/2). Tato úprava umožňuje nahrávat programy pro ZX-Sp 80K z microdrivu, včetně CP/M. V (18) je popsána úprava podle J.Lamače, která využívá podobnou implementaci CP/M pro microdrive, bez nutnosti provádět v IF1 nějaké dodatečné úpravy. Toto řešení však jde na úkor kompatibility s hranovým konektorem standardního Spectra. Adresa A15GE (resp. A15' podle (18)) je vyvedena na vývod adresy A15. Proto v módu 64K není na hranovém konektoru adresa A15 přístupná.

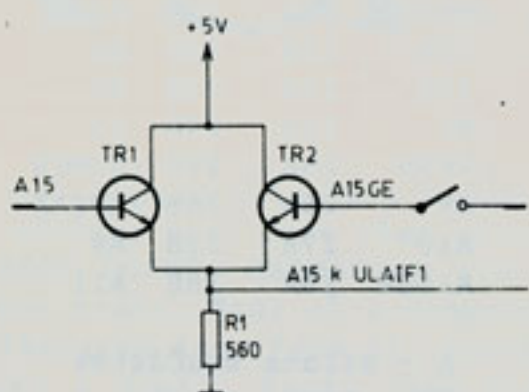
ad b) Z IF1 vyvedeme signál RAMS (RAM Select, vývod č.18B) na stejnojmenný signál konektoru Spectra (viz kap.5). Signál RAMS IF1 se vyvede přes spínač z odporu R32 (uvedeno na plošném spoji IF1), asice z konce, který je blíže systémovému konektoru (viz obr.4/3). V případě, že signál RAMS ZX-Sp 80K není realizován obvody LS, MOS apod., ale hradlem TTL, je nutné výstup signálu RAMS z IF1 oddělit emitorovým sledovačem nebo LS hradlem AND. Spínačem je přerušen adresový vodič A15GE v IF1. Jeho ovládním se volí druh provozu IF1 na ZX-Sp 80K.

ad 5) Způsob odstránkování 64K RAM u ZX-Sp 80K

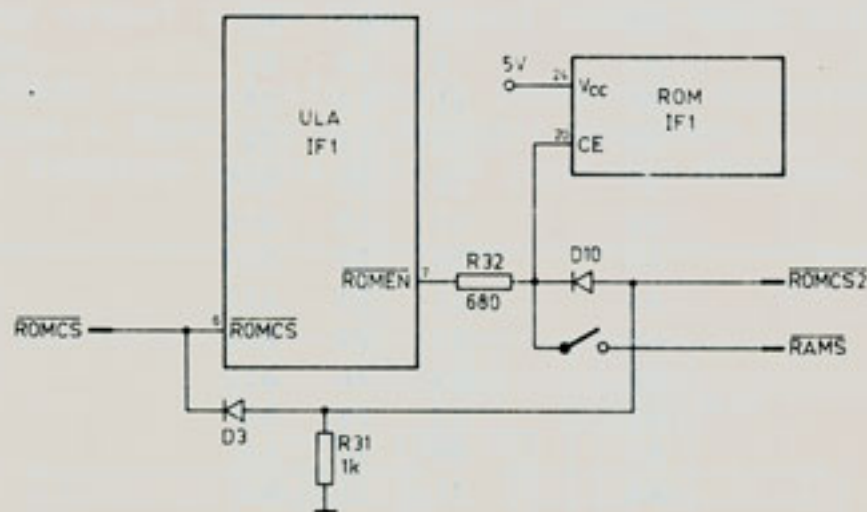
Chceme-li využít větší paměť než 80K, musíme dokázat odpojit (odstránkovat) paměťové obvody 64K a místo nich připojit libovolně velkou vnější



Obr. 4/1



Obr. 4/2 Úprava IF1 pro spolupráci se ZX Spectrem 80K

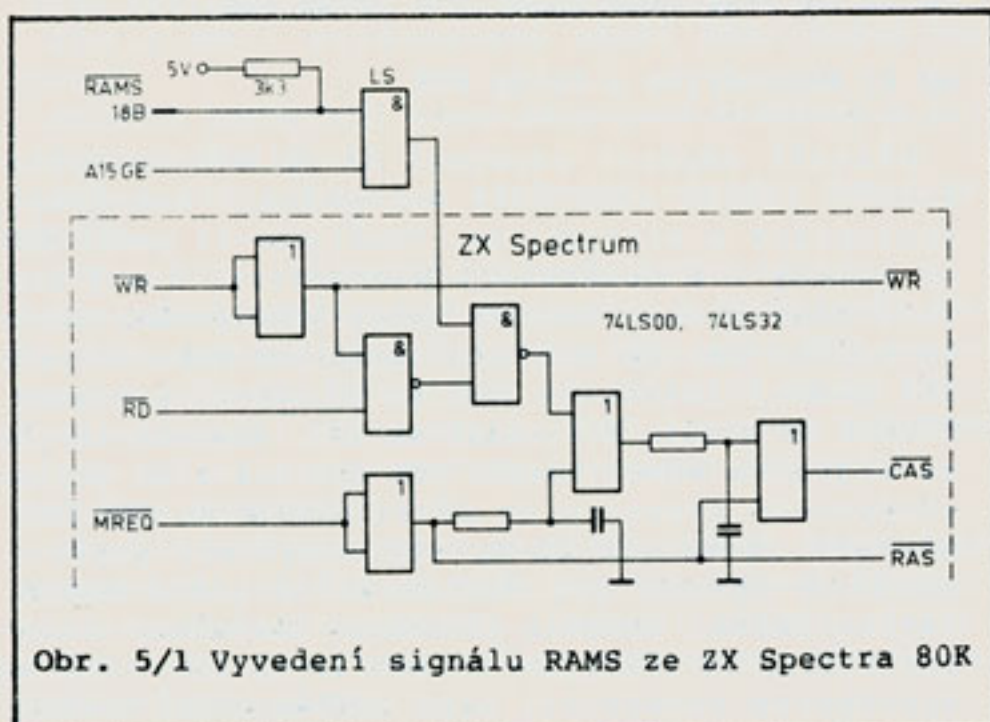


Obr. 4/3 Vyvedení signálu RAMS z IF1

16K video stránky lze využít různými způsoby. Většinou tam bude basicová část programu. V některých případech sem můžeme umístit třeba buffer pro třídící algoritmy apod.

Rozšířit místo pro data můžeme tak, že program napsaný ve strojovém kódu rozdělíme do dvou částí. Část, která se stará o čtení klávesnice, zobrazení a různé tabulky, může být ve video stránce, část pracující přímo s daty bude uložena od nejvyšších adres směrem dolů. Rovněž rutiny pro spolupráci s magnetofonem musejí být ve společné části s daty. Protože pro tyto operace nelze přímo použít rutiny romky (a mít v horní části paměti všechny rutiny pro LOAD, SAVE, M63RGE a VERIFY, je neefektivní),

paměť. Při práci se ZX-Sp 80K se s touto možností počítá. Proto byl na sběrnici Spectra vyveden signál RAMS (RAM Select). Vlastní zapojení je velmi jednoduché. Potřebujeme jedno hradlo AND, či jiná, která uvedenou funkci splní. Zapojení je na obrázku 5/1.



systemů musíme jeho signály posílit. Pro posílení signálů systémového konektoru ZX-Sp 80K byl navržen posilovač sběrnice.

Pro úplnost uvádím zapojení konektoru ZX-Sp 80K a sběrnice (e) STD. Doporučuji projít si jednotlivé signály a porovnat s literaturou (19).

Zapojení konektoru ZX-Sp 80K

A15	1A	1B	A14
A13	2A	2B	A12
D7	3A	3B	+5V
/RESFF	4A	4B	+9V
SLOT	5A	5B	SLOT
D0	6A	6B	GND
D1	7A	7B	GND
D2	8A	8B	CLK
D6	9A	9B	A0
D5	10A	10B	A1
D3	11A	11B	A2
D4	12A	12B	A3
/INT	13A	13B	/IORQGE
/NMI	14A	14B	GND
/HALT	15A	15B	VIDEO
/MREQ	16A	16B	Y
/IORQ	17A	17B	V
/RD	18A	18B	/RAMS
/WR	19A	19B	/BUSRQ
-5V	20A	20B	RESET
/WAIT	21A	21B	A7
12V	22A	22B	A6
12V	23A	23B	A5
/M1	24A	24B	A4
/RFSH	25A	25B	/ROMCS
A8	26A	26B	/BUSAK
A10	27A	27B	A9
A15GE	28A	28B	A11

A - strana součástek
B - strana spojů

Obr. 5/1 Vyvedení signálu RAMS ze ZX Spectra 80K

Zapojení konektoru (e) STD

+5V	1	2	+5V
GND	3	4	GND
	5	6	-5V
D3	7	8	D7
D2	9	10	D6
D1	11	12	D5
D0	13	14	D4
A7	15	16	A15
A6	17	18	A14
A5	19	20	A13
A4	21	22	A12
A3	23	24	A11
A2	25	26	A10
A1	27	28	A9
A0	29	30	A8
/WR	31	32	/RD
/IORQ	33	34	/MEMRQ
/IOEXP	35	36	/MEMEX
/RFSH	37	38	/(MCSYNC) /M1
/A15GE / (STATUS1)	39	40	/(STATUS0) /HALT
/BUSAK	41	42	/BUSRQ
/INTEX / (INTAK)	43	44	/INTRQ
/WAITRQ	45	46	/NMIRQ
/SYSRES	47	48	/PBRESET
CLOCK	49	50	/CNTRL
IEO (PCO)	51	52	(PCI) IEI
AUXGND	53	54	AUXGND
+12V	55	56	-12V
/DMARQ (BAO)	57	58	/BAI
A17	59	60	A19
A16	61	62	A18

Vysv.: / před signálem znamená negaci

Princip odstránkování je následující:

Paměťový blok 64K je ovládán obvodem 74LS00 a 74LS32, do nichž vedou signály MREQ, RD, WR a A15GE, který je využit jako pozitivní výběrový signál. Tyto obvody generují signály RAS a CAS. Podle stavu signálu MREQ, RD, WR a A15GE je paměť buď aktivní (A15GE=H), nebo neaktivní (A15GE=L) s probíhajícím cyklem refresh. Vytvoříme-li pomocí jednoho hradla AND elektronický přepínač (vstupy A15GE a RAMS, výstup hradlovaná A15), adresa A15GE se dostane ze vstupu tohoto hradla na výstup a Spectra se bude chovat známým způsobem. Jestliže vstup RAMS na sběrnici uzemníme, bude na výstupu hradla AND úroveň L. V důsledku toho bude blok paměti 64K neustále vykonávat cyklus refresh a paměť se nebude na sběrnici hlásit. V tomto okamžiku lze přistránkovat vnější paměť.

Signál RAMS je na vývodu č.18B konektoru Spectra, tedy na rozdílovém barvonosném signálu U. Ten je třeba přerušit a připojit signál RAMS. Důvodem, proč byl vybrán právě signál U, je skutečnost, že jej nebylo možno v praxi využít, protože je značně zarušen samotným Spectrem. Tento signál nevyužívá žádná profesionální periférie. Připojením jakékoli nestandardní periférie, která by uvedený signál využívala, nedojde ke zničení periférie, ani Spectra.

Poznámka: Na obrázku 5/1 je vývod RAMS připojen přes odpor 3k3 na +5V. Vlivem velkého vstupního odporu hradel LS může někdy dojít k nestabilitě; hodnotu odporu pak bude nutno snížit (např. na 1 k).

ad 6) Sběrnice (e) STD

Máme-li mikropočítačový systém, který není uzavřený, můžeme k němu připojit řadu periférií podle konkrétních potřeb uživatele. Tak z jednoduchého mikropočítače vytvoříme univerzální počítač pro různé oblasti použití.

Abychom mohli použít doplňky (periférie) různých výrobců, musíme se dohodnout na jednotné normě způsobu připojení. Takovou normou je (e) STD, která definuje rozložení signálů na konektoru, jejich význam, zatížení a způsob použití. Je vhodná pro mikropočítačové systémy s osmibitovou datovou sběrnici s možností přímého adresování až 1 MB paměti.

Norma byla popsána v Amatérském radiu (19-25). Je uzpůsobena pro připojení 8 desek (pozic) malého evropského formátu (100 x 160 mm). Protože srdcem každého mikropočítače je mikroprocesor, který ale nemůže budit libovolný počet obvodů, u větších

(pokračování)
Ing.Ladislav Sieger

TURBO INTERFACE PRO ZX SPECTRUM

V AR 9/87 byl popsán interface Kempston vhodný pro ovládání většiny her. Jsou však velmi zajímavé hry (např. MACH POINT, BARBARIAN atd.), které mohou hrát dva hráči. Tady už s Kempstonem nevystačíme. Potřebujeme interface II, který je u nás těžko dostupný a drahý. Proto jsem hledal jiné řešení.

Velmi zajímavé řešení jsem našel v /1/. Vychází z jednoduché úvahy - při stavbě interfacu potřebujeme takový IO, který má 10-11 vstupů a 8 výstupů. Tuto funkci může zabezpečit vhodně naprogramovaná paměť EPROM. Z hlediska dostupnosti a ceny vyhoví i MHB 2716, který vyrábí TESLA Piešťany.

TURBO rozezná tři druhy joysticků: Kempston, Protek a joysticky pro Sinclair Interface II.

- Kempston načítá bity od joysticku příkazem IN 31

- Protek používá řízení kurzory (5-7) a střelbu 0

- Interface II umožňuje připojit dva joysticky, přičemž první ovládá první polovinu kláves horní řady (1-5) a druhý ovládá druhou polovinu (6-0)

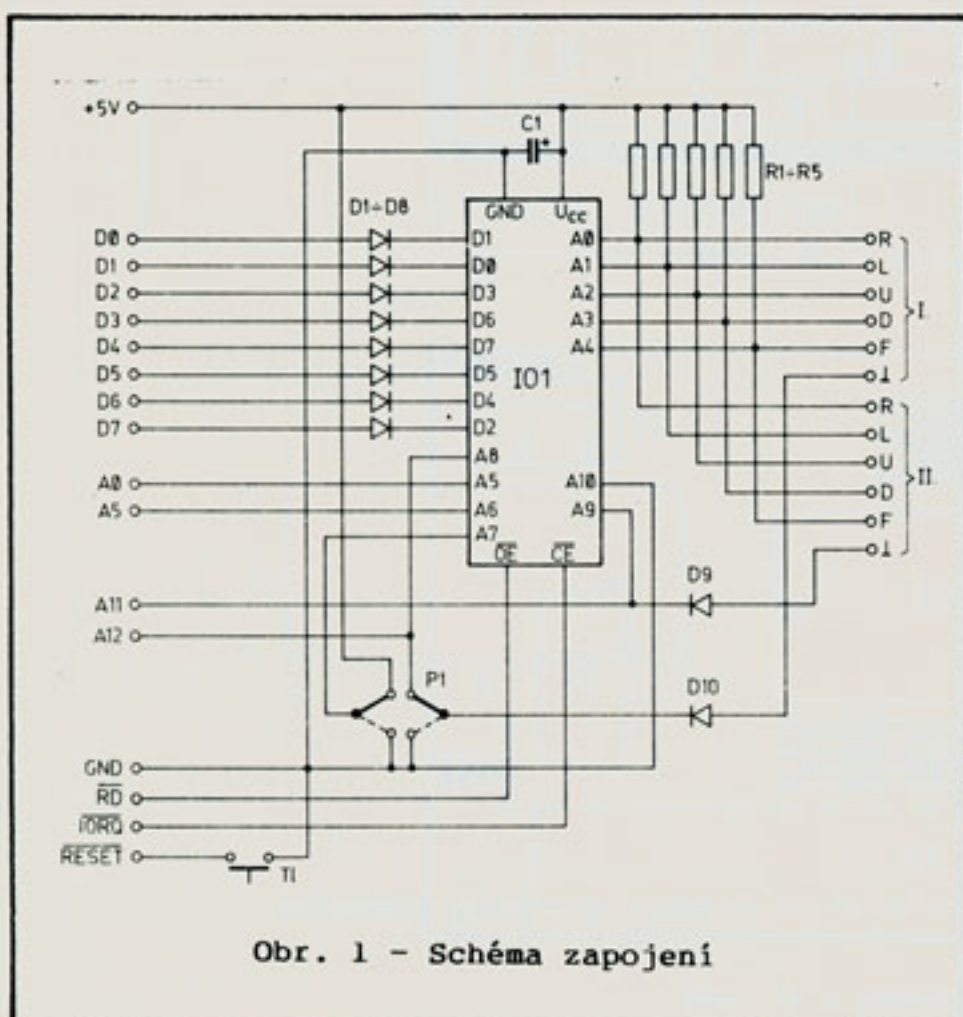
Vidíme, že Protek a Interface II se překrývají. Tento problém je možné odstranit přepínačem, jehož jedna polovina přepíná jeden vstup paměti EPROM do stavu 0 nebo 1 a druhá polovina přepíná společný bod prvního joysticku na GND nebo na A12 (obr. 1). V první poloze přepínače funguje interface jako Kempston a Interface II, ve druhé poloze jako Protek. Kromě toho je na vstup připojena A0, která adresuje klávesnici, A5 adresuje Kempston, A11 a A12 adresují dva horní půlřádky klávesnice. Připojením IORQ na CE a RD na OE je zabezpečeno, že paměť EPROM je aktivizovaná jen na IN operaci. Diody D1 až D8 zabezpečují, aby interface neblokoval klávesnici - jsou to 1N4148 (např. KA 207 apod.).

Při stavbě je vhodné dát EPROMku do objímky, protože se pájením může ničit. Připojení joysticků je řešeno pomocí zásuvek CANON, které se u nás těžko shánějí. Ale můžeme je nahradit měděnými dráty o průměru 1 mm, dlouhými asi 10-15 mm. Pro majitele "gumového" Spectra je užitečné namontovat na plošný spoj mikrospínač, který poslouží jako tlačítko RESET.

Pro správnou funkci interfacu je třeba paměť EPROM naprogramovat podle výpisu na obr. 2. Program zabírá jen 1 KB, proto je vstup A10 spojen s GND. Bajty, které ve výpisu nejsou uvedené, mají být FFH (255).

Plošný spoj je oboustranný (obr. 3), krabici interfacu jsem vyhotovil z kuprextitu, dílce jsou na obr. 4. Spojení dílců krabice je provedeno spájením. K tomu slouží dvoumilimetrový pás okolo plošného spoje.

Před připojením interfacu přeměříme zapojení a zjistíme, zda nenastal zkrat. Když je paměť EPROM správně naprogramovaná, interface funguje hned při prvním připojení k počítači. Když se po zapnutí počítač nechová tak, jako před připojením interfacu, ihned ho vypneme, ještě jednou překontrolujeme zapojení a odstraníme chybu.



Obr. 1 - Schéma zapojení

0020 CB C9 CA C8 C3 C1 C2 C0
 0028 8B 89 8A 88 83 81 82 80
 0030 4B 49 4A 48 43 41 42 40
 0038 0B 09 0A 08 03 01 02 00

0120 CB C9 CA C8 C3 C1 C2 C0
 0128 8B 89 8A 88 83 81 82 80
 0130 4B 49 4A 48 43 41 42 40
 0138 0B 09 0A 08 03 01 02 00

0140 7F 7F FF FF 7F 7F FF FF
 0148 7F 7F FF FF 7F 7F FF FF
 0150 7F 7F FF FF 7F 7F FF FF
 0158 7F 7F FF FF 7F 7F FF FF

01C0 34 35 36 37 3C 3D 3E 3F
 01C8 74 75 76 77 7C 7D 7E 7F
 01D0 B4 B5 B6 B7 BC BD BE BF
 01D8 F4 F5 F6 F7 FC FD FE FF

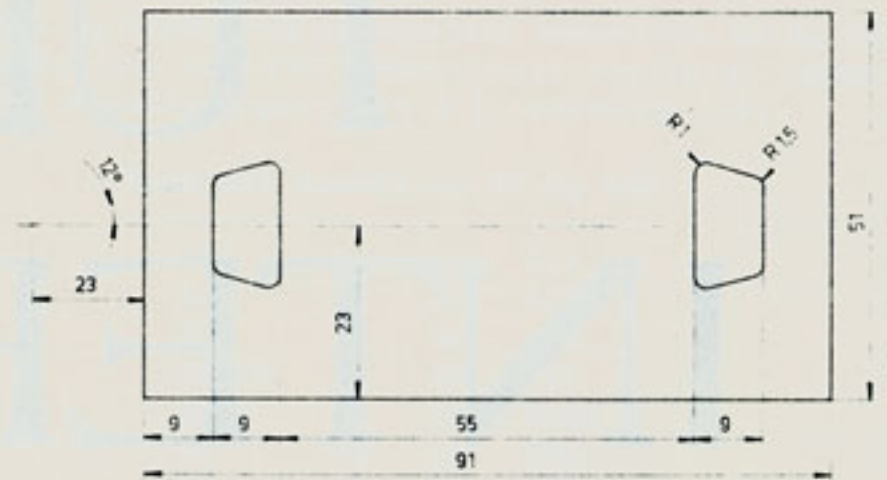
0220 CB C9 CA C8 C3 C1 C2 C0
 0228 8B 89 8A 88 83 81 82 80
 0230 4B 49 4A 48 43 41 42 40
 0238 0B 09 0A 08 03 01 02 00

0240 35 3D 35 3D B5 BD B5 BD
 0248 75 7D 75 7D F5 FD F5 FD
 0250 37 3F 37 3F B7 BF B7 BF
 0258 77 7F 77 7F F7 FF F7 FF

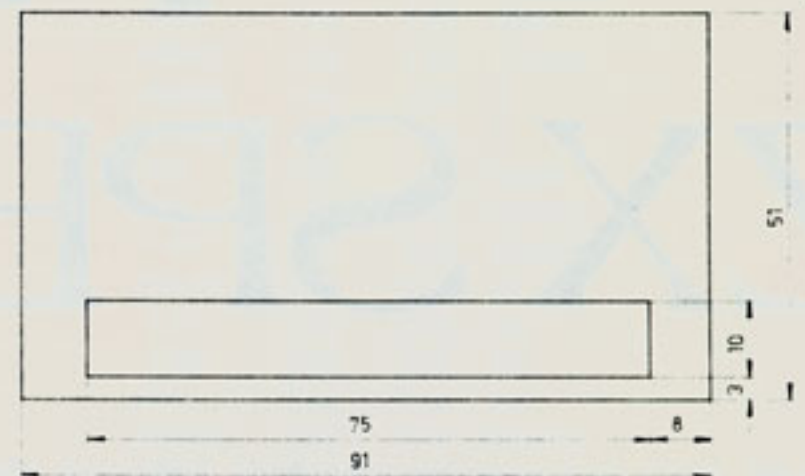
02C0 34 74 B4 F4 3C 7C BC FC
 02C8 35 75 B5 F5 3D 7D BD FD
 02D0 36 76 B6 F6 3E 7E BE FE
 02D8 37 77 B7 F7 3F 7F BF FF

0320 CB C9 CA C8 C3 C1 C2 C0
 0328 8B 89 8A 88 83 81 82 80
 0330 4B 49 4A 48 43 41 42 40
 0338 0B 09 0A 08 03 01 02 00

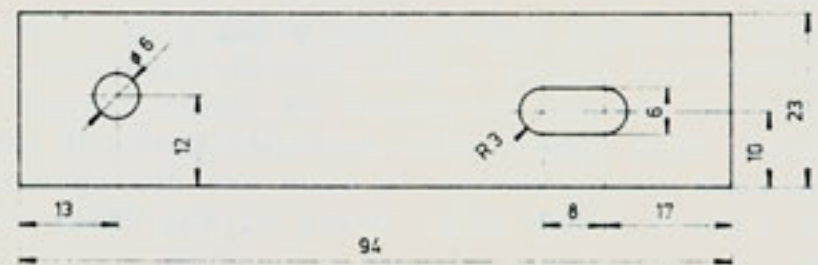
Obr. 2 Obsah paměti EPROM



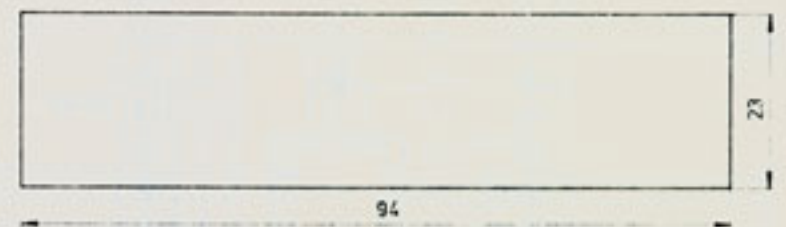
Přední strana



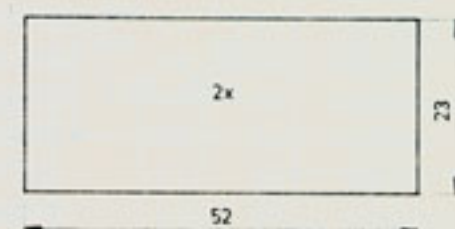
Zadní strana



Vrch

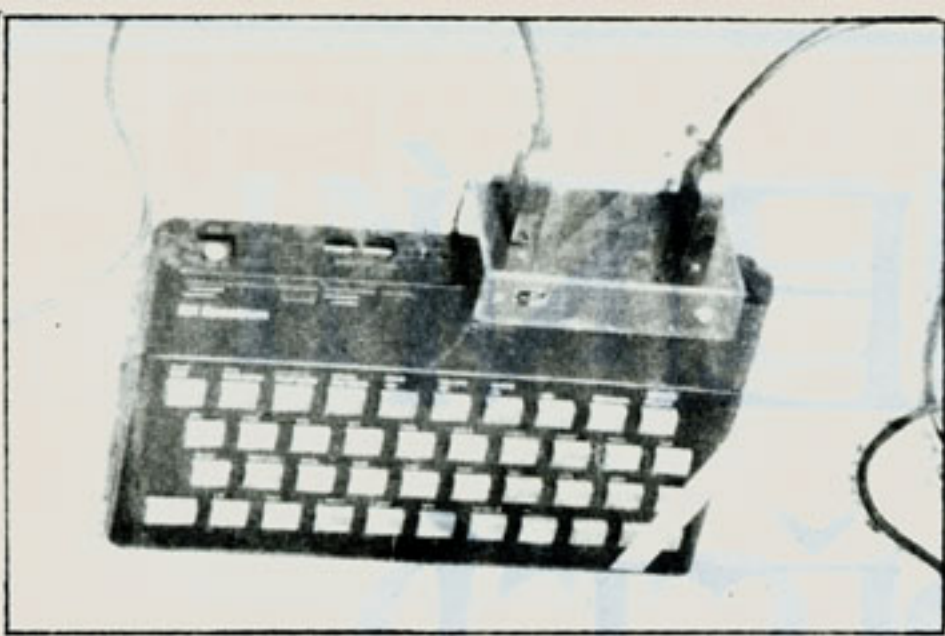


Spodek



Boky

Obr. 4 - Díly krabičky



Obr. 6 - Turbo interface

Těm, kteří se rozhodnou ke stavbě TURBO inter-
facu, přeji dobrou zábavu.

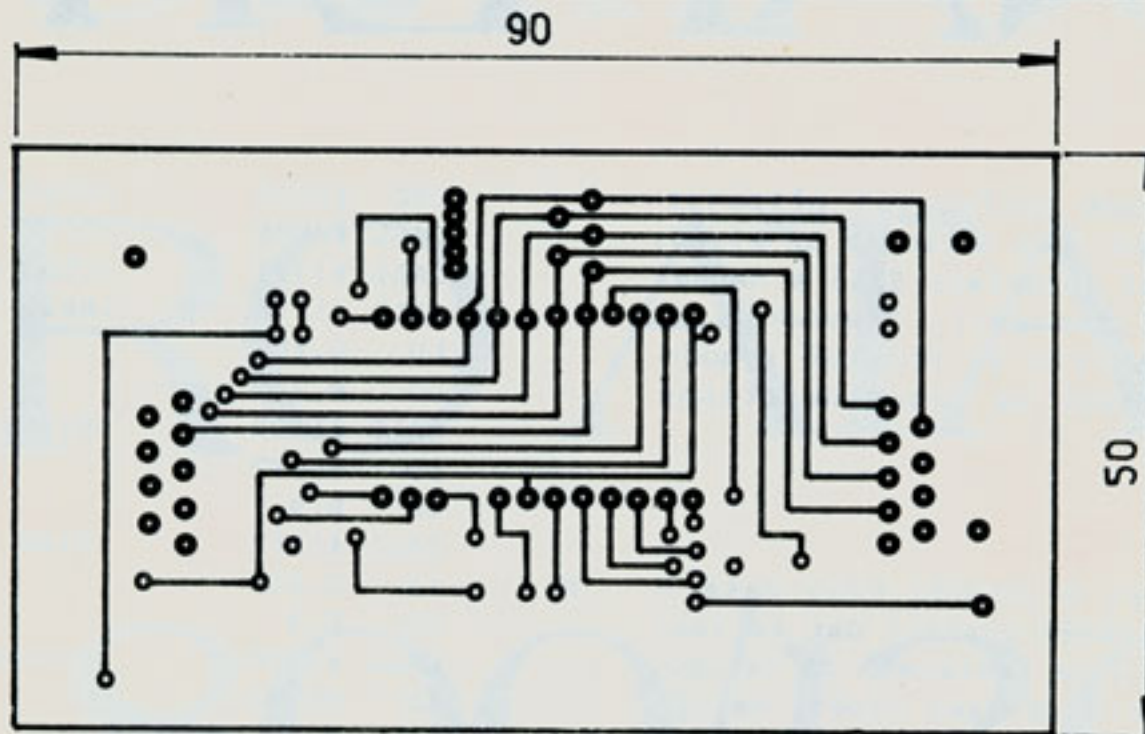
Ing.Ladislav Zakhariás

Literatura

- /1/ Mikroszámítógép Magazin 9/86
- /2/ AR 9/87

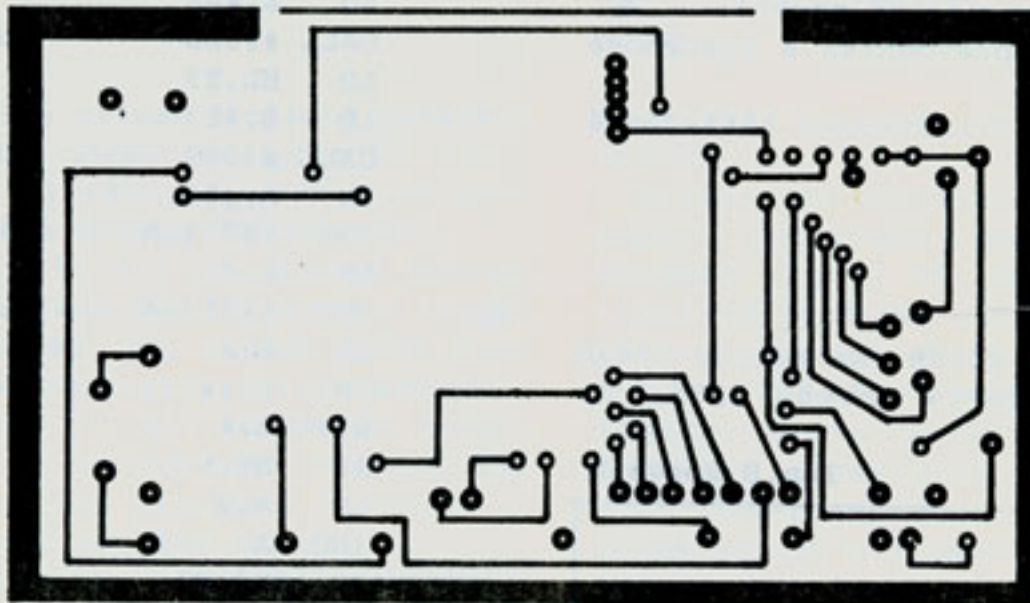
Seznam součástek

I01	MHB 27 16
D1 až D10	KA 207
R1 až R5	1k, TR 151
C1	10M/6V, TE 156
X1	WK 465 80
T1	WN 559 00



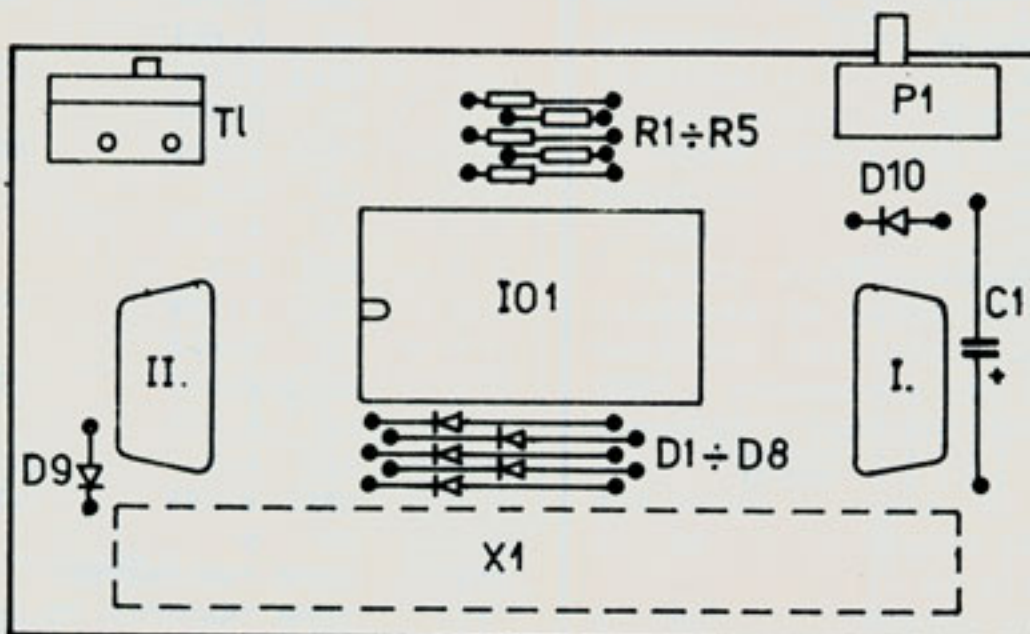
STRANA SPOJOV

Obr. 3a - Strana spojů



STRANA SÚČIASTOK

Obr. 3b - Strana součástek



Obr. 5 - Rozmístění součástek

MĚŘENÍ NAPĚTÍ

K počítači Sord M5 jsem potřeboval připojit převodník A/D. Použil jsem schéma z AR 11/83, připojil k němu paralelní interface 8255 a napsal software umožňující měřit napětí. Převodník je třeba nastavit na rozsah +10 V. Počítač vyhodnocuje napětí s přesností na jedno desetinné místo.

Činnost programu

Počítač posílá na port B obvodu 8255 data v rozsahu 00 až FFH. Po každém vyslání dat se testuje bit 0 portu C, který je nastavený do vstupu. Když je měřené napětí stejné jako napětí na výstupu převodníku, překlopí se komparátor A110. Na jeho výstupu je log.1. PC0 snímá stav na výstupu komparátoru. Když je tam log.1, řízení odskočí do subrutiny, která vyhodnocuje měřené napětí.

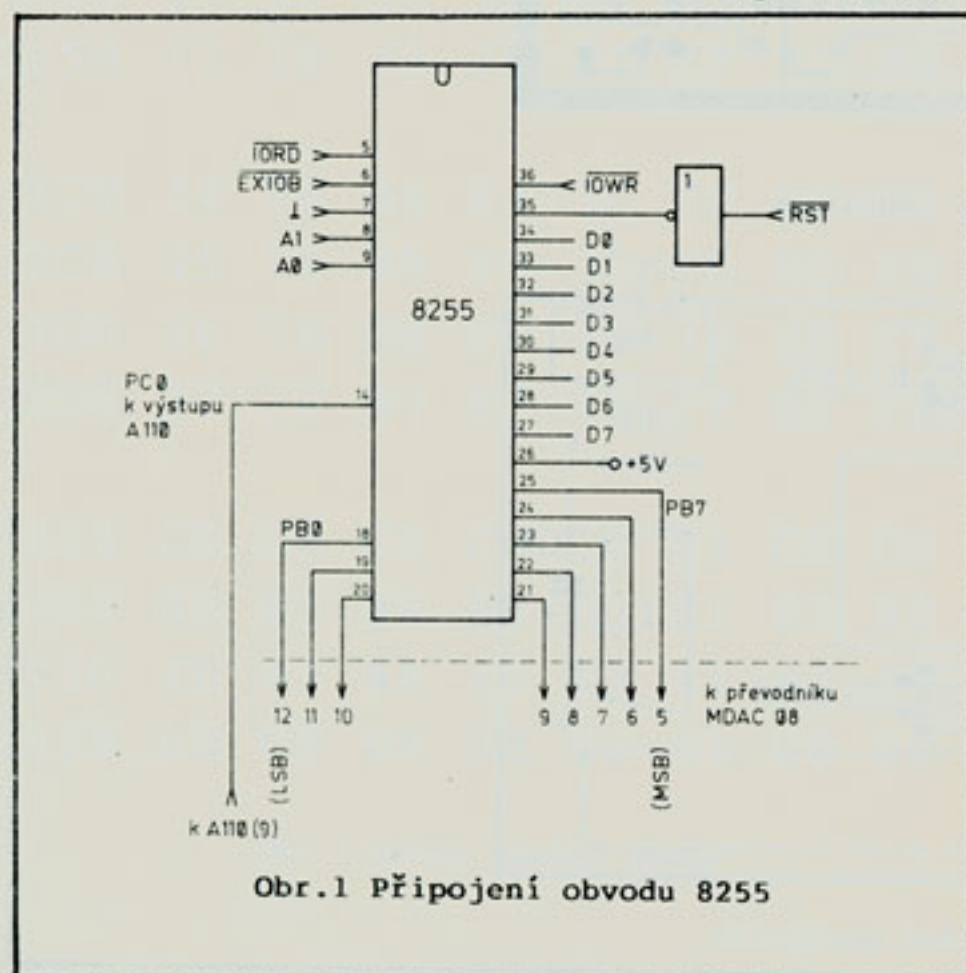
Připojení obvodu 8255 k převodníku a systémové sběrnici je na obr.1.

Obvod 8255 má porty A,B,C nastavené následovně (pro všechny je CW=81H):

- PC0 - PC3 vstup
- PC4 - PC7 výstup
- PB výstup
- PA výstup

Program začíná na adrese C000H. Na adrese C037H je nastavené řídicí slovo CW obvodu 8255. To můžete změnit dle potřeby.

Ing.D.Horváth



```

ORG #0C00          ;Startovací adresa
CALL #0DD8         ;G1
CALL #1393         ;CLS
CALL #1176         ;Bell
LD D,#0B
LD E,2
CALL #13DD         ;Kurzor
LD HL,T2
LD B,9
CALL #1063         ;Tisk textu T2
LD D,#0B
LD E,3
CALL #13DD         ;Kurzor
LD HL,T4
LD B,9
CALL #1063         ;Tisk textu T4
LD D,2
LD E,#0A
CALL #13DD         ;Kurzor
LD HL,T3
LD B,#13
CALL #1063         ;Tisk textu T3
LD A,#81
OUT (#73),A        ;CW
LD A,0
OUT (#71),A        ;Data na PB
LD B,A
IN A,(#72)         ;Data na PC0
BIT 0,A
JP NZ,N1
LD A,B
INC A
JP NC,N2
RET
N1 LD A,B          ;Výpočet napětí
LD L,#27
CALL #1441         ;Násobení
LD A,#64
CALL #0151         ;Dělení stem
LD A,#0A
CALL #0151         ;Dělení deseti
LD D,A
LD A,#30
ADD A,L
CALL #1088         ;Tisk celé části
LD A,#2E
CALL #1088         ;Tisk řádové čárky
LD A,#30
ADD A,D
CALL #1088         ;Tisk desetinné části
JP N
T2  DEFM "VOLTMETER"
T3  DEFM "NAPETIE VO VOLTOCH"
T4  DEFM "-----"

```


INTERFACE

C2111~03

PRO ATARI

800/130

Následující úpravou lze získat 8-bitový paralelní výstup přes konektor joysticku (port č. 1,2). Ale pro připojení na tiskárny bychom jich potřebovali 9 (8 datových a devátý řídicí - SC).

Nechceme-li tedy použít bit pro řízení motoru MGF, tak místo něj použijeme bit D7 dat a v konektoru tiskárny jej uzemníme. Tak nebude možný tisk kódů větších než 127 (obsazeno pro tisk azbuky). Bajt s hodnotou vyšší než 127 ale můžeme do interfacu posílat, protože se stejně bude tisknout kód o 128 menší. Tak lze bez problémů tisknout i inverzní znaky (vytisknou se jako normální).

Tiskárna dává dva signály o své připravenosti (A0, AC). Ty jsou napojeny na adresy AC - 53296,

A0 - 53297 (jinak používané jako tlačítka joysticku).

Ukázalo se zbytečným testovat signál A0 ("došel papír", "otevřená tiskárna" apod.), protože to vše obsáhne signál AC. Dojde-li k něčemu podobnému, tiskárna přestane tisknout; po odstranění příčiny pak v tisku pokračuje.

Tisk vyžaduje vstupní úroveň signálů v TTL logice, proto je na straně ATARI nutné použít budič. V konstrukci jsem využil šestici pozitivních oddělovacích a budících stupňů s otevřeným kolektorovým výstupem z PLR (UCY 7407 nebo 7417) v jednom IO. Zapojení jsem doplnil indikací připojení jednotlivých výstupů portů diodami LED. IO jsou napájeny z vnitřního zdroje počítače.

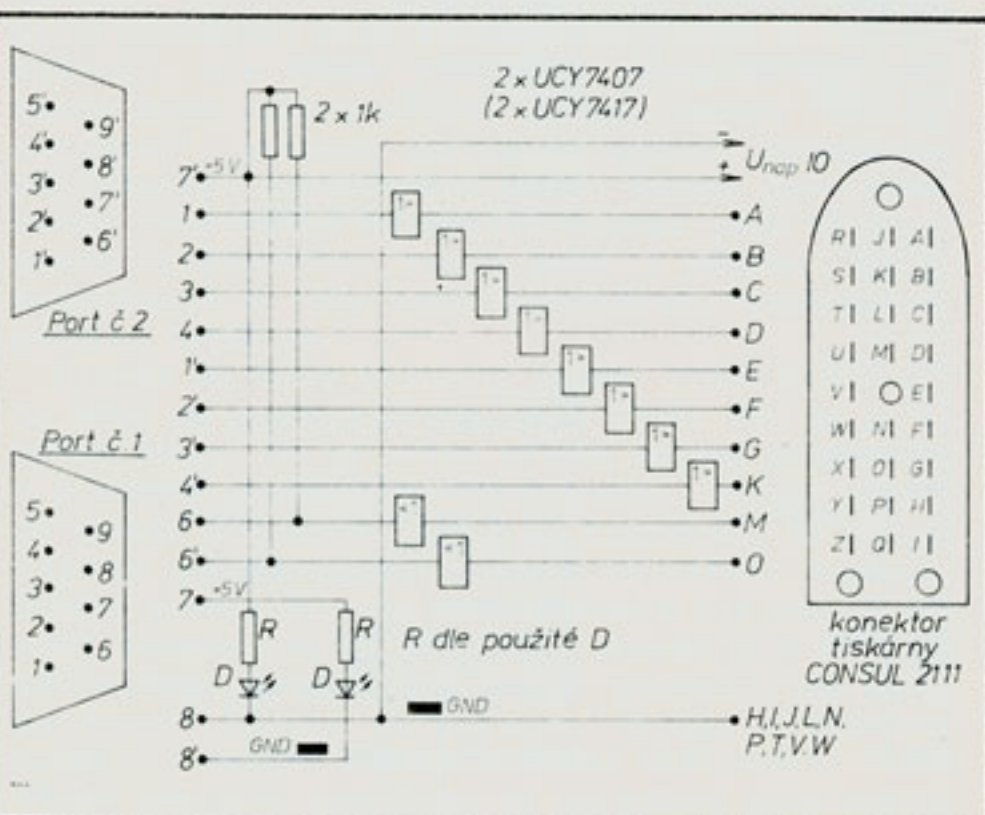
V zásadě lze tento Interface ovládat přímo ATARI Basicem. Pro ilustraci uvádím příkaz k tisku tisk písmene A: (je uveden bez testu připravenosti tiskárny!):

```
10 POKE 54018,56: POKE 54016,255:
   POKE 54018,60 20 POKE 54016,65+128: POKE
54016,0:
  RUN
```

Tento styl ovládání je však příliš pomalý, tiskárna je nadměrně mechanicky namáhána. Tyto nevýhody odstraňuje ovládací program "PRINT-DRIVER" v assembleru CPU 6502.

Komentář k zapojení jednotlivých vývodů:

A-H	D0-D6	Data /SI 1-SI 2/
J	S0	Signál připravenosti RJ vysílat data (L RJ-připravena)
K	SC	Řídicí signál přenosu dat z RJ (H potvrzuje platnost dat)
M	AC	Řídicí signál od tiskárny (H tiskárna požaduje data)



O A0 Připravenost tiskárny
(L tiskárna připravena)
T SI 9 Potvrzuje paritu
(L data bez parity)
W SI 12 Volba režimu práce
(L Start/Stop režim)
I,L,N,P,V,J,T,W,H
S0,SI9,SI12,SI8 GND-uzemněno

Toto uzemnění znamená: RJ je připravena vysílat data, data jsou bez parity, je nastaven Start/Stop režim a nejvyšší bit dat je v úrovni L - tedy nelze tisknout azbuku. Vysvětlivky: RJ - řídicí jednotka (počítač ATARI...)

L - log. nula
H - log. jednička

Popis připojení do joystickového konektoru:

Vývod:	bit:
1	D0
2	D1
3	D2
4	D3
6	sign. AC
8,8'	GND-zem
1'	D4
2'	D5
3'	D6
4'	sign. SC
6'	sign. A0
7,7'	+5 V

Popis subrutiny PRINT-DRIVER

Strojový program tohoto ovládacího programu zabírá 124 bajtů a má 3 způsoby ovládní (volání).

Ve stránce 0 používá adresy 204-207. Podprogram je psán pro 6. stránku paměti (adr.1536), ale může být umístěn podle potřeb uživatele, např. v řetězové proměnné.

Způsoby jeho ovládní (volání):

U = USR (1536,KOD):

provedení jednoho řídicího kódu U = USR (1536,ADR(A\$),LEN (A\$)):

tisk A\$ - řetězec musí být před tiskem řádně deklarován a obsazen U = USR (1536,ADR(A\$),LEN (A\$),NO):

tisk A\$ bez koncových mezer

Seznam řídicích kódů:

8 - posun tiskové hlavy o jeden znak zpět (pokud není na samém začátku stránky!)

9 - tabulační kód posune tiskovou hlavu o X znaků vpřed, X je v rozmezí 1-125; pokud je X větší než 125, najde si tisková hlava pozici X-125 na dalším řádku. Tento kód inicializujeme následovně:

- nejprve vyvoláme příkazem U nastavení módu "tabulátor"

- při opětovém volání U místo kódu 9 použijeme číslíci pro nastavení tiskové hlavy do patřičné tiskové pozice

10 - návrat tiskové hlavy na začátek a posun papíru o jeden řádek vpřed (pokud ovšem není hlava na začátku řádku - pak se provede jen posun o řádek vpřed)

11 - posun papíru vpřed dle programu (funguje pouze se stránkovacím zařízením!)

12 - posun papíru na začátek nového formuláře (obdoba kódu 11)

13 - návrat tiskové hlavy zpět na začátek řádky rychloposuvem

17 - nastavuje červenou barvicí pásku na tiskové hlavě

18 - nastavuje černou barvicí pásku na tiskové hlavě

27 - posun papíru o jeden řádek zpět

32 - posune tiskové hlavy o jeden znak vpravo

Zbývající kódy jsou obsazeny (v rozsahu 33-127) grafickými znaky, číslicemi a malými a velkými písmeny anglické abecedy.

Jediná změna znaků na této tiskárně je u znaků:

\$ se tiskne jako

0 se tiskne jako

Bližší informace jsou uvedeny v "Návodu pro obsluhu tiskárny CONSUL 2111-03".

Příklad tisku písmene "A":

U = USR (1536, 65)

František Vondrouš

	Adresa (dek.)	
POCPAR	204	Počet parametrů volání
ADRESA	205	Adresa uloženého textu v tisku
ADRESL	205	Nižší bajt adresy
KOD	205	Kód řízení (pokud byl zadán)
ADRESH	206	Vyšší bajt adresy
DELKA	207	Počet znaků k vytisknutí (délka textu)
READY	53296	Hlášení od tiskárny (AC) - stlačení TRIG na 1 (0) joysticku
READY 0	53297	Hlášení od tiskárny (A0) - nepoužito TRIG na 2 (1) joysticku
PORTA	54016	ADR portu A (joysticku)
PACTL	54018	Řízení portu A

Subrutina PRINT-DRIVER:

LDA #56	Port A pro výstup
STA PACTL	
LDA #255	
STA PORTA	
LDA #60	
STA PACTL	
PLA	Výběr a uložení počtu parametrů, adres či kódu
STA POCPAR	
PLA	
STA ADRESH	
PLA	
STA ADRESL	
LDA POCPAR	
CMP #1	Test jednoho parametru
BEQ A5	Tisk kódu (funkce)
PLA	
PLA	
STA DELKA	
LDA POCPAR	
CMP #2	Test druhého parametru
BEQ A1	Tisk řádky
PLA	Výběr třetího parametru
PLA	
LDY DELKA	Zkrácení délky o koncové mezery
BEQ A3	Nulová délka - konec
A0 DEY	
LDA (ADRESA),Y	
CMP #32	
BNE A1	Znak na konci není mezera, konec zkrac.
STY DELKA	Byla mezera, nová délka (-1)
BEQ A0	Skok vždy, dále zkracuj
A1 LDY #0	Tisk řádky
CPY DELKA	Test nulové délky
BEQ A3	Nulová délka - konec
A2 LDA READY	Test připravenosti tiskárny
BEQ A4	Nepřipravena!
LDA #128	Nastavení řídicího bitu D7
ORA (ADRESA),Y	Výběr znaku z paměti
STA PORTA	Vlastní tisk
INY	
CPY DELKA	
BNE A2	Ještě je co tisknout
A3 LDA #56	Konec, port A pro výstup
STA PACTL	
LDA #0	
STA PORTA	
LDA #60	
STA PACTL	
RTS	Návrat
A4 LDA #0	Vynulování řídicího bitu D7
STA PORTA	
BEQ A2	Skok vždy, nový test
A5 LDA READY	Test připravenosti tiskárny
BEQ A6	Nepřipravena!
LDA #128	Nastavení řídicího bitu D7
ORA KOD	Kód funkce
STA PORTA	Vlastní provedení
BMI A3	Skok vždy, konec
A6 LDA #0	Vynulování řídicího bitu D7
STA PORTA	
BEQ A5	Skok vždy - nový test

METAKOMUNIKACE

PROGRAMUJ, ALE ZŮSTAŇ ČLOVĚKEM

S chutí jsem si přečetl článek Eda Posta o pojídačích koláčů a opravdových programátorech (Mikrobáze 5/88). Teď už vím zcela jistě, že se nechci stát opravdovým programátorem. Ale ani pojídačem koláčů. Vysvětlím.

Postův opravdový programátor je bytost s nadáním, které nemá každý. Je zřejmé, že vedle vši nezbytné píce, vůle a chtění (vnitřní motor o vysokých otáčkách) je nutný právě onen zcela zvláštní typ nadání. Není každému shůry dáno, aby opravil neznámou chybu v operačním systému telefonické navigací běžného uživatele. Člověku s jednostranně vyvinutou schopností psychologové říkají nepřiliš taktně idiot savant. Takový jedinec se bez větší námahy naučí nazpaměť telefonní seznam nebo zpaměti provádí početní operace vyšší matematiky apod. Nerad bych, aby mě někdo podezíral z toho, že opravdoví programátoři jsou pro mě idioti. Ani omylem. Naopak si myslím, že psychologové se trochu mýlí. Vykazuje-li mozek takové schopnosti, pak je zřejmé schopen je vykazovat proto, že je takový, jaký je. A mohl by je vykazovat i u jiných lidí. Což jsme vlastně všichni. Takže tu jde spíš o otázku - jakto, že někomu to takhle funguje, ostatním nikoli, a když už, proč někdy tak jednostranně? (Viz i tzv. extrasensové.) To je otázka pro širokou interdisciplinární disputaci, která snad přinese plody za nějakých pár století. Je pravdou, že "savanti" bývají po jiných stránkách své psyché poněkud mimo. Ale rovněž je pravda, že vynikající jednotlivci, o kterých se nedomníváme, že by byli nějak zvlášť mimo, mají "savantí" schopnosti, a přesto je za idioty nepovažujeme (alespoň podle našich společenských konvencí; ufoní by se na to a na nás vůbec možná koukali úplně jinak).

Svůj problém vidím jinde. A myslím, že v tom mám mnoho společného s mnohými. Především nemám ono nadání řešit problémy systémů podle telefonických zповědí. Dál vím, že když se začnu příliš zabývat programováním, sice mi sílí analytické a logické myšlení, ale jinak (vnitřně) význačně pustnu. Skoro bych takový proces nazval osobní "dekulturizací", ztrátou společného gruntu, vytvářením odloučeného samonosného světa. Řečeno jednoduše - blbnu (via idiot savant?). Hledal jsem nějakou cestu vyváženosti. Zatím jsem ji nenašel. Buďto programuju a jedu jako kolovrátek, nebo žádný program neudělám.

Člověk se může ohloupit různě. Třeba klasicky - to když se o něm v širém okolí říká: Ten je do ní úplně blbej. To je ovšem případ celkem příjemného - a hlavně přechodného - zhroupení. Ztráta gruntu je rovněž přechodná, ale o to fikanější, že potenciálně předpokládá budoucí vytváření nového. Ztráta gruntu šíleným programováním nic takového v sobě nenese. Je to něco jako propadliště drogové závislosti. Dokonce ani vnější pustnutí (Ed Post: banánové šlupky na podlaze, vajgly v hrnci s kafem...) nenese známky bohémské ležérnosti. Vedle všeho okolního nepořádku má opravdový programátor

ve svých datech precizní přehled.

Jenže... Nejen opravdový programátor si může začít myslet, že stejně precizně má fungovat svět kolem něj, protože všechno jsou vlastně jen programová data. Termínem psychologa B. Blažka lze takovou deviaci nazvat datismus (zde ve smyslu generalizace). Neříkám to z titulu nějaké povýšenosti, ale proto, že podobnou potrhlost jsem si už na sobě "odpracoval". O to lépe ji dnes dokážu rozeznat a pochopit u druhých. A o to víc se skláním před slovnou ženskou "přízemností" a "nelogičností", které nás tak provokují, aniž si uvědomujeme, že nás zároveň odvádějí od úplné anihilace v našich lopotně konstruovaných iracionálních světech. Žena v roli životního debuggeru...? Bezbřehé téma nejen pro romanopisce.

Znám programátora, který začal ZX Spectrem. Tenkrát jsem se s ním domluvil. V dalších letech se zcela ponořil do programování. Strašně moc toho ví a zná. Ale když se ho chci na něco zeptat, skončí to jeho nerudnou reakcí. Prostě nefunguju jako jeho programy. Nedokáže se mnou mluvit na bázi gruntu. Jeho existenci už si ani nepřipouští. Dokáže se bavit jen s člověkem, který má logickou zkušenost blízkou té jeho - proti tomu samotnému nic. Ale jinak...?

Nedávno jsem byl na přednášce RNDr. Z. Neubaera. Zabýval se narůstající neschopností domluvy mezi vědci. Obory vědy se hustě větví, odborník často vnímá už jen úzký horizont svého oboru, jímž se zabývá skoro jen on sám. Specializace dosahuje takové míry, že se vědci z příbuzných oborů pomalu nedokáží domluvit mezi sebou. Každý z nich má jiný "specializovaný" prožitek, jiný slovník (i stejně znějící slova nabývají jiného významu). A ouha, když se má přejít na grunt. Jeden komunikační kolaps stíhá druhý. Na šedé šachovnici do sebe narážejí strakatě vybarvené stereotypy a paradigmata.

Dnes dominující lineární přístup se dostává do konfliktu s přirozenou nelinearitou, ireversibilitou života. Jsou to dvě rozdílné cesty povstávání řádu z chaosu (viz I. Prigogine, F. Capra a řada dalších).

Když se objevily první domácí počítače, napadlo mne - ha! konečně něco, co pohne myšlením a vědomím lidí! Ta struktura, ty funkce, ty možnosti využití! Jenže po těch letech...kde nic moc, tu nic moc. Naopak začínám mít dojem, že počítače lidem jen nahustily duše daty a ještě víc je vzdálily sobě i realitě (za to ovšem počítače nemohou).

Nejednou se na stránkách Mikrobáze objevily lamentace nad izolovaností počítačů, které se nestávají součástí struktur informačních metasystémů, ale dřímou na našich stolech jako ta opěvaná panenka v koutě. Tak opravdu skoro k ničemu nejsou. Počítač je svou povahou především komunikačním nástrojem pro přenos, tok, potok, řeku informací. Jen zvednutí zbytečných hrází může okysličit zatuchle stojaté vody, splavit z nás

anachronickou počítačovou symboliku samohonkového bastlení. Jen tak se počítač stane skutečným nástrojem obecného užiku. A já se nebudu muset dovolávat nerudného datisty, ale spojím se se specializovanou databankou, která na mě všechno vysype. Místo věčně neúspěšného telefonování spoustě lidí (ztráta nervů a času) jim v klidu naťuknu své vzkazy v editoru počítače a modem už ostatní zařídí sám. A přihlásím se do elektronického kursu čehokoli, co mě bude zajímat. Když budu mít čas, připojím se na školní či fakultní databanku chytře zpracovaného, stále aktuálního vzdělávacího systému a budu se učit, co, kdy a jak budu chtít. Atd. apod.

Nějak nám nedochází, že člověk je s počítačem propojen neviditelným uživatelským interfacem. Počítač a já tvoříme jednu komunikační mikrostrukturu. Když je počítač izolován, spolu s ním jsem od vědomostí a akční potence ostatních lidí separován i já, oni ode mne, jeden od druhého, všichni ode všech a ode všeho. U některých aplikací to vadí jen částečně. Pro hromadu jiných (a pro vývoj současné společnosti vůbec) je to zoufalá překážka. A tak sedáme v koutku a jen tak si bastlíme a čas letí. A degradace pojmu "uživatel počítače" praská ve švech. Savantí jízda v zamknuté garáži...

Co říci k pojídačům koláčů, jak Post nazývá pascalské programátory? Především to, že jakákoli segregace je nesmyslná. Nepodezřívám z ní Eda Posta, ale známe své Pappenheimské. Nechť si každý programuje, v čem chce. Moje osobní sympatie patří assembleru. Z prvních doteků s počítačem si ještě pamatuju Basic. Občas se mi hodí. Před nějakým rokem jsem překládal kurs Pascalu z polského Bajtku. Říkal jsem si - konečně se přinutím ten opěvovaný jazyk vstřebat. Omyl. Nepadli jsem si do oka. Pascal se mi prostě nelíbí. I když vím, v čem se liší od Basicu, v jistém odstupu nemohu ze sebe setřást dojem, že jde jen o jeho lepší variantu (prosím pojídače koláčů, aby je po mně přestali házet). Zůstal jsem u assembleru. Víím, že jím nezvládnou matematické výpočty jako s Pascalem. Ale jednak takové problémy nevyhledávám a v nejhorším mám po ruce kalkulátor Basicu. U assembleru se mi líbí, že jde na dřevě věci a nijak neomezuje v rozletu, i když program kyne pomalu (málo droždí v těstu na koláče). Ale výsledek všechno vynahradí.

Právě ta skoro neohraničená svoboda volby programové struktury mě na assembleru vábí nejvíc. Jinde ji postrádám (každou svobodu si člověk holt musí něčím vykoupit). Běžně narážím na potřebu vytvořit pomocnou rutinu pro nějakou konverzi, komprimaci, přenos apod. Trochu ťukání do editoru assembleru, a hotovo. Pár desítek či stovek bajtů. Pohoda. A větší program? To je právě to - kde vzít ten čas, jehož nekonečným žroutem assembler je (tady mají pojídači koláčů výhodu). Jeden takový program mám rozdělaný, pořád ho nosím v hlavě, zdroják zpoloviny hotov, jen kdyby byl k mání ten čas assemblerového vykoupení... Ale abych nevypadal jako oběť fixních iluzí o assemblerové nepřekonatelnosti - zalíbilo se mi Céčko. Programové části nevyžadující přímo strojákovou rychlost budu za nějaký čas psát v něm (snad mi assemblerový bůžek odpustí). Pokud jde o eleganci slečny Ady či mohoucnost páně Prologa, chovám k nim uctivý vztah, který nejspíš nepřestoupí práh své platoniky. Mám za to, že umět opravdu myslet i (intuitivně) cítit jedním, dvěma programovacími jazyky, je hodně. Pro udržení této vysoké úrovně třeba v pěti jazycích nemám žádný praktický důvod. To už bych radši dal přednost švédštině, dánštině, finštině... (nejen pro blondaté seveřanky...).

Nemohu se zbavit dojmu, že současný způsob tvorby programů spolu s dost stereotypní komunikací člověk-počítač na nás seslal nějaký pekelník. Rád bych se narodil ještě jednou, až se počítače nebudou programovat, ale učit (a my s nimi), až

budou mít něco jako intuici, a přesto neztratí nic ze své střežlivé ekvilibristiky s logickými úrovněmi.

Před pár dny jsem v nějakém sborníku četl pojednání o expertním systému pro výuku psychoterapie. Autor - programátor - tam rozvinul svou představu o tom, jak jeho program (v roli klienta/pacienta) povede se studujícím psychoterapie dialog, díky němuž se student rychleji naučí léčit skutečné lidi. Typický produkt personifikace počítače a datismu. Na schopnosti konfigurace CPU-paměť roubuje nekonečné variace vnitřního modelu lidských světů a příliš ho nezajímá, jak komplexní proces prezentace takového modelu v člověku probíhá. Sumárně jde o velice subtilní "mind-body problem", o to komplexnější v dynamické komunikaci dvou lidí. Pro korunovací svého díla onen expert do programu zasadil i slovní reakce vyléčených pacientů! Chybí už jen soška andělíčka, který se zakývá, když do něj hodíte korunu.

Měli jste možnost výuky řízení automobilu na simulátoru jízdy? A pamatujete se na rozdíl mezi touto výukou a svým prvním vyjetím do ulic? Všechno je najednou jinak, i simulátorem naučené reflexy mohou zradit, dokud nezískáte nadhled, postupně budovaný zkušenostmi přímo z terénu. A to má auto čtyři kola! Před první jízdou na motocyklu nebo šlapacím kole bude každému jakýkoli simulátor houby platný.

Oproti tomu - když má někdo dlouhodobou, zřídka přerušovanou zkušenost jen s logikou počítače, snadno ji do sebe vstřebá natolik, že ať se pak pustí do jakéhokoli společenského terénu, bude jevit sklon k pojímání všeho "duší" svého klávesnicového přítele, mikroprocesorového důvěrníka. Když někdo podlehne pití, je to na něm vidět. Když se někomu změní psychika vlivem déletrvajících, jednostranně excitujících náporů, nic tomu navenek nemusí nasvědčovat. Postižený je přesvědčen, že je v naprostém pořádku - na rozdíl od alkoholika nemá žádný zjevný tělesný feedback. Všechny (i srdce-rvoucí) indikace okolí považuje za chybu v cizím systému.

Psychoterapeuti počítačově rozvinutých zemí za nejkritičtější programátorský věk považují léta od čtyřicítky nahoru. Křesla ordinací se plní dosud velmi úspěšnými jedinci, kteří si stěžují v podstatě na jedno - mají pocit, že jim ujel nějaký vlak a chtěli by vědět, kde ho dohnat a jak nasednout. Jeden génius Křemíkového údolí se rozhodl pro radikální řešení. Jednoho dne všechen křemík hodil za hlavu, koupil si starý hubertus a připojil se k americkým tulákům. Poněkud černý humor do věci vnáší cílevědomé rojení psychoterapeutických ordinací kolem počítačových center ("jejich zaměstnanci velmi slušně vydělávají a dříve či později se stanou našimi klienty..."). Trochu tristní podoba zákona nabídky a poptávky (Koně se také střílejí).

I když je Křemíkové údolí od nás daleko, není to žádné štěstí. Procesu hloupnutí bohatě vyhoví i obyčejné ZX Spectrum. Jde o to, s čím a jak k programování sami v sobě přistupujeme - zda chceme, aby naše činnost všestranně rozvíjela naše schopnosti komplementárního, celostního pohledu na svět, obohacovala náš lidský, obecný grunt. Pro to si ale musíme uvědomit dost podstatných věcí předem a hlavně na ně průběžně nezapomínat.

Někde jsem četl větu: Programování učí logice, ne etice. Tato sentence, která se nemusí vztahovat jen k programování, je odosobněným "přiznáním hříchu", neboť vpravdě říká: Ten, kdo neučí etice, ale jen logice, je učitel programování (platí i pro samouky). Vždy záleží na tom, jak s čím člověk zachází (nechce-li tím zároveň scházet). Odvěký problém hledání zlaté střední cesty... Z pohledu učenců tao - jak být jin a neztratit jang?

Ale teď už mě omluvte, na dveře mi klepe Céčko. -elzet-

PŘEHLED LASEROVÝCH A SPECIÁLNÍCH TISKÁREN

V minulém čísle Mikrobáze jsme uvedli přehled jehličkových tiskáren. Celkový přehled nyní zakončíme tiskárnami laserovými a speciálními. Tiskárny typu daisy wheel neuvádíme, protože u nás mají nejmenší význam. Všechny ceny jsou opět v anglických librách (bez daně) z období letních prázdnin roku 1988.

Použité zkratky v přehledu laserových tiskáren:

Lstg - cena v anglických librách
s/m - rychlost tisku (počet stránek za minutu)
M - typ mechaniky
Ca-Canon Ci-Citizen G-Genicom H-Hitachi
I-IBM Ko-Konica Ky-Kyocera M-Mita N-NEC
R-Ricoh To-Toshiba Te-Tec X-Xerox
min - minimální velikost stránky
max - maximální velikost stránky
(míry uvedeny v palcích)
USL - 8,5 x 14 (US Legal)
Let - 8 x 11 (US Letter)
Exe - 7 x 9 (Executive)

PI - počet papírových zásobníků na vstupu
pO - počet papírových zásobníků na výstupu
mI - možnost manuálního posuvu papíru
f - počet zabudovaných znakových sad (fonts)
c - počet konektorů pro přídatné sady znaků (cartridges)
DL - paměť pro načtení znakové sady zvenčí (download)
pams - rozsah standardní (dodávané) paměti
pamm - maximální rozsah rozšířené paměti
E - emulace Epson FX80
D - emulace Diablo
L - emulace Laserjet
p - pracuje s jazykem Postscript
P - paralelní interface Centronics
S - sériový interface RS232
sp - speciální interface
* - uvedená možnost je v ceně tiskárny
+ - možno/nutno přikoupit
č. - odkaz na seznam názvů tiskáren:

1 - Apple LaserWriter IINT	40 - Kyocera F-2200
2 - Apple LaserWriter IINTX	41 - Kyocera F-3000
3 - Apple LaserWriter IISC	42 - Laser Master CX II
4 - Apricot Laser 2	43 - Laser Master RX
5 - Apricot Laser PSB	44 - Mannesmann Tally MT 910
6 - AST Turbolaser	45 - NCR WT910
7 - Blaser Star	46 - NEC Silent Writer LC815
8 - Brother KL-8	47 - NEC Silent Writer LC890
9 - C. Itoh LIPS 10	48 - Oki Laserline 6
10 - Canon LBP-8 II	49 - Olivetti PG208
11 - Citizen Overture 110+	50 - Panasonic KX-4450
12 - CTP LP-15	51 - Pragma Laserpro 805
13 - CTP LP-6	52 - Pragma Laserpro 810
14 - CTP LP-8 GS	53 - Pragma Laserpro 1510
15 - CTP PS-8	54 - Pragma Laserpro Express
16 - Dataproducts LRZ-1230	55 - Pragma Laserpro Silver Ex
17 - Datasouth Pagewriter 8	56 - Pragma Silver Exp.2
18 - DEC LN03	57 - Pragma QMS Lasergrafix 80
19 - DEC LN03 Plus	58 - Pragma QMS PS BOC II
20 - DEC Script Printer	59 - Pragma QMS-PS810
21 - Decision Data 6408	60 - Pragma Smart Writer Plus
22 - Decision Data 6415	61 - Printronix L1012
23 - Epson GQ-3500	62 - Qume Laser Ten Plus
24 - Facit Opus 2 P7000	63 - Qume ScriptTen
25 - Facit Opus 2E P7080	64 - Rank Xerox 4045
26 - Facit Opus 3E P7150	65 - Rank Xerox 4046 50
27 - Facit Opus 4 P6010	66 - Ricoh LP 4081-R1
28 - Genicom 5010	67 - Ricoh PC Laser 6000
29 - Genicom Pageprinter 8	68 - Ricoh Postscript 8
30 - Hermes F-1000	69 - Ricoh Postscript 15
31 - Hermes F-1200	70 - Star Laserprinter 8
32 - Hermes F-2200	71 - InterTAM LP 1000
33 - Hermes F-3000	72 - TI Omnilaser 2015
34 - HP Laserjet Series II	73 - TI Omnilaser 2108
35 - HP Laserjet 500+	74 - TI Omnilaser 2115
36 - Honeywell Laserpage 801	75 - Toshiba Pagelaser 12
37 - IBM 4216	76 - Wang LDP8
38 - Kyocera F-1000	77 - Wang 8/1
39 - Kyocera F-1200	78 - Wang 12/1

č. Lstg s/m M min max pl pO mI f c DL pams pamm E D L p P S sp

20	4910	8	Ri	A4	A4	1	1	8	*	2M				*	*			
21	4150	8	Ri	A4	A4	1	1	8	2					*	*	*	*	
22	5999	15	Ri	A4	USL	2	1	8	2					*	*	*	*	
23	1795	6	Ri	A5	USL	1	2	*7	2	*512K	2M			*	+	*	+	
24	2995	8	Ri	Let	A4	1	1	4	2	*				*	*	*	*	
25	3695	8	Ri	Let	USL	1	1	4	2	*512K	2M			*	*	*	*	
26	4995	15	Ri	Let	USL	2	1	4	2	*512K	2M			*	*	*	*	
27	1995	6	Ri	Let	USL	1-2	1	4	1	*1M	2M			*	*	*	*	
28	2995	10	H	Let	USL	2	1	*6	4	*				*	*	*	*	
29	1745	8	G	A5	USL	1	1	*1	1	+	512K	2M		*	+	+	*	*
30	2320	10	Ky	A5	USL	1	1	*36	2	*512K	1M			*	*	*	*	*
31	3460	10	Ky	A5	USL	1	1	*36	2	*1,5M	2M			*	*	*	*	*
32	4760	10	Ky	A5	USL	2	6	*78	2	*1,5M	3,5M			*	*	*	*	*
33	6660	18	Ky	A5	USL	2	6	*78	2	*1,56M	3,5M			*	*	*	*	*
34	2249	8	Ca	Let	USL	1	1	*6	2	*512K	4,5M			*	*	*	*	*
35	3995	8	Ca	B5	USL	2	1	*6	1	*512K				*	*	*	*	*
36	2995	8	R	Let	A4	1	1	*6	2	*512K	2M			*	*	*	*	*
37	3370	6	1	A5	USL	1	2	*11		*2,5M				*	*	*	*	*
38	2320	10	Ky	A5	USL	1	1	*36	2	*512K	1M			*	*	*	*	*
39	3460	10	Ky	A5	USL	1	1	*36	2	*1,5M	2M			*	*	*	*	*
40	4760	10	Ky	A5	USL	2	6	*78	2	*1,5M	3,5M			*	*	*	*	*
41	6660	18	Ky	A5	USL	2	6	*78	2	*1,5M	3,5M			*	*	*	*	*
42	3295	8	Ca	A4	USL	1	1	*60		*1,5M	2M			*	*	*	*	*
43	4100	8	Ri	A4	USL	1	1	60		*1,5M				*	*	*	*	*
44	2995	10	Ky	A4	A4	2	2	*3	2	*512K	1,5M			*	*	*	*	*
45	3300	10	Ky	A4	A4	2	2	*3	2	*512K	1,5M			*	*	*	*	*
46	2250	8	N	6x11	USL	1-2	1	4	2	*128K	1,3M			*	+	*	*	*
47	3950	8	N	6x11	USL	2	1	*35		*3M	3M			*	*	*	*	*
48	1750	6	Ri	A5	A4	1	1	-2	*15	1	256K	600K		+	+	+		
49	2479	8	Ca	Let	USL	1	1	*6	2	*512K	4,5M			*	*	*	*	*
50	1995	11	P	A4	A4	2	1	4	1	512K	1,5M			*	*	*	*	*
51	3041	8	Te	4x6	USL	1	1	*19		*640K				*	*	*	*	*
52	3884	8	Ri	Let	USL	1	1	*22		*768K	1,2M			*	*	*	*	*
53	5528	15	Ri	4x6	USL	2	1	5		*768K				*	*	*	*	*
54	1810	8	Te	B5	USL	1	1	*19	+	*512K				*	*	*	*	*
55	2181	8	Te	B5	USL	1	1	*25	1	*	768B			*	*	*	*	*
56	1665	8	T	A4	USL	1	1	*19	1	*640K	640K			*	*	*	*	*
57	5888	8	Ri	B5	USL	1	1	*						*	*	*	*	*
58	5018	8	C	A4	USL	2	1	*35		*2M	3M			*	*	*	*	*

č.	Lstg	s/m	M	min	max	pl	pO	ml	f	c	DL	pams	pamm	E	D	L	p	P	S	sp
59	4453	9	Ca	B5	USL	1	2	*	35	*	2M	3M		*	*	*	*	*	*	*
60	2329	8	Ca	B5	USL	1	2	*	17	2	*	1M	2,5M	*	*	*	*	*	*	*
61	2895	12	M	Let	USL	2	1	6	*		512K	512K	*	*	*	*	*	*	+	
62	2683	10	H	Let	USL	2	1	*	3	3	*	512K	512K	*	*	*	*	*	*	*
63	4595	10	H	Let	USL	1-2	1	*	35	*	3M			*	*	*	*	*	*	*
64	3995	10	X	A5	USL	1	1	2	3	*	1M	2M	+	*	+	*	*	*	*	*
65	5595	10	X	A5	USL	3	1	2	4	*	1M	2M		*	*	*	*	*	*	*
66	3195	8	Ri	Let	A4	1	1	4	2		512K		+	*	+	*	*	*	*	*
67	2195	6	Ri	A5	USL	1	1-2	*	4	1	*	1M	2M	+	*	+	*	*	*	*
68	5195	8	Ri	A5	A4	1	1	4	2	*	2M			*	*	*	*	*	*	*
69	7195	15	Ri	A5	USL	2	1	4	2	*	3M			*	*	*	*	*	*	*
70	2195	8	Ca	Exe	USL	1	2	*	8	2	*	1M	2M	*	*	*	*	*	*	*
71	1795	6	Ri	A5	USL	1	1	*	4		1,5M			*	*	*	*	*	*	*
72	4445	15	Ri	Let	USL	2	1	2	2	*	512K			*	+	*	*	*	*	*
73	4250	8	Ri	Let	A4	1	1	4	2	*	2M			*	*	*	*	*	*	*
74	5945	15	Ri	Let	USL	2	1	4	2	*	3M			*	*	*	*	*	*	*
75	2995	12	T	85	USL	1-2	1	*	3	3	*	512K	1,5M	*	*	*	*	*	*	*
76	1995	8	Ca	A4	USL	1	1	*	6	2	*	512K	2M		*	*	*	*	*	*
77	3656	8	Ri	A4	A4	1	1	2		*	1,5M	2M	*	*	*	*	*	*	+	*
78	3095	12	T	A5	USL	2	1	*	6	3	*	512K	2M	*	*	*	*	*	*	*

Další tabulka podává přehled o speciálních tiskárnách - termálních a inkjetových.

Použité zkratky:

Lstg - cena v anglických librách (bez daně)
e - počet tiskových elementů
z/s - rychlost tisku (cps)
šp - max. šířka papíru v palcích
: - tisk na perforovaný papír

z/" - počet znaků na palec (na řádce)
P - paralelní interface Centronix
S - sériový interface RS232
T - typ tiskárny:
1 termální
2 ink jet
3 7-barevný ink jet
4 bubble jet
* - uvedená možnost je v ceně tiskárny
+ - možno/nutno přikoupit
a - možná alternativa (v ceně tiskárny)

Pozn.: Všechny tiskárny tisknou i na volné archy papíru.

	Lstg	e	z/s	šp	:	z/"	P	S	T
Olivetti TH760	998	40	60		+	10-15	*	*	1
Sharp CE-700p	395	32	84	10,0		10-17	V		1
Diconix 150	399	12	150	8,5	*	5-16	V	+	2
HP Thinkjet	425	12	150	8,5	*	5-18	6,8	a	2
HP Quietjet	480	12	192	9,0	*	10-12	6,8	*	2
Mannesm. Tally MT90	495	24	220	9,5	*	5-16	V	+	2
Canon PJ1080A	499	4	37	8,5		5-10	V		3
Integrex Colourjet	569	4	40	8,0		5-17	6,8		2
Siemens PT88	735	9	270	10,0	*	5-15	V		2
HP Quietjet Plus	625	12	192	15,0	*	10-12	6,8	*	2
Canon BJ130	795	48	220	16,0	*	5-16	V	+	4
Diablo 'C' Colour	890	12	20	8,5		10	V		3
Epson SQ-2500	1345	24	450	16,0	+	5-20	V	a	2

ZDOMOVA

Od ledna 1989 zajišťuje Kovo služba ve svých pražských provozovnách (v ul. Dostálových a v pasáži Černá růže) servis "spectrovských" joysticků Kovodružstva Náchod. Servis probíhá i zásilkovou formou. Drobné opravy pro své zákazníky s nevšední ochotou dosud prováděli pouze zaměstnanci prodejny Interservisu ve Spálené ulici.

Když na předvánoční trh měla do prodejny ve Spálené ulici v Praze přijít zásilka 50 ks počítačů Didaktik Gama, začala se před prodejnou tvořit fronta už v 6 hodin ráno. V 9 hodin stálo ve frontě asi 35 koupěchtivých občanů. Poslední, pětáctýřicátý se zařadil asi ve 12.30. Vzhledem k potřebné rezervě na reklamace, nemohli prodáváci uspokojit větší počet zájemců. Uvnitř prodejny mohlo z bezpečnostních důvodů čekat jen 15 lidí. Jako vždy při mimořádných událostech, i zde se ukázala síla lidské solidarity - čekající v prodejně se střídali s těmi venku (bylo sice nad nulou, ale foukal nepříjemný vítr). A personál prodejny vařil vyčerpaným zákazníkům kávu! V 16.10 konečně vytoužené počítače přivezli. Hned v 16.20

byl zahájen jejich prodej. Je nabíledni, že ten den už nemohli být obslouženi všichni, protože předvedení a vyzkoušení jednoho Didaktiku trvá zkušeným prodávacům asi 10 minut. Tento způsob prodeje se zdá být perspektivní - jednak osvěžuje sílu lidské vzájemnosti a zájemcům o výpočetní techniku umožní, aby si během své dovolené vyměnili zkušenosti a navázali četná perspektivní přátelství.

Kovo služba Praha se chystá zavést novou službu majitelům ZX Spectra. Pokud příslušná jednání dopadnou dobře, bude Kovo služba rozšiřovat RAMky ZX Specter na 272K podle návodu z Amatérského radia 9/88. Takto upravená RAMka může sloužit jako RAMdisk a zároveň umožní implementaci operačního systému CP/M. Potřebný program je součástí programové nabídky Mikrobáze. Úprava vzbudila velký zájem na osmibitovém konkursu, který se konal vloni na podzim v Brně. Cenu paměti bude zřejmě nutno uhradit v TK, ostatní v Kčs. Předběžné cenové kalkulace jsou přijatelné.

Daniel Meca

STŘEDISKO VTI PRO ELEKTRONIKU



ELO DE 01/88

Fórum čtenářů [4] Železniční modely: digitální řízení [6] Energie z větru [14] Satelity ve světových a evropských službách [23] První počítače [28] Evropská raketová startovací rampa u rovníku [32] Optické lokální sítě: světlo řídí stroje [37] Zkoušky satelitního navigačního systému [37] Sluneční baterie obstojí v náročných zkouškách [38] Propůjčení ceny za inovaci v elektronice [38] Laser v lékařské technice [39] D2-MAC: jeden čip se postaral o rozruch [40] Německé počítače pro Dálný Východ [40] Matematika pro elektroniky [41] Současná technika elektronek [46] Obsah časopisu ročníku 1987 [50] O výstavě Hobby-tronic 88 [57] Zprávy z výstavy Hobby+Elektronik 87 [58] Porovnávací test přenosných radiopřijímačů sony Pro 80 s Pan Crusade [60] Veletrh IENA 87 [65] Nové výrobky [66] Informace o novinkách rozhlasového vysílání ve světě [71] Knižní novinky [72] Profesionální vzdělání: inženýr pro výrobu řízenou počítačem [76] Elektronický dálkový spínač [78] Železniční modelářství: hlášení obsazené koleje, konstrukční návod [80] Hifi technika: programovatelný ukazatel vybuzení, konstrukční návod [82] Zajímavé integrované obvody: řízení topení [87] Desky tištěných spojů a jejich osazení součástkami pro konstrukce z časopisu [89] Program pro vytváření barevných pruhů počítačem [93] Vytváření různých kódů pomocí paměti EPROM [94] Překódování dat pro přenos z C64 do PC [98]

ELO DE 02/88

Fórum čtenářů [4] Arsen a špičková technika polovodičů [6] Dějiny počítačů [14] Biologické snímáče: nové oči pro elektroniku [22] Profesionální vzdělání: inženýr pro expertní systémy [28] Databanky: počítač ve francouzských kláštřích [30] Nejmenší paměťová buňka na magnetickém disku [30] Nový robot pro vesmír [31] Počítač čistí odpadové vody [31] Nové přenosové počítače pro frankfurtskou policii [32] Netopýr s vysílačem na zádech [33] Reportáž o tom, jak vznikají osciloskopy [34] Zpracování dat: z bitů se stává byte [40] Současná technika elektronek [42] Nová televizní norma D2 - MAC [44] Stále lepší televizní přijímače signálu ze satelitu [49] Paměťová část pro osciloskopy [52] Přenosné přehrávače kompaktních desek [58] Nové výrobky [62] Knižní novinky [66] Informace o novinkách rozhlasového vysílání ve světě [67] Konektor jako zdroj závad [75] Automatika určení měřicího rozsahu digitálního voltmetru - konstrukční návod [76] Hlášení výpadku napájení [80] Zajímavý integrovaný obvod: SG1548 [81] Desky tištěných spojů a jejich osazení součástkami pro konstrukce z časopisu [83] Generování a měření frekvencí v laboratoři, konstrukční návod [88] Paměťová média pro velká množství dat [96] Analogo-digitální konvertor pro ZX Spectrum [99]

ELO DE 03/88

Fórum čtenářů [4] Superpočítač v akci [6] Mramor pro hifi reproduktorové soustavy [14] Jak pracuje srdeční stimulátor [22] Supravodivost: vodivost, která nastává při ochlazení [26] Obrazový telefon se stane skutečností [32] Chybné užití počítačů v

hotelových službách [32] Osvětlení v domácnosti [33] Nápad pro automobil zítřka [33] Sluneční články pro plavce kolem světa [34] Malý zájem o zpracování dat přináší nebezpečí [34] Výstava přístrojů Heinricha Hertze [34] Aktuální technika elektronek [36] Matematika pro elektroniky [38] Poptávka po specialistech v oboru zpracování dat [42] Optické paměti rychle jako světlo? [44] Fotoaparáty s automatickým zaostřováním [46] Referát o výstavě CeBIT [52] Nové výrobky [54] Knižní novinky [58] Informace o novinkách rozhlasového vysílání ve světě [59] Generování a měření frekvencí v laboratoři (2) - konstrukční návod [68] Širokopásmový zesilovač pro krátké vlny - konstrukční návod [74] Zajímavý integrovaný obvod: LT1054 [79] Zkoušení operačního zesilovače [81] Jednoduchý analogo-digitální konvertor: Schmittův klopný obvod [82] Počítač řídí relé výstupem Centronics pro tiskárnu [84] Změň počítačových jazyků [86] Desky tištěných spojů a jejich osazení součástkami pro konstrukce z časopisu [89] Jak se měří napětí [91] Bezpečné napětí - 50 V, 42 V nebo 25 V? [92]

ELO DE 04/88

Fórum čtenářů [4] Elektronika v závodním automobilu [6] Předpověď počasí z počítače [12] Továrna budoucnosti řízená počítačem [22] Světlovodný kabel pro spojení kontinentů [28] Počítač bez daní [28] Nerušený příjem rozhlasu VKV v automobilu [29] Nejrychlejší škola - škola ve vlaku [29] Světový rychlostní rekord dráhy na magnetickém polštáři [30] Walter Bruch - průkopník televize [30] Pošta sází na satelity [31] Zemědělec používá mikroelektroniku [31] Matematika pro elektroniky [32] Současná technika elektronek [36] Profesionální vzdělání: technik pro lékařské přístroje [40] Mekka elektroniků: Hobby-tronic 1988 [42] Elektronické experimentální stavebnice [44] Nové výrobky [50] Knižní typy [54] Informace o novinkách rozhlasového vysílání ve světě [55] Univerzální nízkofrekvenční zesilovač [64] Zapisovač měřených charakteristik, konstrukční návod [66] Pokojová anténa na míru, konstrukční návod [68] Pomocný zvonek telefonu, konstrukční návod [72] Zajímavý integrovaný obvod: DP 8000-2 [75] Generování a měření frekvencí v laboratoři (3), konstrukční návod [80] Desky tištěných spojů a jejich osazení součástkami pro konstrukce z časopisu [87] Elektronická kontrola napětí nízkofrekvenčních zesilovačů [89] Změň počítačových jazyků (2) [90]

ELO DE 05/88

Fórum čtenářů [4] Zkrocená energie Slunce [6] Televize digitálně [14] Nový informační systém německých železnic [22] Evropa buduje novou vesmírnou hvězdárnu [22] Výkonný počítač kontroluje radioaktivní odpad [23] Elektronický zámek dveří automobilu [24] Iontový paprsek pro opravu čipu [24] Nová komunikační služba pošt [25] Soukromé vysílače v NSR [26] Výuka a učení s pomocí počítače [34] Současná technika elektronek [40] Vysokofrekvenční technika pro začátečníky [42] Výuka pomocí počítače - přehled výukových programů [48] Informace o novinkách rozhlasového vysílání ve světě [54] Přehled přijímačů signálu televize ze satelitu všech cenových kategorií [55] Nové výrobky [65] Knižní typy [68] Profesionální vzdělání: návrhář grafiky pomocí počítače [70] Zdroj konstantního proudu v širokém napěťovém rozmezí [74] Zajímavý integrovaný obvod: SLE 4501 [75] Televizní přijímač třídy hifi, konstrukční návod [78] Generování a měření frekvencí v laboratoři (4), konstrukční návod [88] Jeden čítač pro všechny druhy použití [94] Desky tištěných spojů a jejich osazení součástkami pro konstrukce z časopisu [97]



Software

PROGRAMOVÁ NABÍDKA



Pokyny k objednávání programů

Nabízené programy si zájemci objednávají výhradně na korespondenčních listcích adresovaných na 602. ZO Svazarmu, Wintrova 8, 160 41 Praha 6. Programy zasíláme na dobírku, je ale možný přímý nákup ve středisku VTI v Martinské 5, Praha 1.

Programy základní nabídky pro ZX Spectrum

Dr.MG 135 Kčs

Upravená verze spojených programů GENS 3 a MONS 3.

Datalog 186 Kčs

Databázový program, který svým komfortem v mnoha směrech převyšuje obdobné programové produkty. Pracuje s českou a slovenskou abecedou.

uB-PASCAL 205 Kčs

Prostředek pro editaci, překlad a běh programů, vhodný i pro výuku programování.

CP/M 191 Kčs

Vstupenka do světa profesionálních osmibitových počítačů; možnost využívání množství programů, které jsou tímto systémem řízeny. Instalace vyžaduje hardwarovou úpravu počítače (AR/A 10/1988).

Assembler 80 198 Kčs

Původní program, výkonný pomocník při programování ve strojovém kódu.

BASIC S 119 Kčs

Výukový program určený hlavně začátečníkům. Seznamuje s hlavními zásadami programování.

PROFESOR II 120 Kčs + kazeta za maloobchodní cenu

Univerzální výukový program, základ pro instalaci dodávaných znalostníchází typu STUDENT z nejrůznějších oborů.

STUDENT 1 96 Kčs + kazeta za maloobchodní cenu

Pět znalostníchází pro program PROFESOR (Města v ČSSR, Evropská pohoří, Světová moře a oceány, Slovní druhy, Souhvězdí).

STUDENT 2 96 Kčs + kazeta za maloobchodní cenu

Pět znalostníchází pro program PROFESOR: Naše pohoří, Významné vrcholy, Města světa, Křižovatky (dopravní výchova), Malá násobilka (pro děti).

TESTEDITOR 418 Kčs + kazeta za maloobchodní cenu

Program pro vlastní tvorbu znalostníchází typu STUDENT. Práce nevyžaduje znalost vnitřní struktury programu.

MULTITASKING 99 Kčs + kazeta za maloobchodní cenu

Operační systém pro ZX Spectrum, který umožňuje současný běh více programů na jednom počítači. Jeden z programů je v popředí, druhý v pozadí; je možné je libovolně prohodit. S programem dostanete ukázkovou pomůcku "Kalkulátor a zápisník".

GROS 79 Kčs + kazeta za maloobchodní cenu

Grafický rozhodovací systém je účinný prostředek pro podporu rozhodovacích procesů. Umožňuje volit nejvýhodnější z několika možných variant řešení daného problému.

ODA 110 Kčs + kazeta za maloobchodní cenu

Osobní databázový systém s jednoduchým a názorným ovládním. Kromě obvyklých editačních a vyhledávacích možností dovoluje aritmetické výpočty, volbu formátu zobrazení a tisku.

PROGRAF 89 Kčs + kazeta za maloobchodní cenu

Prostorové grafy jsou programem pro názorné zobrazení prostorových funkcí s možností odstraňování zakrytých částí a s bohatou volbou parametrů (úhly natočení a nadhledu, počty řezů apod.).

TEMPERAMENT 49 Kčs + kazeta za maloobchodní cenu

Zábavný psychologický test k určení typu temperamentu. Můžete se dozvědět, "kdo" vlastně jste, kam můžete zařadit své příbuzné a známé.

STOPKY 110 Kčs + kazeta za maloobchodní cenu

Časoměrný program s editorem pracujícím v paralelním režimu. Tento multitasking umožňuje zároveň odečítat časy a editovat záznamy o jednotlivých závodnicích nebo jiných měřených jevech. Vhodný pro různé sportovní soutěže nebo pro měření, když je zapotřebí sbírat velké množství časových údajů doplněných komentáři.

SONDA 4D 49 Kčs + kazeta za maloobchodní cenu

Zábavný program, který vám umožní cestu do fantastického světa čtvrtého rozměru. K programu lze přistupovat různým způsobem - od pohledu průzkumníka, který se pouze seznamuje se zajímavým prostředím, až po úroveň hlavolamu.

Nabídky z minulých čísel zůstávají v platnosti

Mikrobáze Vám nabízí originální výukový systém

*** PROFESOR II ***

vybavený mnoha výukovými databázemi.
Podrobnosti v programových nabídkách Mikrobáze.

DOPRAVNI
1986 SITUACE
c M. Stepanek

Jako první
Projede

a) 1+3
b) 4+2
>c) 1+4+6

Správně:

Zdenicka ♦♦
Dan ♦♦
>Mirek ♦

SPACE-posun kurzoru, ENTER-vyber
Mikrobáze

***** MALA NASOBILKA *****
c Martin Stepanek 1986

Kolik je: 4 x 3 =

a) 9
>b) 14
c) 12
d) 10
e) 16

Karel ♦♦♦
Radek ♦♦♦
>Roman ♦♦♦
Jirka ♦♦♦

SPACE-posun kurzoru, ENTER-vyber
Mikrobáze

>Ales ♦♦♦♦
Libor ♦♦♦

MORE
+
OCEANY

Stepanek
c 1986

Amundsenovo more
Arabske more
>Stredozemni more
Tichy ocean
Amundsenovo more

SPACE-posun kurzoru, ENTER-vyber
Mikrobáze



Ve dnech 22.-29 listopadu se v Příbrami konala jubilejní, již 20. celostátní přehlídka technické tvořivosti v elektronice a radioamatérství - ERA 88. Jak dokládají naše snímky, byla zde řada exponátů z oboru výpočetní techniky. Po celou dobu výstavy měli návštěvníci možnost si vyzkoušet práci s počítačem. Zvláště mezi mládeží byl obrovský zájem. V samostatné expozici seznamovala pražská 602. ZO veřejnost se svou bohatou činností, zaměřenou nejen na výpočetní techniku. Výstavy i doprovodných akcí se zúčastnily delegace branných organizací BLR, MLR, NDR a SSSR.

