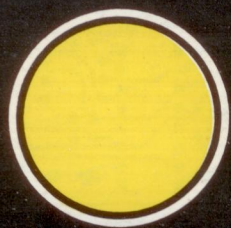


Cena 20 Kčs

MIKROBÁZE



hobby

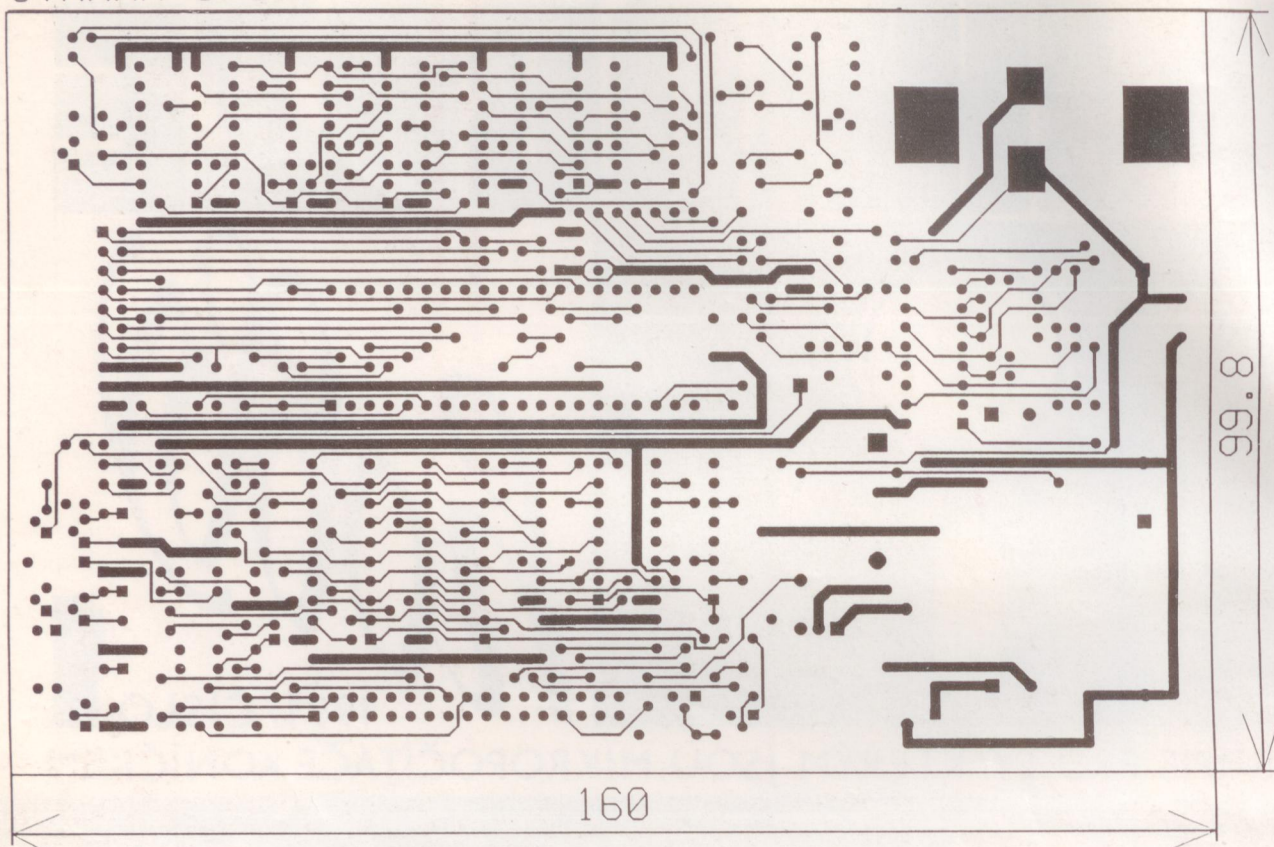
1

1991/ČÍSLO

+ ČASOPIS PRO TY, KTERÝM JSOU MIKROPOČÍTAČE KONÍČKEM



STRANA SPOJU

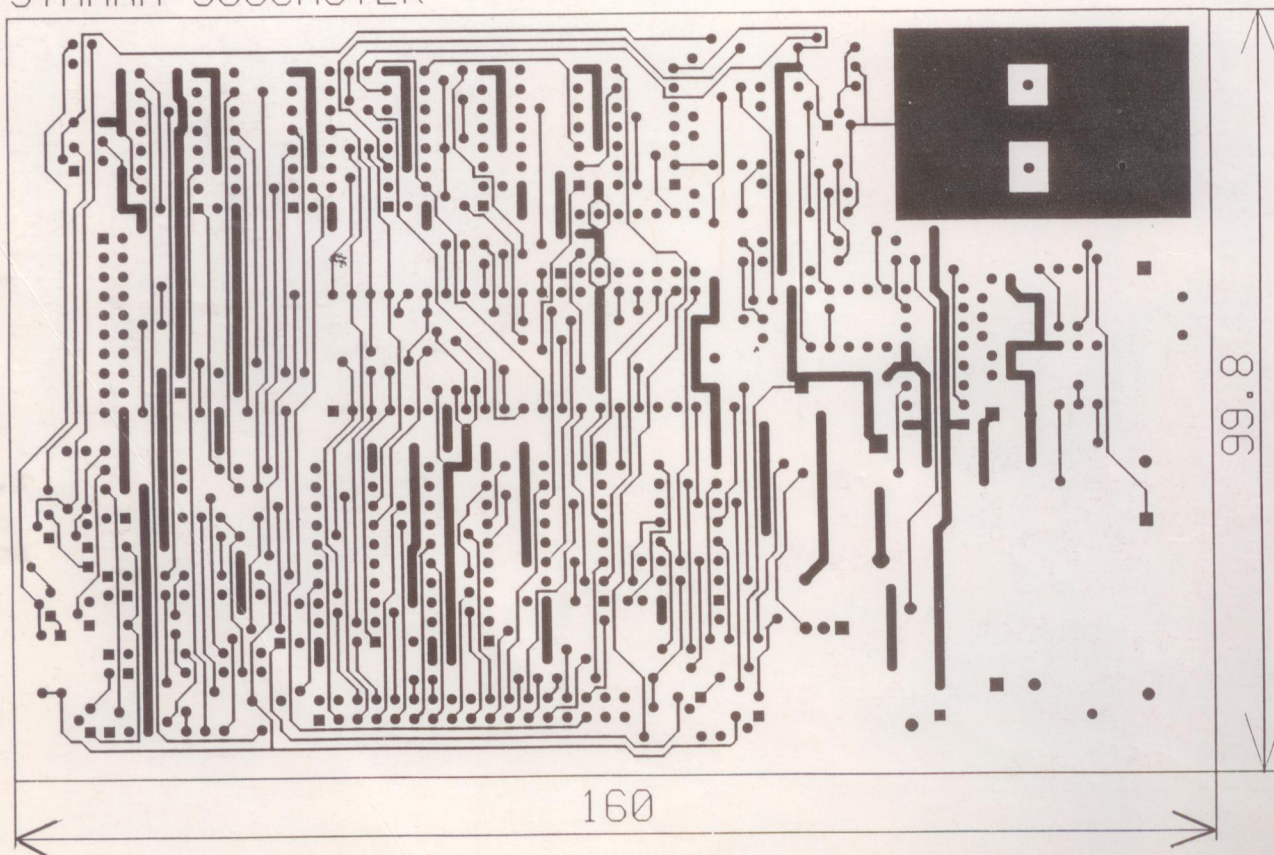


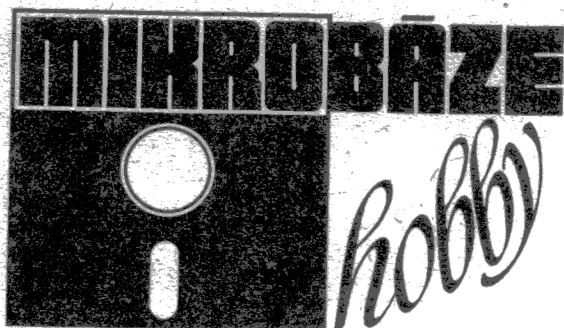
Obrazec plošných spojů programátoru EPROM
podle popisu v tomto čísle.

POHLED
ZE STRANY SOUC.

Mikrobáze hobby 1/91

STRANA SOUCASTEK





1991/1

OBSAH

Prátele (místo úvodníku)	1
Milí čtenáři (úvodník)	2
Přijem teletextu pomocí osobního počítače (software) (M. Matyska)	3
RAMdisk pro SHARP MZ-800 (M. Mrazík)	6
Postavte si s námi diskový řadič (7) (D. Meca, H. Urbanec)	8
Rozšíření paměti BOBO 64 na 256KB (Ing. V. Daněček)	12
Přímo z bloku (P. Perutz)	14
Recenze (D. Meca)	15
Uděláme si katalog (D. Meca)	15
Číslicové obvody TTL (1)	16
Opět programátor EPROM! (Ing. J. Hrejša)	18
Programovací jazyky a jejich porovnání (1) (Ing. R. Pecinovský, CSC.)	28
Nabídka Mikrobáze hobby	31

1. str. obálky: Daniel Meca ml.: "Táto, to by byla hračka!" - foto B. Zmítka

2. str. obálky: Obrázec plošných spojů na programátor EPROM.

3. str. obálky: Obrázec plošných spojů na "Commander" a držák patice k programátoru.

4. str. obálky: Naše reklamní letáčky vám nelhaly! (některé z chystaných konstrukcí).

Mikrobáze hobby - časopis pro ty, kterým jsou mikropočítače koníčkem. Vydává Daniel Meca. Adresa redakce: Jihlavská 76, 140 00 Praha 4. Šéfredaktor, redaktor, technický redaktor, grafik a vedoucí vydání: Daniel Meca. Sekretářka redakce: Zdeňka Válková. Redakční radou jsou všichni čtenáři, kteří napíší své názory. Za původnost a věcnou správnost příspěvků zodpovídají autoři.

Ročně vyjde 6 čísel. Cena výtisku 20 Kčs, roční předplatné 120 Kčs. Objednávky přijímá a expedici zajišťuje redakce. Rukopisy budou vráceny jen pokud budou vyžádány a bude k nim přiložena ohrančená obálka.

Texty byly zpracovány textovým editorem Text 602.

Tiskne PRAGOPRESS Praha.

Mezinárodní indexové číslo: 46 898

© 1991 Daniel Meca

PRÁTELE

Dovolují si vás takto prostě oslovit, protože v přenosech z Federálního shromáždění jsme se už navázili všichni až dost. Ostatně myslím, že důležitější je, jak vás osloví náš časopis.

Obvyklý úvodník v neobvyklé délce najdete na straně 2.

A teď už k tomu, co najdete v *Mikrobázi hobby*. O tomto čísle už není třeba mluvit - kostky byly vrženy. V dalších číslech se dočtete zajímavé podrobnosti o počítači Sam Coupé, povíme si něco o interfejsích pro tiskárny, o problémech s češtinou na osmi i šestnáctibitových počítačích, najdete zde pokračování seriálu o diskových řadičích, podrobný popis systému CP/M a MS DOS, včetně využití systémových služeb. Budou zde návody na hlasový výstup z počítače, na mluvčí sondu, na zajímavé využití ZX Spectra, návod na jednoduchý i složitý digitizer, kartu do IBM PC pro spojení s děrovačem, kartu CP/M, kartu optického spojení, ZX Spectrum 1MB a další. V průběhu ročníku získáte (s malou pomocí čtenářů) jeden z nejuplněnějších katalogů číslicových obvodů TTL a CMOS, referenční příručky programů WordStar, Turbo Pascal, dBase a dalších. Dozvíte se řadu informací o grafických kartách pro IBM PC, chystáme se otisknout schema motherboard do PC XT. V rubrikách *Přímo z bloku*, *Maličkosti v Assembleru* (Pascalu, Céčku, Basicu), *Začínáme s IBM PC* a dalších naleznete inspiraci a poučení. Připravujeme další recenze tuzemských programů a zajímavé pojednání o logických polích PAL a GAL.

Mnohé jistě zaujme naše pravidelná poradna, kam můžete posílat nejrůznější dotazy na softwarové i hardwarové problémy. Pro tuto rubriku se podařilo získat takové spolupracovníky, jako je ing. Václav Daněček, ing. Jan Hrejša, Jiří Lamač, Tomáš Laška, Michal Matyska, ing. Rudolf Pecinovský CSC., Herbert Urbanec a další. To by mělo být zárukou kvality. Proto pište své dotazy.

To však zdaleka není všechno, řeč zde byla jen o příspěvcích, které jsou hotové, nebo v určitém pokročilejším stádiu rozpracovanosti. Je tu však ještě celá řada nápadů a námětů. Napište nám, co by vás zajímalo a co by se podle vás mělo na stránkách našeho časopisu objevit. Kdo víte o nějakém zajímavém hardwarovém, nebo softwarovém projektu, napište nám. Dobré tipy odměníme, zajímavé nápady mohou naši spolupracovníci rozpracovat a dovést do finále. Kluby a podnikatelé v oboru počítačů, ozvěte se - pochlubte se výsledky své činnosti. Popis vašeho výtvaru v *Mikrobázi hobby* bude pro vás zároveň tou nejlepší reklamou. Nabídněte své programy redakci k recenzi, případně k šíření. Dejte vědět o zajímavých službách, které vy, či někdo ve vašem okolí provádíte. Rádi se na vás přijedem podívat a poinformujeme čtenáře. *Mikrobáze hobby* je tu proto, aby sloužila čtenářům.

Na vaše dopisy se těší váš

D. Meca
Daniel Meca
vydavatel

Jenže - a to je změna - nabídkou článků zdaleka nekončí vše, co jsme pro vás připravili. Jak jsme slíbili v propagačních letáčkách, *Mikrobáze hobby* pro vás přináší také celou řadu u nás (zatím) nebyvalých služeb. Že to bylo míněno vážně, o tom se můžete přesvědčit, když si nalistujete stranu 31.

Když jsem před drahnou dobou s těžkým srdcem psal poslední úvodník do posledního čísla zpravodaje Mikrobáze (předchůdce tohoto časopisu), ještě jsem ani netušil (nebo snad v hloubi duše už trochu tušil, ale moc na to nevěřil), že vás jednou budu moci opět takto oslovit, tentokrát dokonce ve vlastním soukromém časopise. Ne že by bylo mou jedinou touhou a ctižádostí mít vlastní časopis, či alespoň být šéfredaktorem nějakého časopisu. Jen jsem měl velikou chuť pokračovat ve své práci, která se mi zdála být snad i posláním. Ano, myslím si, že je to tak trochu poslání, pomáhat v šíření znalosti a vědomostí, zprostředkovávat výměnu zkušeností a vůbec být nápomocen i při vzájemném poznání lidí společných zájmů.

Když je tím společným zájmem tak uslechtilý koníček, jako jsou mikropočítače, stojí to snad i za jisté oběti. Snažil jsem se proto spolupracovat s některým časopisem, který by mi, třeba i v nadávku, umožnil připravovat pro čtenáře rubriku pro mikropočítačové fanoušky. Jenže to nebylo nic jednoduchého. Všechny stávající i nově vznikající časopisy si kladly jiné cíle. Všechny hnala touha po světovosti a rychlých ziscích. V horších případech se specializovali v podstatě na napodobování zahraničních časopisů a to i za cenu častého používání překladů a převzatých (v jednom případě dokonce kradených) informací. V lepších případech šlo vlastně o české redakce zahraničních časopisů. V těch nejlepších (Bajt) sice jsou původní a zajímavé články, ale zaměřené čistě softwarové a někdy na můj vkus až dost filosofující.

Většina počítačových časopisů se snaží až příliš rychle vstoupit do Evropy, takže se například zabývají testy laptopů s plazmovými displeji v cenových kategoriích kolem 10 000 USD a podobně. Většinu z nich uniká, že náš člověk má většinou v tomto oboru poněkud prozačtější problémy - třeba jak připojit BT100 k ZX Spectru, lacinou výprodejní tiskárnu Robotron k čemukoli, disketovou jednotku k bazaru k čemukoli a podobně. Všechny naše dosavadní počítačové časopisy jsou zaměřeny spíš na uživatele (nebo snad přímo spotřebitele) počítačů. Prostě - pro amatéry nic! Vlastně - skoro nic, nesmíme zapomenout na Amatérské radio a jeho osm zelených stránek.

Ted mi dovolte malou odbočku. Mnohdy se o amatérech a amatérismu mluví hanlivě. Řada lidí si totiž plete amatéra s diletantem. Slovník cizích slov však jednoznačně říká, že amatér je ten, kdo provozuje určitou činnost ze záliby. K tomu přidává vysvětlivku, že je to opak profesionála. Tak to se mi nezdá - proč by nemohl profesionál dělat svoji práci ze záliby? V oblasti výpočetní techniky je to dokonce velmi častý případ. Aby tedy nedocházelo k omylu, dostalo se nakonec do názvu tohoto časopisu slovo hobby - koníček. Ale to jsem trochu předběhl události.

Zkrátka a dobře, moje anabáze po redakcích počítačových časopisů se nakonec minula účinkem. K tomu snad jen malou perličku. Šéfredaktor jednoho z nových počítačových časopisů "velkého" světa se sice nechtěl zahazovat s amatéry, nebo dokonce s osmibitovými počítači, ale na druhou stranu se mu líbilo, že bych si jako věno přinesl svých 10 000 čtenářů. Vymyslel tedy (vlastní hlavou) toto šalamounské řešení. Čtenářům slíbíme rubriku pro amatéry (včetně těch osmibitových) a až si zaplatí předplatné, tak s tím do ztracena přestaneme. Při odchodu z této redakce jsem možná trochu poškodil dveře - já jsem dlouho klidný, ale když se naštvu ...

Tehdy jsem začal být "nalomený" na založení vlastního časopisu. Dorazilo mne však, když jsem po (velmi opožděném) vydání posledního čísla Mikrobáze dostal v krátké době hromadu dopisů (skladují si je ve velké krabici od bot), ve kterých mi řada čtenářů děkovala za dosavadní práci a projevovala litost nad ukončením vydávání Mikrobáze. Ještě dnes se mi svírá hrdlo, když si na to vzpomenu. Tehdy jsem, místo dosavadního shánění redakce či vydavatele, začal shánět půjčku, tiskárnu a všechna potřebná povolení a registrace.

Kdo to nezažil, nevěřil by, co to dá za běhání a postávání v různých frontách. V krátké době jsem projezdil asi 300,-Kčs v pražské MHD (Pražáci pochopí, že nejde o ty peníze), prostál kolem 30 hodin ve frontách na úřadech a tak jako v pohádce jsem bezmála prochodil boty. Celou tu dobu jsem se utěšoval tím, že teď už je to jednodušší a že nás stát dělá vše pro rozvoj soukromého podnikání. A je to celkem pravda. Velká část zastaralých zákonů a nařízení byla zrušena (i když se pravda na některé přeci jen zapomnělo), mnohdy dokonce bez náhrady. Mnoho nových zákonů a nařízení bylo schváleno, leč nedoručeno na příslušná místa. V mnoha případech se mi na kompetentních místech dostalo následující rady:

Mikrobáze hobby 1/91

Dělejte to jak uznáte, podle nejlepšího vědomí a svědomí, když to budete dělat v dobré víře, tak se vám snad nic nestane, kdyby se pak ukázalo, že to mělo být jinak.

To víte - hned jsem hledal skálu, do které bych ta moudrá slova vytesal - v Praze je však v poslední době mnoho podobných rad a málo skal.

Jak je vidět na příkladu malé privatizace, k podnikání u nás obvykle, kromě pár razítek, stačí už jen nějaký ten milionek z úspor a chuť do práce. To první jsem tedy získal, to poslední jsem zatím neztratil, takže schází jen to prostřední. Jelikož nejsem ani veks-lák, ani stará struktura, podařilo se mi za třídvalet let práce našetřit na Trabanta, což je sice vozidlo z valutové oblasti, ale při jeho stáří (a nejen proto) se moc výhodně prodat nedá a ani spořitel-na mi ho nepočítá jako záruku na úvěr. Pak mám sice ještě dvě počíta-če, postarší Euro PC a ZX Spectrum, ale ty jednak potřebuji ke své profesi, jednak je nedám. Půjčka do podnikání tedy není tak jedno-duchá záležitost. Nakonec se to přeci jen nějak podařilo.

Další martyrium mě čekalo kolem tisku. Jak známo, ceny v polygra-fickém průmyslu v poslední době stoupají a stoupají ... S cenou papí-ru to není o nic lepší. Prošel jsem mnoho pražských i mimopražských tiskáren, včetně soukromých, dostal jsem mnoho slibů, které však brzy byly vzaty zpět, zkrátka "vzrůšo". Ted sice nakonec tiskárnu mám za-jistěnou (alespoň se to tak jeví), ale ještě pořád neznám to sladké tajemství - kolik bude vlastně výroba Mikrobáze hobby stát. Místo předkalkulace se spíš dozvídám různé prognózy a více, či méně katas-trofické scénáře. Tak snad něco mezi 8 a 15Kčs - ale kdo ví. V noci se budím zbrocený potem, známí se mi smějí, přátelé mě litují. K žá-dosti o půjčku jsem musel přiložit ekonomický rozbor svého podnikání. Je naprostá groteska (pro nezúčastněné), když se přesně počítají ne-přesná čísla, obhajují se procenta zisku a podobně, aby se nakonec výsledek vynásobil známou kouzelnou konstantou sórok, našťásti známou ještě ze školních let, takže konečný výsledek vypadá dobře. Bohužel, konstanta sórok, jak mnozí vědí může nabývat prakticky libovolné hod-noty v oboru kladných i záporných, reálných i nereálných čísel.

Jak jsem se tedy nakonec dopracoval k ceně Mikrobáze hobby? No tak - původní Mikrobáze stála 12,-Kčs. Protože to však byl jen klubový zpravodaj Svazarmu, musel každý předplatitel zaplatit ještě klubový příspěvek 25,-Kčs ročně. Bylo deset čísel do roka, takže jedno číslo stálo vlastně 14,50Kčs. Velký zisk to neneslo, ale zase se na tom ži-vilo dost lidí. Při současném velkém zdražení všech novin a časopisů je tedy naděje, že za 20,-Kčs se jedno číslo ještě prodá. Při výrobní ceně 15,-Kčs už nezbyvá na honoráře a mzdy, těch 5,-Kčs padne prak-ticky na expedici a režii. Při výrobní ceně 8,-Kčs by na mzdy a hono-ráře zbylo, dokonce by byl i určitý podnikatelský zisk, jenže to je asi utopie. Zkrátka - držte mi palce. A koho časopis zaujal - před-platěte si ho prosím. Když si budete kupovat jednotlivá čísla, je to sice nakonec skoro totéž, ale já budu po celý rok v existenční nejis-totě a mohlo by se to projevit i na kvalitě obsahu. To přece nechceme nikdo.

Ještě malá perlička k expedici. PNS si účtuje za expedici neuvěři-telných 22,5% ceny, tedy v našem případě 4,50Kčs. Navíc jsou s ní velmi špatné zkušenosti. Proto jsem zvolil cestu vlastní expedice. A teď zlatý hřeb. V jedné tiskárně, se kterou jsem jednal, mají údaj-ne jakýsi předpis, že tomu, kdo neexpeduje pomocí PNS, mají připočít-at 42% daně k výrobní ceně. Tak tomu říkám nenásilný nábor pro PNS. Nebo - že by staré struktury? Tuto daň by prý nemuseli účtovat, pokud budu mít povolení k obchodní činnosti. Tak ho tedy nakonec mám, i když v té tiskárně netisknu.

Na stránkách Mikrobáze hobby vás vítá

váš

D. Meca
Daniel Meca

!!! Prosím pozor - důležitá informace !!!

Protože se ukázalo, že složenky s předplatným pu-tují na účet Mikrobáze hobby běžné 3 až 4 týdny a dokonce mám podezření, že některé platby nedora-zily vůbec, bude asi vhodnější (pro urychlení celé věci) posílat předplatné na adresu redakce. Jen tak se dá zajistit, že vám bude Mikrobáze hobby zasílána včas. Vaši adresu pište čitelně. Do zprá-vy pro příjemce napište která čísla a v jakém poč-tu objednávejte. Pokud vám nedojdou objednaná čís-la, upozorněte včas redakci.

Přijem teletextu pomocí osobního počítače

Michal Matyska

Hardwér, tj. vlastní adaptér pro příjem teletextu byl popsán v číslech 7+9/89 zpravodaje Mikrobáze, který byl předchůdcem Mikrobáze hobby. Naše redakce se pokusí zájemcům zajistit tato čísla (cena 3x12,-Kčs + poštovné). Napište nám.

Koncepce programu

Hlavním úkolem programu pro příjem Teletextu je zajistit emulaci klasického přijímače Teletextu, který je určen k zabudování dovnitř TVP. Výhody, které se softwérovým zpracováním získají, plynou z konfigurace počítače.

Ideové řešení programu lze popsat následovně: opakovaně se vybírají data z teletextového adaptéru, z nich se vyberou řádky obsahující skutečné Teletext a prohlídne se její hlavička. Když je nalezena řádka obsahující záhlaví požadované stránky, jsou další řádky ukládány do bufferu. Toto ukládání se ukončí příchodem hlavičky další stránky, nebo řádky příslušející jinému magazínu. Podle názvosloví Teletextu se obsah Teletextu dělí na 8 magazínů, každý po 100 stránkách. V podstatě číslo magazínu určuje první číslici z čísla stránky. Aby byly také průběžně uspokojovány požadavky obsluhy, je ke sledování klávesnice použita přerušovací rutina. To má tu podstatnou výhodu, že v časové vypjatých okamžicích lze jednou instrukcí přerušit zakázat a získat tak trochu času na jiné potřebné věci. Po načtení jedné teletextové stránky je tato zobrazena.

A zde se setkáváme se zdánlivě neřešitelným problémem. Je totiž třeba zobrazit znaky v matici 25 x 40 znaků, vždy s možností volit jednu z osmi barev pro pozadí i kresbu znaku. Celá věc je vyřešena velkým kompromisem. Na jednom řádku se na ZX Spectru dá pohodlně zobrazit požadovaných 40 znaků. Místo obvyklé matice znaků 8 x 8 se pak zobrazují znaky v matici 6 x 8. Tím se na pozici 3 původních znaků vejdu čtyři. Takto lze zobrazit až 42 znaků na řádku. Celá věc má háček v tom, že na čtyři znaky připadají jenom tři atributy. A to se při barevných grafických kreacích redaktorů Teletextu projeví tím, že znak nemění svoji barvu na celé ploše, ale jen na 1/3 nebo 2/3. Naštěstí se těchto obrázků v Teletextu mnoho nepoužívá. Při změně barvy textu se tento efekt neprojeví, protože místo řídicího znaku pro změnu barvy se zobrazuje mezera, která tento nedostatek překryje.

V normální teletextové stránce se pro zajištění znaků používá lichá parita. Tím se omezuje množina základních znaků na 96 zobrazitelných a 32 řídicích. Pomocí speciálního řádku se tato základní stránka doplňuje znaky z druhého tzv. grafické stránky a znaky s diakritikou. Na tomto řádku je určeno umístění znaku, diakritické znaménko (1 z 16 možných) a znak se kterým se znaménko má kombinovat. Tyto řádky dekodují až teletextové přijímače úrovně 1,5. Zdálo by se že taková metoda je velmi vhodná i pro komplikované jazyky. Má však svoje technické omezení. Řádků přenášejících tyto národní znaky může být maximálně 15 a na každém se dá vymezit 13 národních znaků. Tak se dá změnit asi 22% znaků ze základní stránky.

Abych se však vrátil k zobrazování stránky. Problém jak zobrazit najednou 25 řádek na 24 řádcích Spectra je prakticky neřešitelný. Pokud si však uvědomíme, že řádek 25 se používá k zobrazení

informací o tzv. sdružených stránkách, je řešení již nasnadě. Jedná se o řádek, který obsahuje čtyři barevné texty, které odpovídají čtyřem barevným tlačítkům na dálkovém ovládní. Tím se usnadňuje listování mezi některými stránkami. Protože se řádka 25 používá jen u několika málo stránek, není v základním pohledu vidět. Pokud je tato řádka k dané stránce připojena, program to dá najevo blikajícím číslem zvolené stránky. Uživatel má nyní možnost přepnout do druhého pohledu, kdy je řádka 25 zobrazena. Aby tím nebyl porušen text, je vložena místo záhlaví. A ze zcela zřejmých programových důvodů se zobrazuje na nejvyšší řádce obrazovky, aby se celá obrazovka nemusela posunovat o řádek výš a při zpětném přepnutí zase níž. Při tomto zobrazení se přicházející záhlaví nezobrazují a tlačítka 1 až 4 reagují jako ta výše zmíněná tlačítka na dálkovém ovladači.

Ještě než se budu věnovat ukázkám programového řešení, je třeba se zmínit o koncepci zacházení s pamětí. Vzhledem k časovým nárokům je program umístěn v paměti nad 32KB, aby nebyl zdržován obvodem ULA. Není to kvůli ČTV-TEXTu, který vysílá na 4 řádcích v pulsníku, ale kvůli RAI UNO, dostupnému přes satelit, kde se teletext vysílá na 12 řádcích. Protože jsou i v ČSFR poměrně rozšířené 128K varianty Spectra (128K,+2,+3,+2A), které stránkuji paměť v horní čtvrtině, zabírá program paměť v rozsahu 08000h - 0bfffh, tedy ve třetí čtvrtině. Horní čtvrtina paměti je využita k ukládání posledně načtených stránek. Každá stránka zabírá 1KB, tedy u 48K verze máme k dispozici 16 stránek, u 128K verzi Spectra je to 96 stránek.

Oproti vestavěnému dekodéru poskytuje počítačové řešení několik lahůdek navíc. Jedná se zejména o vytištění teletextové stránky na tiskárně a možnost uchování stránek na kazetovém magnetofonu pro další zpracování, a to jak ve formě klasické spectrovské screen, tak i ve vnitřním 1K formátu pro textové editory apod..

Nyní již krátké ukázky z programu:

```
*****
; PRIJEM - rutina pro načtení dat z adaptéru do vyr. pam.
; výstup HL - začátek přečtených dat
*****

PRIJEM LD HL,PRBUFF ;zač. vyrovnávací paměti
LD C,95 ;adresa portu
LD DE,128 ;hodiny na PC7

PRIJEM_1 IN A,(C)
AND 2 ;TTXEN na bitu PC1
JR NZ,PRIJEM_1 ;čekej na nulu

PRIJEM_2 IN A,(C)
AND 2
JR Z,PRIJEM_2 ;čekej na jedničku
;konec impulsu TTXEN
LD (HL),027h ;na posl. místo ulož FC
OUT (C),D
OUT (C),E

PRIJEM_3 IN A,(31)
DEC HL ;posl. platný byte dat
LD (HL),A ;přečti byte dat
OUT (C),D ;čtu vlastní zezadu
IN A,(95) ;uložím do paměti
OUT (C),E ;hodiny do nuly
AND 4 ;přečti stav BO
JR NZ,PRIJEM_3 ;hodiny do jedničky
;testuj bit 2
;opaku
```



```
LD (LASTHL),HL ;uschovej zač. ještě
; nepřečtených dat
RET
```

```
RET
DEFB 255,255,1,1,255,0,1,255 ; 0 - 7
DEFB 255,2,1,255,10,255,255,7 ; 8 - 15
.
.
atd.
DEFB 8,255,255,5,255,14,13,255 ;240 - 247
DEFB 255,14,15,255,14,14,255,14 ;248 - 255
```

Pro maximální urychlení jsou číslo portu a vysílané hodnoty pro hodiny uloženy v registrech. Spolu s rutinou pro vyhledávání řádku je dostatečně rychlá i pro Teletext vysílající více než 4 řádky TTX v TV pulsímku.

V Teletextu se používá kódování 8 bitů do 2 bytů. Pro tyto účely je použita rutina HAMM, protože se kóduje jako dvě zcela nezávislé čtveřice bitů.

```
;*****
; RADEK - ve vyr. paměti najde vhodný TTX řádek
; výstup HL ukazatel na zač. řádku
; A číslo TTX řádku po dekodování Hamming. kódem
;*****
RADEK PUSH BC ;uschovej registr
LD HL,(LASTHL) ;dosud nepoužitá data
RADEK_1 LD A,PRBUF/256 ;vyšší byte adresy
SUB H
LD B,A ;délka prohledávané
LD C,1 ;oblasti
LD A,027h ;hledám Framing Code
CPH R ;hledám Framing Code
LD A,PRBUF/256-1
CP H ;nejsem už na konci ?
JR NC,RADEK_3
RADEK_2 CALL PRIJEM ;další data
JR RADEK_1 ;a hledám znovu
RADEK_3 CALL HAMM2 ;dekoduj podle Hamminga
JR C,RADEK_1 ;není správné
PUSH HL
LD BC,42
ADD HL,BC
LD B,A ;ulož si i registr A
LD A,(HL) ;podívej se o 42 byte
; za zač. řádku
LD (LASTHL),HL ;ulož konec použ. dat
POP HL ;obnov HL
AND A ;test byte
JR NZ,RADEK_1 ;nebyla to nula ?
LD A,B ;obnov hodnotu v A
RADEK_4 POP BC ;obnov reg. BC
RET
LASTHL DEFW 0 ;místo pro uložení
```

```
;*****
; HAMM2 - dekóduje 2 byte podle Hamminga
; vstup HL ukazatel na dvojici bytů
; výstup A dekódovaný byte a CY=0
; při chybě je CY=1
;*****
HAMM2 LD A,(HL) ;vezmi první byte
CALL HAMM ;a překóduj ho
RET C ;chyba ? ano->návrat
PUSH BC ;dočasná úschova
LD C,A
INC HL ;další byte
LD A,(HL)
DEC HL
CALL HAMM ;překóduj ho
JR C,HAMM2_1 ;chyba ? ještě pop bc
RLCA
RLCA
RLCA
RLCA ;vyšší čtyři bity
OR C ;a nižší čtyři
HAMM2_1 POP BC ;obnov registry
RET
```

V teletextovém řádku 26 se používá k zabezpečení dat ještě Hammingův kód pro tři byty. Tento obsahuje 18 datových bitů a 6 kontrolních. S takovou redundancí lze pouze detekovat 1 chybu. Proto jsem se rozhodl, že datové bity se nebudou kontrolovat, pouze se vyberou ze všech. Vzhledem k dalšímu zpracování je výstup této rutiny ve třech registrech. V registru A je 6 bitů určujících pozici národního znaku, v registru D je 5 bitů definujících diakritické znaménko a konečně v registru E je 7 bitů, které určují znak, k němuž se toto znaménko přidá.

```
;*****
; HAMM3 - dekóduje trojici byte podle Hamminga
; vstup HL ukazatel na trojici byte
; výstup A,D,E A - 6 bitů
; D - 5 bitů
; E - 7 bitů
; HL - vstup(HL) + 3
; není prováděna kontrola na správnost
;*****
HAMM3 PUSH BC ;uschovej registr
LD C,(HL) ;vezmi první byte
INC HL
RR C ;a vyber patřičné bity
RR C
RR C
RRA
RR C
RRA
RR C
RRA
RR C
LD C,(HL) ;další byte
INC HL
```

Vzhledem k tomu, že adaptér přečte na TTX řádku ještě jeden nulový byte, je v této rutině jeho hodnota testována. Tím se dá zmenšit počet řádků, které adaptér přijal navíc, nepatří k Teletextu a náhodou se jejich první dva byte dají dekódovat Hammingovým kódem.

Hammingův kód se dá spočítat pomocí operace XOR nad jednotlivými bity. Protože by to však bylo časově neúnosné, je pro dekódování použita tabulka. Na pozici x byte od začátku je hodnota dekódovaná, příp. opravená, která odpovídá kódovanému bytu x. Pokud se z hodnoty x nedá určit původní (tento kód umí detekovat 3 a opravit 1 chybu), obsahuje tabulka hodnotu 255.

```
;*****
; HAMM - dekóduje Hammingovým kódem
; vstup A
; výstup A překódované a CY=0
; A=255 a CY=1 při chybné kombinaci
;*****
HAMM PUSH HL ;uschovej registry
PUSH BC
LD HL,HAMTAB ;začátek tabulky
LD C,A ;přidej posun
LD B,0
ADD HL,BC
LD A,(HL) ;vezmi překódovanou
POP BC ;obnov registry
POP HL
AND A ;nastav flags podle A
RET P ;je <127 ? ano->návrat
SCF ;nastav příznak chyby
```



```

RR C
RRA
RR C
RRA ;A=b5 b4 b3 b2 b1 b0 0 0
SRL A ;posuň do správné pozice
SRL A
LD B,A ;a ulož dočasné
LD A,C
AND 01fh ;dalších 5 bitů
LD D,A ;rovnou do výst. registru
LD E,(HL) ;další byte
INC HL
RES 7,E ;jen 7 bitů
LD A,B ;první z výsledků
POP BC ;obnov registry
RET

```

Ovládání programu

Ovládání programu je poměrně jednoduché, a navíc lze zkrácený hlep vyvolat vždy tlačítkem **M - Menu**. Pro přímou volbu stránek se používají číselná tlačítka 0+9. Pokud je stránka vybavena časovým subkódem, dá se tento zadat pomocí **T - Time subcode**. Pro odkrytí skrytých slov je použito tlačítko **R - Reveal**. Pomocí mezerníku se přepíná zobrazování záhlaví nebo 25. řádku na horním okraji obrazovky. Pro zlepšení uživatelského přístupu jsou funkční i tlačítka **A - Automatické listování** všemi stránkami, **N - Next**, tj. přechod na další teletextovou stránku a **P - Previous**, tedy stránka předchozí. Pomocí tlačítka **C - hardCopy** se vyvolá rutina pro tisk obrázku, a tlačítka **S - Save** a **X - internal format save** lze uložit jednu stránku na kazetu, buď ve formátu klasické screen, nebo v interním formátu. Tyto stránky v interním formátu jdou dále zpět nahrát a zobrazit tlačítkem **L - Load**.

Program byl psán s normou Teletextu v ruce. To by mělo zajistit jeho plnou slučitelnost s vysíláním Teletextu. Bohužel je nutno konstatovat, že u nás např. v záhlaví stránky jednotlivé kontrolní a řídicí bity neodpovídají vždy tomu, co praví norma.

V závěru bych chtěl poděkovat pracovníkovi Výzkumného ústavu spojů ing. Kameníkovi, s jehož laskavou pomocí byl celý adaptér Teletextu zprovozněn.

Literatura: Návrh Československé státní normy
TELETEXT - Ing. Reček ČST Praha,
březen 1988.
(Vyšla též v příloze AR 1989)

Pozn. red. - Mohu vám s potěšením oznámit, že konečně došlo k dohodě mezi autorem popsaného programu a naší redakcí. **Mikrobáze hobby** tedy může svým předplatitelům nabídnout program pro ZX Spectrum na kazetě či disketě (Betadisk). Cena pro předplatitele je stanovena na pouhých 80,- Kčs + poštovné a to včetně kazety a brožurky. Verze na disketě by měla stát asi do 70,- Kčs (podle ceny diskety v době distribuce). Je možné nakopírovat program i na zaslanou disketu. Ta musí být prázdná, bezchybná a naformátovaná Betadiskem v požadovaném formátu.

Doufám, že uvedená cena je dostatečně přijatelná, aby už nedocházelo k nelegálnímu kopírování programů, které je nejen trestné podle současných zákonů, ale především je hluboce nemorální a bezohledné k autorům. Proto apeluji také na ty, kteří si tento program sami nakopírují, aby zaslali na adresu naší redakce 20,- Kčs jako odměnu pro autora. Obratem pošty za to obdrží potvrzení o tom, že jsou legálními uživateli programu. Další podrobnosti najdete na straně 32 v nabídce **Mikrobáze hobby**.
D.M.

216 SKYTEXT 251 Wed 24 May 1243:35

Tři minuty před tiskem.

Všechno je jinak!!!

Ano, tato stará moudrost se opět potvrdila. Jen jsem odevzdal přepracované první číslo **Mikrobáze hobby** opět do tiskárny, začalo se dít mnoho věcí. Když pomínu vyvalivší se problémy s daněmi (což by vás normálně nemuselo zajímat, ale jde tu o bytí a nebytí časopisu - podrobnosti příště), staly se ještě nejméně dvě věci o kterých vás musím informovat.

Nejdříve ta dobrá. Jedná se o nabídku hotových doplňků ze strany 32. Při dalším jednání s výrobcem se podařilo dohodnout dvě varianty cen. Ta původní zůstane pro podniky - platba na fakturu, půl roku záruka a pět let servis. Naši čtenáři si však zpravidla případné opravy provádějí sami, takže ta záruka a servis je tolik nezajímají. Pro ně tedy byla stanovena zvláštní cena - budeme ji říkat cena pro amatéry. Předpokládá se předložení kupónu ze strany 31 a platba v hotovosti (dobírkou), přičemž záruka je jen 10 dní. Cena ovšem je zajímavá. Posuďte sami. Programátor EPROM, podle popisu v tomto čísle bude stát 4200,- Kčs pro podniky, ale jen 2000,- Kčs pro amatéry. Karta CP/M bude za 4990,-/2500,- Kčs a karta hlasového výstupu za 4000,-/1500,- Kčs. U karty pro připojení snímače a děrovače se předpokládá čistě profesionální vy-

užití, takže je stanovena jediná cena (8000,-Kčs). Dále je také na dobré cestě jednání o doprodeji různých přebytečných součástek za výhodné ceny. Seznam a ceny můžete získat, když si pošlete obálku s adresou a dvě korunové známky.

Tak to byla ta lepší věc. A teď ta horší. Bude se to týkat jen některých předplatitelů, kteří si zaplatili předplatné složenkou na náš účet ve spořitelně. Někteří zřejmě dostali složenku zpět, některým prostě jen náš časopis nechodí. Velice se omlouváme - není to naše chyba a jestli ano, tak spočívá jen v mé důvěřivosti s jakou jsem si založil běžný účet ve spořitelně. Slibovali vzorné služby, ale opak je pravdou. Podrobnosti se dozvíte příště - nevím, zda to neskončí u soudu. Složenky zaplacené začátkem prosince nebyly ještě v půli ledna na účtu! Některé podniky mne upozorňují, že se jim platba vrátila s tím, že účet toho čísla neexistuje. Přitom jsem si nechal složenky narazitkovat přímo ve spořitelně, která za správnost čísla ručí. Mimochodem, účtovali 20hal. za kus (tj. čtyři razítka) - to je kšeftík co? A to se prý budou ještě všechny účty přepisovat. To tedy pánbůh s námi a od spořitelny pryč! Musím honem založit nový účet v bance. Zatím prosím posílejte předplatné na adresu redakce a vy postižení se ihned ozvěte.

D.M.

Every week a chance to win
#50 on Page 212
MAIN INDEX-100

RAMdisk 512KB

pro SHARP MZ-800

Miroslav Mrazík

V různých časopisech bylo uvedeno několik zapojení RAM-disku pro počítač MZ-800. Některé z nich se chovají stejně jako originální RAM-disk a mohou tedy spolupracovat s dodávaným Basicem. Řešení popsané v tomto příspěvku má zcela jinou organizaci a bylo navrženo s ohledem na několik požadavků - minimum součástek pro řízení paměti (tj. maximální využití vlastností Z80 CPU), přístup k datům prostřednictvím I/O komunikace, organizace dat vhodná pro OS CP/M (po sektorech dlouhých 128 bytů) a co nejrychlejší přenos mezi RAM-diskem a operační pamětí.

RAM-disk zabírá tři I/O adresy určené pro periferní zařízení. Tyto adresy byly zvoleny v oblasti, kterou nevyužívá zákaznický obvod. Přítomnost RAM-disku tedy ROM-monitor MZ-800 vůbec neregistruje a nemůže na něj reagovat. První dvě adresy (0F8h a 0F9h) slouží k zápisu informace o požadované stopě a sektoru. Kanálem daným třetí adresou (0FAh) lze zapsat/přečíst blok dat o délce 128 bytů.

Celková kapacita RAM-disku je 512 KB. Tato kapacita je organizována v matici 64 x 64 buněk. Každá buňka představuje jeden sektor o délce 128 bytů. Přístup k libovolnému sektoru je velmi jednoduchý. Nejprve se zadá stopa a sektor, ke kterému chceme mít přístup:

OUT (0F8h),TRK

OUT (0F9h),SEC

Tim je určen řádek a sloupec v matici buněk. Čtení sektoru provedeme následujícím způsobem. Určíme délku bloku dat a adresu periferie:

LD BC,80FAh

Dále určíme adresu v operační paměti, kam sektor přeneseme:

LD HL,DMA

Nyní přeneseme celý sektor z RAM-disku do paměti instrukcí:

INIR.

Je zde využita vlastnost mikroprocesoru Z80, který při vykonávání blokového přenosu INIR mezi periférií a pamětí posílá obsah registrů BC na sběrnici jako adresu periferie. Registr C je odeslán na spodních 8 bitů adresové sběrnice a určuje adresu periferie (0FAh). Obsah registru B je odeslán na horních 8 bitů adresové sběrnice a RAM-disk jej používá jako adresu jednotlivých bytů v dané buňce (sektoru). Obsah registru B se v uvedeném případě mění od 80h do 1 a to odpovídá adresaci uvnitř sektoru 0h, 7Fh, 7Eh, ..., 1h.

Při zápisu sektoru však není možno použít instrukci OTIR, neboť tato instrukce není inverzní k INIR. Je to způsobeno tím, že INIR nejprve čte periférii adresovanou BC, pak dekrementuje B, Zapiše přečtenou informaci na (HL), inkrementuje HL a testuje B na rovnost nule. Naproti tomu OTIR nejprve čte data z (HL), pak dekrementuje B, zapiše data na periférii adresovanou BC, inkrementuje HL a testuje B na rovnost nule. Při čtení dat je tedy vyslán na sběrnici původní obsah BC, zatímco při zápisu až po dekrementaci B. Tim by vznikl posun v datech o jeden byte. Tento posun nelze kompenzovat změnou počátečního nastavení registru B, neboť RAM-disk využívá pro adresaci uvnitř sektoru jen 7 bitů. Zápis dat je tedy nutno

obejít kombinací:

OUT (C),A

INC HL

DJNZ.

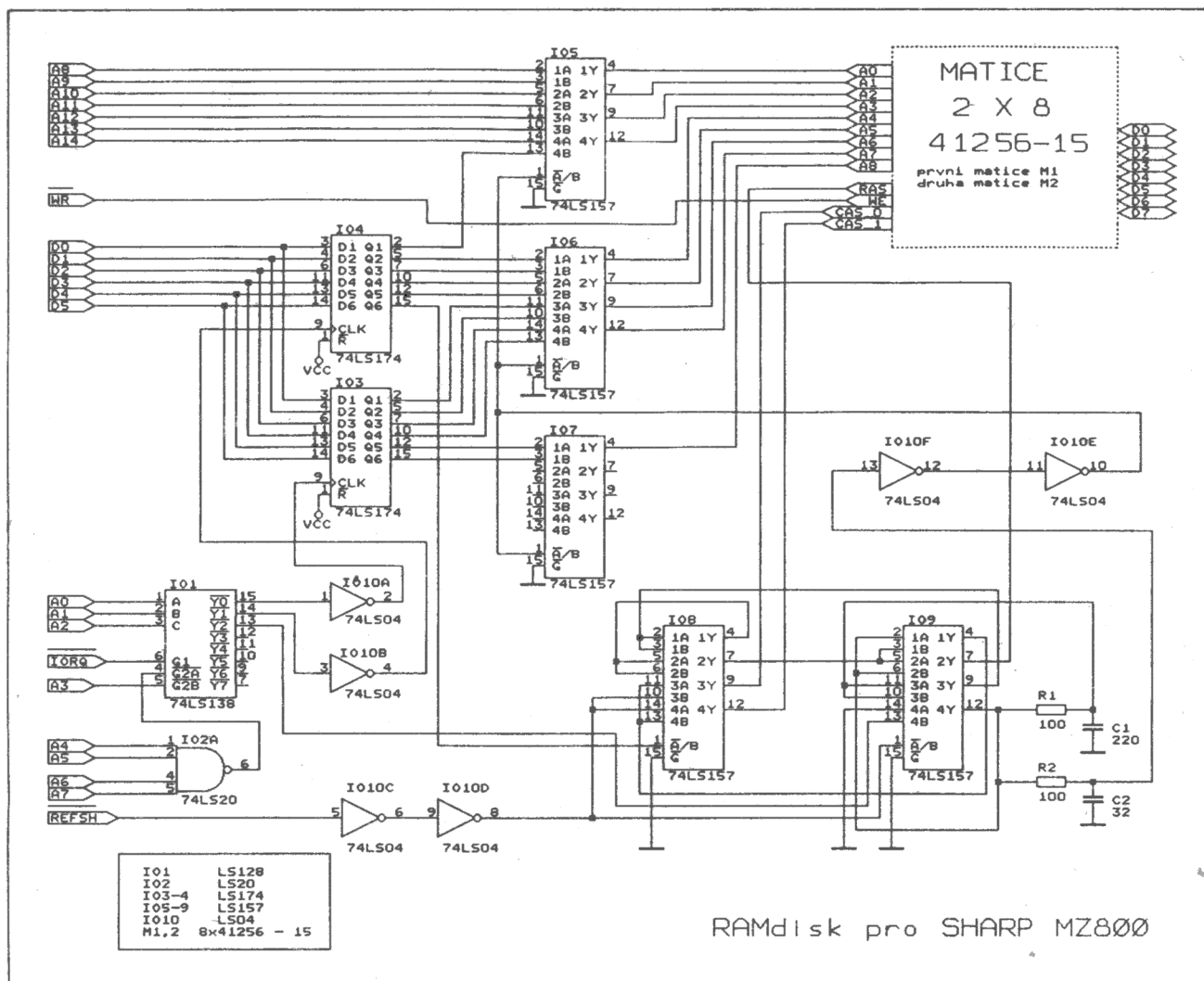
Schema zapojení RAM-disku je uvedeno na obrázku 1. Obvody IO1 a IO2 dekódují I/O adresy 0F8h - 0FAh. Výstupy 0 a 1 IO1 řídí přes H1 a H2 IO10 zápis do šestnáctibitových registrů IO3 a IO4. Výstup 2 IO1 aktivuje paměť pro zápis/čtení. K výběru informace z paměti slouží 18 adres. Prvních 7 adresových bitů tvoří horní část adresové sběrnice mikropočítače (A8 - A14). Ty slouží k výběru jednotlivých bytů uvnitř sektoru. Dalších 5 adresových bitů dodává IO4. Tim je určena stopa (track). Šestý bit dodávaný IO4 přepíná mezi sebou bloky paměti M1 a M2. Výstupy M1 a M2 jsou připojeny přímo k datové sběrnici mikropočítače. Vstupy D a výstupy Q paměti jsou po jednotlivých bitech spojeny paralelně.

Řízení paměti obstarávají dva obvody 74157 (IO8 a IO9). Impuls pro odstartování sekvence RAS a CAS pro paměti vyvolává přechod H-L na výstupu H1 IO9. Signál RAS se odebírá z výstupu H3 IO9 a signál CAS se odebírá z výstupu H4 IO9. RAS je odvozen přímo z výstupu H1 IO9, zatímco CAS je zpožděn R1, C1 a tvarovači H2 IO9 a H4, H3 IO8. Stav na vstupu A IO8 určuje, do kterého pamětového bloku bude CAS přiveden. O to se starají H1 a H2 IO8. Výběr je řízen šestým výstupem registru IO4.

Při žádosti o přístup do paměti (adr=0FAh & IOREQ) se na vstupu A1 H1 IO9 objeví úroveň L. Ta je vyslána H3 jako RAS. Se zpožděním R2 a C2 se přepnou přes tvarovače H5 a H6 IO10 adresové multiplexery. Se zpožděním R1 a C1 je přes tvarovače H2 IO9, H4, H3 IO8 přivedena úroveň L na A1 H4 IO9. Tim je generován signál CAS a ten je přes přepínač H1, H2 IO8 přiveden do jednoho z bloků paměti. Nyní je vybrán příslušný byte v paměti. Zápis/čtení je řízen signálem WR na sběrnici mikropočítače.

Pro refresh paměti je použit speciální režim CAS-before-RAS refresh. Tento režim lze popsat takto: Přejde-li signál CAS před signálem RAS, zůstane výstup paměti ve třetím stavu, adresové vodiče se přepnou na interní čítač, provede se refresh jedné řady pamětových buněk a interní čítač se inkrementuje (platí pro 41256). Je-li aktivován signál REFSH, přepne se u IO9 směr toku dat na A0 - Y. Uzemněný vstup A0 H1 IO9 simuluje žádost o přístup do paměti. Tim je vyvoláno generování sekvence RAS a CAS. Přepínač H3, H4 IO9 však tyto dva signály mezi sebou prohodí. Nejprve přejde přes úroveň L signál CAS. Ten je sice přiveden na přepínač H1, H2 IO8, ale na jeho druhé dva vstupy je přiveden signál REFSH, který klesl na úroveň L již asi před dvaceti nanosekundami. Tak je zaručeno, že se CAS objeví na obou blocích paměti nezávisle na stavu vstupu A IO8. Se zpožděním je vygenerován signál RAS.

Poznamenejme, že signál REFSH není vyveden na sběrnici v expanzní jednotce SHARPU. Je nutno jej vyvést přímo od mikroprocesoru a připojit na jeden z volných pinů sběrnice. Teoreticky by bylo možno ke stejnému účelu použít signál M1, ale zdě by mohla nastat kolize v okamžiku, kdy mikroprocesor



dá na vědomí akceptování interruptu. Tehdy se na sběrnici objeví aktivní signál IOREQ i M1 v těsném sledu. Tím by se narušilo časování paměti.

Následující výpis ukazuje možnou implementaci RAM-disku v BIOSu CP/M:

```
;Popis disku
DPBL:    DW 64          ;SECTORS PER TRACK
         DB 4           ;BLOCK SHIFT
         DB 15          ;BLOCK MASK
         DB 0           ;EXTENT MASK
         DW 251         ;DISK SIZE-1
         DW 127         ;DIRECTORY MAX
         DB 192         ;ALLOC 0
         DB 0           ;ALLOC 1
         DW 0           ;CHECK SIZE
         DW 1           ;OFFSET
```

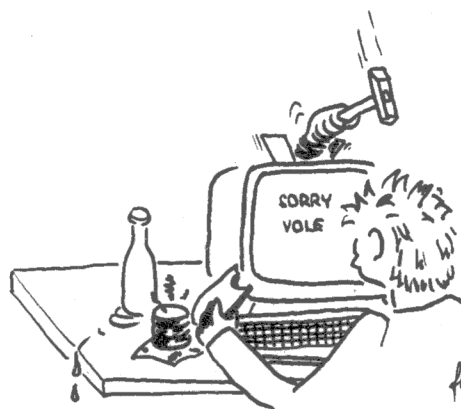
```
;Zapis sektoru
LPWRI:   LD BC,80Fah
LPWRI1:  LD A,(HL)
         OUT (C),A
         INC HL
         DJNZ LPWRI1
         RET

;Cteni sektoru (HL=DMA)
LPREA:   LD BC,80Fah
         INIR
         RET
```

Celou konstrukci RAM-disku lze umístit do prostoru, který je určen pro originální RAM-disk firmy SHARP. Napájecí zdroj má velkou výkonovou rezervu, takže její připojení RAM-disku neovlivní. Nejsou ani problémy s chlazením. V mém počítači je umístěn ještě kontroller pro floppy drive, oddělovač

sběrnice a zdroj 12V a přesto ani po několikátý denním trvalém provozu nedochází k nadměrnému ohřevu expanzní jednotky. RAM-disk je v provozu již asi dva roky a zatím se u něj nevyskytla žádná závada ani zkomolení dat.

Pokud se při stavbě dodrží zásady správného rozvodu zemí, napájení a blokování jednotlivých obvodů, neměly by při ožiování vzniknout žádné problémy. Na závěr bych rád podotkl, že RAM-disk je schopen funkce i jen s jedním pamětovým blokem. Pak se samozřejmě zmenší počet stop nebo sektorů na 32. Při předpokládané kapacitě pouze 256 KB je možno vypustit I08. Zvýšení kapacity je možno snadno provést rozšířením registrů I03 a I04 a připojením nezapojených tří hradel I05 k dalším adresovým vodičům paměti (pokud mají stejný způsob řízení).



Postavte si s námi diskový řadič

/7/

Daniel Meca, Herbert Urbanec

Po delší (díky výrobním a expedičním termínům ne zas až tak moc dlouhé) době se opět scházíme nad seriálem o stavbě a používání Betadisku, ale i jiných diskových řadičů. Nové čtenáře je nutno upozornit, že seriál začal původně vycházet ve zpravodaji Mikrobáze, který vydávala 602. ZO Svazarmu a který byl předchůdcem Mikrobáze *hobby*. Koho by zajímaly přechodní díly seriálu, ten ať napíše naši redakci - pokusíme se zajistit dodání předchozích dílů seriálu a to buď přímo posledních šest čísel Mikrobáze (cena 6x12,-Kčs), nebo jako separátní výtisk všech dosavadních dílů (cena asi kolem 20,-Kčs).

Jak už to v životě bývá, všechno je jinak, takže se obvykle vše předpokládané nepodaří, ale též naopak - nepředpokládané se podaří. Tedy napřed to (téměř) nepředpokládané. Jak dokazuje to, co právě čtete, podařilo se zajistit vydání zbytku seriálu a samozřejmě i řady dalších (jak doufám) zajímavých návodů a to formou soukromého časopisu pro počítačové "bastliře", že to nebyla zrovna procházka růžovým sadem, to jednak jistě tušíte, jednak se to můžete dočíst na jiném místě v tomto časopisu. Protože to nemá být časopis leda jaký, zavinila jeho příprava můj značný (nejen) časový stres, z čehož vyplynulo prozatímni neuskutečnění toho předpokládaného. Prostě nestihl jsem v potřebném termínu připravit k expedici některé slíbené programy (univerzální formátovací program, diagnostický program, upravená verze CP/M a některé další). Doufám, že se mi to podaří teď, po odevzdání prvního čísla do tiskárny, takže již v čísle 2 se zaskví v programové nabídce také tyto programy.

Tak tedy k vlastnímu Betadisku. Soudě podle nebývalého dopisového ohlasu a také podle počtu EPROM, které jsem se svými přáteli programoval, bude asi v nejbližší době Betadisk nejrozšířenější periferní zařízení pro ZX Spectrum.

Byl jsem upozorněn, že LECROM dokáže posunuti začátku Basicu, které bylo popisováno v Mikrobázi 10/89, udělat přímo příkazem ERASE "I", určeným původně k odstranění systémových proměnných pro ZX Interface I. Tento příkaz lze použít i v programovém řádku. ISO-ROM má sice příkaz obdobného účinku, ten však je možno použít pouze v přímém příkazovém režimu. Je to příkaz "\$".

Dále musím upozornit, že po zrušení systémových proměnných Betadisku (což je nutný důsledek posunutí začátku Basicu zpět na 23755) si při prvním zavolání TR-DOS znovu testuje rychlost krokování mechaniky a počet jejich stop.

Ve druhém pokračování seriálu jsem napsal, že při prvním startu je automaticky spuštěn program "boot". Správně mělo být uvedeno, že "boot" se automaticky spouští po inicializaci v EPROM Betadisku, tedy po resetu na řadiči.

Mnoho čtenářů si stěžovalo na špatnou dostupnost a vysokou cenu Schottkyho diod. Tedy: V době psaní návodu byly tyto diody k sehnání běžné a to za cenu asi 3,60Kčs. V případě jejich nedostatku lze použít i běžné diody, zhorší se však poněkud odolnost zapojení proti impulsnímu rušení. Je také možné nahradit vždy dvojici diod a příslušný odpor do plusu hradlem AND (74LS08). Takové řešení je z hlediska spolehlivosti a odolnosti systému vlastně nejlepší.

Je dobré dát ty IO, které přímo komunikují s mechanikou do soklů. Jedná se vlastně o IO8 na desce řadiče. Pokud se pak někomu povede otočit napájení mechaniky, není problém vyměnit tento IO, který to zpravidla odnese (na vstupy se dostane 12V).

Rozhodně lze doporučit použití patic pod EPROM a pod WD 1793.

Na desce systému jsou nevyužité inventory 1 a 3 IO1. Dají se použít na indikaci přepnutí do systému Betadisk (z vývodu 9 IO5). Dioda svítí při přepnutí do TR-DOSu. Je však nutno upozornit, že TR-DOS odkazuje do ROM Spectra na jednotlivé rutiny.

Také je nutno upozornit na zakončovací odpory v mechanikách. Pokud jsou zcela odpojeny, bude vše pravděpodobně fungovat, ale mohou vzniknout problémy ze špatného přenosu dat, zvlášť při delším kabelu. Při více připojených mechanikách je nutno ponechat zakončovací odpory jen v jedné, nejlépe poslední, mechanice. Jinak jsou výstupy řadiče přetíženy a celek obvykle nefunguje.

Vyplatí se skutečně použít konektor a ne dráty k propojení obou desek. Pokud použijete FRB a dolů (na desku systému) dáte samičku, získáte možnost připojení k jiným deskám při ožívování.

Můj přítel a jeden z prvních externích spolupracovníků redakce své poznatky se stavbou a ožíváním Betadisku dokonce literárně zpracoval, takže bude nejlepší jeho poznatky uvést v (téměř) nezměněné formě. Tak tedy Herbert Urbanec píše:

"Právě před několika okamžiky jsem objevil příčinu jedné z nejpodivuhodnějších závad řadiče Betadisku. Systém záhadně a naprosto neočekávaně padal bez jakýchkoliv zjevných příčin vždy v nejnepohodnější chvíli, navíc bez možnosti vysledovat jakoukoliv souvislost či pravidelnost. Již jsem pomalu začínal věřit na *ufouny*. K dovršení všeho měla však závada přece jenom jednu spolehlivou vlastnost. Projevila se zásadně v okamžiku, kdy jsem zdánlivě naprosto funkční řadič uvedl *zakrabíčkováním* do finální podoby, zatímco při testování samotného *bastlu* na stole chodilo vše bez závad.

Po již bůhví kolikátém sestavení a rozebrání řadiče a pro jistotu také ověření naprosto shodného chování závady i na jiných Spectrech jsem byl poněkud zdeptán a pomalu jsem začal uvažovat o nějakém jiném oboru lidské činnosti, například o pěstování mrkve. Nakonec jsem se rozhodl zanechat marného snažení a odložit neúspěšnou práci do vzdálené budoucnosti. Avšak při opětovném sestavování řadiče zpět do kompaktního tvaru jsem konečně přišel na důvod veškerého záhadného chování Betadisku. Testovaný řadič byl ve složené i v rozebrané podobě v naprosto stejném stavu až na jednu zcela nepatrnou maličkost. Vzhledem k tomu, že tlačítko *Magic button* bylo napevno zabudováno ve stěně krabičky, bylo nutno vždy při demontáži odpájet jeho přívody a řadič testovat v rozebraném stavu bez něj.

No a tato *zanedbatelná* maličkost byla příčinou všech problémů. Při pohledu do schematu je možno zjistit, že jeden vývod tlačítka je připojen na /A vstup 74LS123 a druhý je spojen se signálem /MRQ počítače. A tak věren heslu, že *na každou otázku existuje jednoduchá, snadno pochopitelná nesprávná*

odpověď, usoudil jsem bez sebemenších rozpaků, že za vše mohou příliš dlouhé přívody k tlačítku. To to přesvědčení mi vydrželo celé dva dny, než jsem objevil skutečnou příčinu výše uvedených zapeklitých problémů. Kdysi dávno, v době vzniku mého řadiče, jsem jej vybavil jediným pro mne tehdy dostupným tlačítkem. Poslední dobou se mi však jaksi znelíbila jeho zářivě rudá barva, tudíž jsem je vyměnil za obdobné v barvě zelené. A zde byl pes (či spíše přímo slon) zakopán! Neb tlačítko, co do vnější podoby až na barvu zcela shodné se svým předchůdcem, bylo totiž zákeřným kapitalistou vyrobeno jakožto rozpinací a tudíž s naprosto opačnou funkcí než důvěru ztratí starší kolega.

A jako u každé správné bajky - na závěr úvodu výchovné poučení: *ufouň* mají dozajista dost problémů sami se sebou, natož aby se ještě starali o nás, no a co se mrkve týče, ta snad poroste i bez mého cenného přispění!

Avšak nyní již vzhůru dolů k dalším problémům okolo stavby a ožiování řadiče Betadisku. Nejdříve bych se rád vrátil k poslednímu číslu Mikrobáze [6], kde se nechtěně objevila poněkud zavádějící informace. Na str. 8, pátý řádek od shora v levém sloupci - *na chování TR-DOSu by se nemělo nic změnit* - tedy pro upřesnění, nezmění se nic co do výsledného efektu libovolného příkazu, avšak doba *odezvy* na většinu příkazů je *několikanásobně delší* než bez připojené desky řadiče. To nám umožňuje otestovat funkčnost řadiče i bez připojené mechaniky (nemáme-li ji například v dané chvíli k dispozici), tedy alespoň do té míry, že se můžeme bezpečně ujistit o tom, že *žije* vlastní řadič (IO5) a dále IO4 a polovina IO2, rovněžtak o správnosti propojení obou desek. Zbytek desky řadiče je však již stejně nutno testovat s připojenou mechanikou. Máme-li k dispozici více mechanik, je lepší (alespoň ze začátku) použít pouze jednu, pokud možno již otestovanou. Taktéž se nedoporučuje příliš dlouhý kabel mezi řadičem a mechanikou (a to nejen pro testování ale posléze i pro vlastní provoz).

Dále bych se rád chvíli podrobněji věnoval otázce spolupráce LECROM (popis viz [7]) s Betadiskem. Jednak proto, že je již mezi Spectristy hodně rozšířena a celkem oblíbená, ale i proto, že ji sám od počátku s Betadiskem používám a mohu zde tudíž uvést několik zajímavých drobností, které je určitě užitečné při práci s touto kombinací znát. První a základní odlišností od původního chování s originální ROMkou Spectra (a pro některé uživatele možná tak trochu problém), spočívá v tom, že *nefunguje* příkaz TR-DOSu RETURN. Je to způsobeno jiným obsahem LECROMky oproti původní ROMce, což má za následek to, že při použití tohoto příkazu se nedostaneme zpět do Basicu, jak bychom očekávali a především potřebovali, nýbrž zůstaneme v TR-DOSu. Tento nedostatek se dá ovšem obejít jinou příjemnou vlastností LECROMky, totiž takzvaným *trojhmatem*, tedy *symbol shift + space + B*, po kterémžto se dostaneme do monitoru a odtud již příkazem Q do Basicu (viz výše zmíněný popis). Při zadávání příkazů přímo z Basicu za použití sekvence RANDOMIZE USR 15619: REM: příkaz je vše v pořádku a návrat do Basicu proběhne bez problémů.

Naprosto stejnou příčinu má i další odlišnost v chování Betadisku ve spojení s LECROMkou - nelze použít celkem výhodné funkce resetování přepínačem na řadiči (viz [2] a [6]). Při zamýšlené úpravě TR-DOSu bude však na tyto nedostatky již pamatováno. (*Mimoходом, stále by za to také upravit LECROM tak, aby správně spolupracovala s Betadiskem a ne s tak archaickým, i když dosud rozšířeným zařízením jako je ZX Microdrive. O této úpravě už redakce začala jednat s J. Lamačem. - pozn. red.*)

S tím vším také souvisí otázka úpravy dekódování volací adresy Betadisku. Tato varianta je určena pro ulehčení práce těm, kteří vlastnili před-

chozí verze Betadisku, aby nemuseli své staré programy pracně přepisovat při použití nového řadiče. Domnívám se ovšem, že problémy toho druhu jsou většinou nových majitelů Betadisku naprosto cizí. Při LECROMce (popřípadě i některých jiných ROMkách), kdy stejně nelze zmíněnou úpravu použít, je patrně nejlepší vynechat součástky určené pro tuto variantu. (*ISOROM nekoliduje - pozn. red.*) V tom případě ovšem nazapomeňte osadit propojku místo R7!

A ještě v návaznosti na to několik drobností k přepínači sloužícímu jako reset a zároveň také jako selekt pro verze 128. Zde musím upozornit na jednu drobnou chybičku, vyskytující se v zapojení tohoto přepínače. Dle schématu v [2] na str. 11 je střední poloha *poloviny* přepínače u D6 nezapojena, správně by však měl být tento vývod spojen s vývodem Grn, aby ve střední poloze nedocházelo k náhodnému startu Betadisku po resetu. (*Nejde zde o chybu ve schématu, ale o konstruktérský prohrěšek, vynucený tím, že se prodáváji velice roztomilé přepínače s nevyvedenou střední polohou - pozn. red.*) Rovněžtak z naprosto stejného důvodu je jistější při neosazení přepínače alespoň spojit na desce jeho vývody označené Blu a Grn (tedy vlastně *opravená* střední poloha), aby bylo spolehlivé zajištěno, že po každém resetu opravdu skočí program do Basicu a ne do Betadisku. Hlubavější povahy již dozajista přišli na důsledky, které má tato *náhodná volba* ve spojení s LECROMkou. Tu a tam prostě Spectrum po resetu a hlavně téměř vždy po zapnutí jaksi zabloudí a nám nezbyvá než znovu použít *poslední pomoc* resetu k nápravě situace. Po tomto úkonu však opět záleží na vůli boží, na čí stranu se vlastně přikloní a veškerá naše snaha se pak spíše podobá populární hře sudá - lichá či simulaci hodu mincí. (*Tato situace nastane snad jen někdy po resetu na Spectru při přepínání ve střední poloze. Po resetu přepínačem se spolehlivě skáče do TR-DOSu a při resetu na Spectru a přepínání v krajní poloze se vždy skočí do Basicu. - pozn. red.*)

Takže malé shrnutí: jako patrně nejefektivnější varianta se jeví použití pouze jednopólového dvoupolohového přepínače (viz [6], odstavec vpravo nahore), přičemž majitelů 128 se týkají stejné důvody ohledně dřívějších verzí (viz výše), pro než je prakticky zbytečné osazovat další přepínač (či spíše vypínač). A vlastníci starých dobrých *gumáků*, kteří snad ještě nemají zabudovaný reset v počítači, mají jedinečnou možnost vybudovat jej touto vymožeností prostřednictvím Betadisku.

Abych však nezakufroval daleko od původního námětu, tedy LECROMky v družném soužití s Betadiskem. Z předchozího textu bylo patrné možno nabýt klamného dojmu, že tato kombinace oplývá spíše nevýhodami než zjevnými výhodami, pokusím se však vyvést Vás z omylu. Budu s dovolením předpokládat, že vymoženosti LECROMky jsou Vám již alespoň částečně známy, ať již z vlastní práce s ní, nebo popřípadě z podrobného popisu v [7]. Omezím se proto pouze na používání některých jejích příkazů v souvislosti s prací s Betadiskem bez širšího vysvětlování funkce vlastních příkazů. A vezměme to pěkně popořádku. Po zapnutí či resetu se nacházíme v úvodním menu. Pokud hodláme rovnou začít pracovat s diskem, máme samozřejmě možnost skočit do Basicu a poté známou sekvencí RANDOMIZE USR 15616, popřípadě s využitím možnosti používat zkracování příkazů ra.u.15616. Každopádně nás to bude od začátku až do konce stát v nejlepším případě alespoň 12 stisků kláves. Je tu však k dispozici daleko jednodušší a rychlejší možnost. Z úvodního menu skočíme do monitoru a z něj potom příkazem g3d00 (což je hexadecimální ekvivalent adresy 15616) rovnou do systému Betadisku. Zde je tedy *přístupová doba* pouhých sedm úhozů. A nyní již vesele můžeme *bušit* příkazy TR-DOSu.

- vše, co bylo řečeno o funkčnosti či nefunkčnosti Spectra se samozřejmě plně týká veškerého dalšího použitého příslušenství, tedy máme-li k tomu možnost, důkladně nejprve otestujeme diskovou mechaniku, nejlépe opět na osvědčeném kolegovi, vlastnicím již naprosto spolehlivé funkční Betadisk nebo se z něj naopak pokusíme vymámit jeho již ověřenou mechaniku i se spojovacími kabely.

- rovněž ne každá právě používaná disketa se také opravdu jako disketa chová, sám mám již také několik takových exemplářů rozvěšených po pokoji jakožto dekoraci.

- některé záhadnější problémy, ponejvíce naprosto nepravdivě se objevující chyby při nahrávání dat z disku, jsou způsobeny nevhodným, většinou slabým zdrojem, ať již pro diskové mechaniky, nebo i pro vlastní počítač. Tedy napětí zdroje musíme měřit nejen naprázdno ale především při zatížení. Nezřídka jsem viděl originální zdroj, kterému po připojení na všelijak ověřené Spectrum jaksi došel dech, nedávno jsem zase naměřil u profesionálního zdroje, kterému výrobce zaručoval tlusté parametry 5V/2.5A a 12V/1.5A po připojení pouhých dvou mechanik místo 12V jen 10.4V a na 5V dokonce 3.4V! (takže důvěřuj ale proměřuj).

- i napětově stabilní zdroj však může mít ještě další nedostatky, vypadávání či napětové špičky a jiné poruchy, ať již za vydatného přispění rušením po síti (v mém případě je například lednička často spolehlivější než reset).

- důvodů nesprávné či dokonce absolutně žádné funkce řadiče je mnohem víc, než bych zde mohl vůbec vyjmenovat, takže na závěr alespoň ještě jednu dobrou radu, zkuste případné závady nehledat pouze ve vlastním hardware řadiče, ale také v jeho blízkém i vzdálenějším okolí (vždyť svět je přece tak malý).

Zdalo by se tedy, že většina problémů je již vyřešena, diskové mechaniky si konečně pod bedlivým a neomylným dohledem Betadisku vesele stepují, od krku se line příjemné teplo, odsouvající hrůzy za okny zuřící sněhové bouře kamsi do jiné galaxie a... A když tu náhle poetická idylka je nečekaně přerušena. Jak již patrně správně tušíte, svých práv se začínají domáhat naši kolegové takyskoro-spectristé, tedy majitelé stále se rozrůstajícího stádečka Didaktů všech barev a nejrůznějších smýšlení (rychlost, s jakou dokázal výrobce pohotově několikrát modifikovat svůj výtvar je vskutku obdivuhodná).

Tyto poněkud zmutované klony Spectra, však mají i přes své drobné masárky dvě naprosto nesporné výhody - jsou v poslední době dostupnější než původní výrobek a především z nich lze velmi nenáročnou přestavbou vyrobit téměř stoprocentně kompatibilní Spectrum a dosáhnout tak počítačového Olympu. A tak na otázku, zda lze připojit Betadisk k Didaktiku, existuje jednoznačně pozitivní odpověď. Obecně lze totiž připojit cokoliv k čemukoliv (horší to pak někdy bývá se sháněním nového zařízení); záleží pouze na míře vynaloženého úsilí ve vztahu k dosaženým výsledkům. Vždyť takové átéčko je vlastně přímo báječnou externí klávesnicí a přídavnou pamětí, popřípadě i náhradním displejem pro tak dokonalý počítač, jakým Spectrum bezesporu je a ještě dlouho bude. Prozatím alespoň několik základních rad:

- Betadisk lze v podstatě bez jakýchkoliv úprav používat i ve spojení s Didaktikem, ovšem samozřejmě po dodání externího napětí 9 a 12 voltů pro desku řadiče.

- totéž tvrzení se bohužel již nedá použít i ve směru opačném, Didaktik se totiž bude muset alespoň minimálně upravit, aby byl schopen spolupráce s Betadiskem. Zároveň vás však mohu všechny uklidnit tím, že případné připojení Betadisku k ještě neupravenému Didaktiku nemůže způsobit zkázu jednoho ani druhého.

- hlavním a základním původcem všech nesnází s připojením Betadisku k Didaktiku je obvod paralelního styku 8255A, osazený přímo v počítači. Při důkladnějším prostudování zapojení desky operačního systému (viz [2], str.11) a také vlastních obrazců plošných spojů (třetí strana obálky [5]) nám určitě nemůže uniknout, že veškeré signály Spectra jsou průchozí deskou systému kromě jediného signálu /IORQ, který je na pokračovacím konektoru řadiče nahrazen signálem /IORQ' (viz také popis v [2], str. 9, druhý odstavec vlevo). Tak řadič v době své funkce zabraňuje, aby mu ostatní periferie lezly do zelí. To, že zde je periferní obvod zapojen de facto před Betadiskem místo za, jak by tomu správně mělo být, je onou výše zmíněnou příčinou problémů. Pomoc je celkem snadná, v podstatě jde o to, aby byla pětapadesátka v době práce s diskem odpojena, ať již automaticky, nebo alespoň ručně. Podrobný popis úprav i s postupem je již připraven do dalšího pokračování. Pro ty nedočkavé alespoň malý námět: Díky tomu, že ve většině verzí Didaktiku je 8255A osazena v soklu, stačí ji prostě vylopnout a vše bude již v pořádku, samozřejmě ovšem za tu cenu, že prozatím nebudeme mít k dispozici služby odstraněného obvodu. Pro vyzkoušení to však stačí.

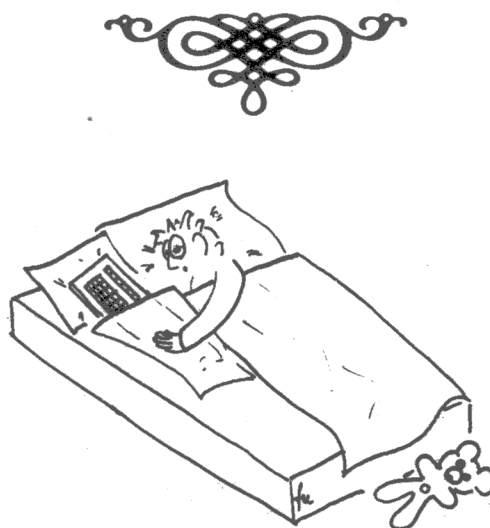
Největším paradoxem na celé věci ovšem je, že ač majitel pravého čistokrevného Spectra (nyní tedy spíše již pouze původem než čím jiným), trápil jsem se s tímto problémem značně dlouhou dobu, neboť jsem si mezi jinými úpravami zabudoval do svého miláčka i onu výše zmíněnou pětapadesátku."

Literatura:

- [1-6] Daniel Meca: Postavte si s námi diskový řadič; Mikrobáze 5-10/89
- [7] Jiří Lamač: LECROM; Mikrobáze 1/89

Potud tedy Herbert Urbanec. Příští pokračování už by se zase převážně mělo věnovat původně zamýšlenému tématu - popisu činnosti integrovaných monolitických řadičů WD 179X a 279X.

(pokračování příště)



ing. Václav Daněček

Pro nové čtenáře nutno vysvětlit, co že to vlastně ten tajemný BOBO 64KB vlastně je. Tedy, je to počítač kompatibilní se ZX Spectrem. Proti Spectru však má navíc ještě další rozšířené zobrazovací módy. Nejzajímavější je asi možnost grafického zobrazení v režimu "hires", který umožňuje zobrazovat 512x256 bodů. Počítač je navíc vybaven standardně vestavěným sériovým a paralelním portem. Podrobný popis a schema je možno nalézt ve zpravodaji Mikrobáze 6+8/89, který byl předchůdcem Mikrobáze *hobby*. Pro případné zájemce může redakce zajistit dodání těchto čísel (cena 3x12,-Kčs + poštovné).

Po nedávném prudkém a nečekaném vzrůstu cen paměti 256Kb se v poslední době jejich cena opět vytváří pohybuje směrem dolů, takže začíná být srovnatelná s cenou paměti 64Kb. Proto přicházejí opět do módy nejrůznější rozšíření paměti počítačů, někdy jdoucí až do absurdity.

Vzhledem ke způsobu využití RAM u počítače BOBO 64 není její rozšíření žádným luxusem. V případě provozu počítače pod systémem CP/M je totiž oblast TPA zmenšena o VideoRAM v rozsahu 7+16KB podle použitého video módu (normal/hires). Proto ještě před zdražením paměti vznikla verze s pamětovými čipy 256Kb. Tím má BOBO k dispozici "čistých" 64KB RAM pro CP/M, 16KB pro VideoRAM a zbývajících 176KB pro RAMdisk. Dnes se tedy tato verze stává opět aktuální, proto následuje její popis.

Při volbě zapojení jsme vycházeli z požadavku maximální kompatibility se ZX Spectrem i s jeho upravenými variantami. Po zapnutí je tedy BOBO plně kompatibilní se standardním ZX Spectrem. Softwarově je pak možno přepnout do konfigurace kompatibilní s verzí 80KB podle Sinsoftu [3,5], nebo s verzí 256KB podle Lecsoftu [2,4,7]. S ohledem na kompatibilitu s verzí Lecsoft musela být změněna adresa portu pro 8251 (SIO) z původní FDh, kterou používá Lecsoft pro registr stránky paměti, na 7Dh. A pro programátory ještě jedno upozornění - porty 7Dh a 7Fh jsou dekodovány jen třemi z osmi možných adresových bitů, takže každý z těchto portů se zrcadlí v možném adresovém prostoru celkem 32x. Přehled portů, jejich adresace a využití jejich jednotlivých bitů je zřejmé z připojené tabulky.

Úprava je provedena podle připojeného schématu. Plošný spoj nebyl měněn a ani nebylo nutno vymýšlet přidavnou destičku, vzhledem k jednoduchosti celé úpravy a možnosti "osedlání" přidávaných IO na stávající. Fyzicky je třeba především zaměnit původní obvody 4164 za 41256 (kdo má paměti v patičkách, ten má vyhráno). Pro ovládání RAMdisku je zapotřebí přidat tři IO (74LS175, 74LS253 a 74LS74). 74LS74 je osazen do volné rezervní pozice, 74LS175 je jako "sedlák" přidán "na záda" obvodu v pozici 404 (registr řídicího slova pro port "BOBO", s kterým má přímo propojené vývody 1, 5, 12, 13, 8 a 16. Při použití EPROM typu 27128 lze zjednodušit dekodování a vypustit z původního zapojení obvod 74LS138 v pozici 405. Na jeho místo se pak osadí (po odškrábnutí přebytečných spojů) obvod 74LS253. Další možnost je tímto

obvodem "osedlat" některý jiný, např. na pozici 502.

Po usazení obvodů je třeba provést zbývajících propojení podle schématu. Telegraficky: propojit vývody 1 u paměti na adresu MA8; odškrábnout spoj 404/6, 404/7 (ROMEN); odškrábnout spoje na hradlo 402/8, 9, 10 a spojit 403/11 a 404/9 (tím se ušetří jedno potřebné hradlo OR); odškrábnout spoj na 309/3 a tuto špičku propojit na zem (malá změna zobrazovacího módu která zabrání přepínání alokace VideoRAM) a udělat všechny zbývajících propojení podle schématu.

Obvod 74LS175 (sedlák 404) v tomto zapojení funguje jako registr pro stránku RAMdisku v režimu 256KB podle Lecsoft [2,4,7], obvod 74LS74 přepíná RAM podle verze 80KB Sinsoft [3,5] a 74LS253 (multiplexer) vytváří přidanou adresu MA8 pro paměti 256KB. Závěrem je nutno upozornit, že vzhledem k jednoduchosti konstrukce nebyla řešena otázka přechodu z jedné verze na druhou softwarově za provozu. Nemělo by to ani smysl. Po resetu počítače je vždy zaručen výchozí stav obvodů.

Literatura:

- [1] V. Daněček, J. Blahůt: BOBO 64K; Mikrobáze 6+8/1989
- [2] J. Lamač: 80K RAM pro ZX Spectrum; Mikrobáze 6/87, str. 41
- [3] P. Troller, P. Císař: Úprava adresování a zvětšení rozsahu paměti počítače ZX Spectrum; ST 11/87, str. 417
- [4] J. Lamač, D. Meca, J. Vaněk: CP/M na ZX Spectrum; ARA 9/88, str. 337
- [5] L. Sieger: ZX Spectrum s rozšířenou pamětí; Mikrobáze 10/88 až 3/89
- [6] rozhovor "Je toho 80 kilo..."; Mikrobáze 10/88 a ohlas na něj; Mikrobáze 4/89
- [7] J. Vaněk: Variace na téma Rozšíření paměti ZX Spectra; Mikrobáze 9/89, str. 6

Pozn. red.: Tento příspěvek je vlastně splnění slibu ze zpravodaje Mikrobáze. Tehdy jsem ještě netušil, že Mikrobáze zanikne, aby později povstala z popela a že tedy řadu slíbených příspěvků budu uveřejňovat až ve vlastním časopise. Protože se sliby mají plnit, vyhledal jsem ing. Václava Daněčka a dohodl s ním spolupráci pro nový časopis *Mikrobáze hobby*. Takže se společně můžeme těšit na řadu zajímavých příspěvků. Možná se dočkáme i toho již avizovaného rozšíření ULA. A ještě něco - těm, kteří se pustí do vlastní stavby BOBO (ať už verze 64KB, nebo 256KB), může redakce zprostředkovat nákup omezeného množství plošných spojů a některých součástek. Prostřednictvím redakce se můžete spojit také s ing. Daněčkem, který vám zodpoví nejrůznější dotazy nejen na problémy kolem počítače BOBO. Zájemci z Prahy se mohou případně na ing. Daněčka obrátit i osobně v jeho provozovně Mikro-počítače v Praze 3, Želivského 27 a to vždy v úterý a ve čtvrtek od 16 do 18 hod. Tam se také můžete s důvěrou obrátit, když váš počítač (nezáleží na značce) přestane fungovat. Ing. Daněček totiž dokáže opravit (téměř) všechno. D.M.

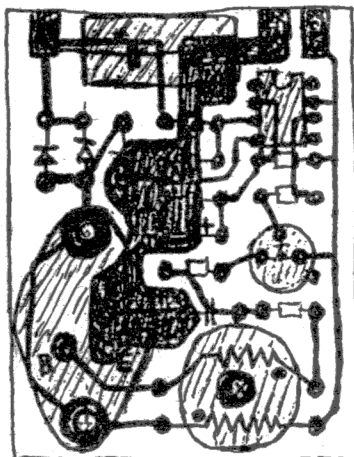
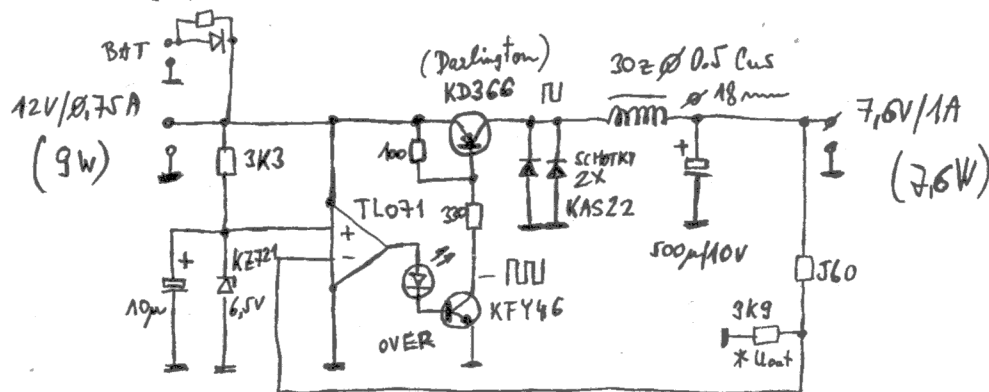
Přímá z bloku

Napájecí pro ZX-SPECTRUM + BETA v.5.03

ověřeno 30.9.90
tomb

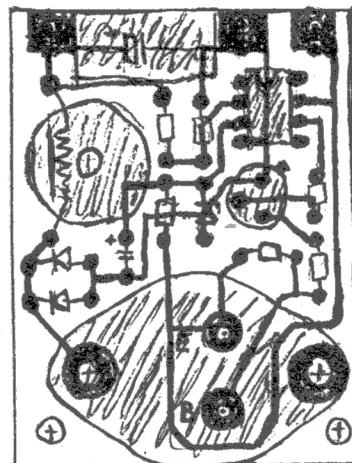
Napájecí cca 10 mA

účinnost $\frac{7.6}{9} \cdot 100 = 84\%$



PŘÍMA Z BLOKU - tak se jmenuje naše nová rubrika, která nebude obsahovat stavební návody, ale nápady a náměty z bloku našich čtenářů. Ty by měly být inspirací pro ostatní. Je proto klidné možné, že se s některými z námětů za čas setkáte znovu, tentokrát už ve formě rozpracovaného a řádně vyzkoušeného stavebního návodu. Důležitý však je prvotní nápad. A tak vás touto cestou vyzýváme - dejte k dispozici ostatním vaše nápady. A naopak - zkuste některé náměty z této rubriky dále rozpracovat a pak nám je poskytnout.

Dnešní námět pochází z dílny Pavla Perutze z Prahy. Spínané zdroje mají větší účinnost a netopí. Tranzistory mohou být bez chladiče. Vzestupný stabilizátor je určen k napájení mechaniky disku z jediného napětí.



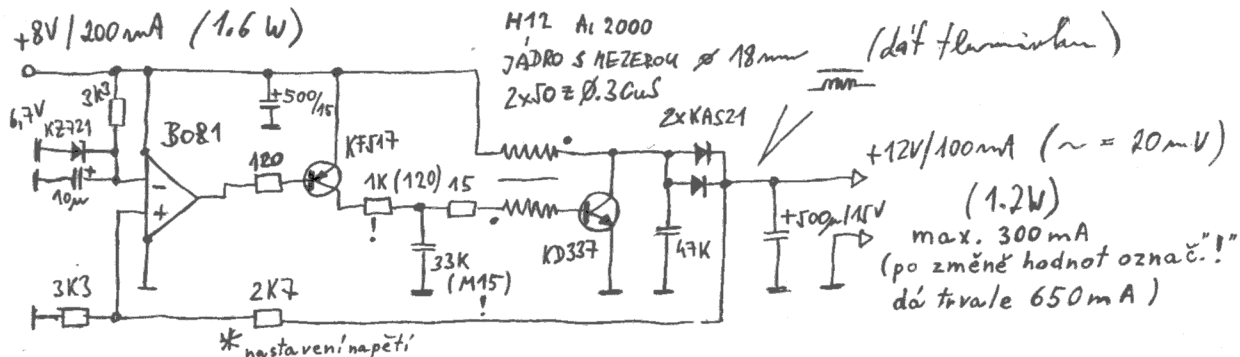
VZESTUPNÝ STABILIZÁTOR +8 → +12V/200mA

ověřeno 3.10.1990
tomb

účinnost $\frac{1.2}{1.6} \cdot 100 = 75\%$

napájecí 8mA

+8V/200mA (1.6W)



* nastavení napětí

Recenze

Daniel Meca

Na podzimním PC Salonu v Kotvě bylo k vidění všelicos. Já se však zvlášť zajímal o programy, které budou vhodné i pro začátečníky, mají širší uplatnění a jsou cenově přístupné. Zvlášť mě zajímal nějaký (nejlépe rezidentní) anglický slovník na *písičko*. Vím že výběr je velký, ale já šel hlavně po dostupné ceně. V tomto ohledu příjemně překvapil slovník dodávaný firmou sympatického názvu **Computer Sapiens**. Jeho cena (259,-Kčs) je rozhodně přístupná a pokud vím, také ze všech nejmenší. I majitel firmy se ukázal být sympatický, a tak jsem získal vzorek na otestování pro **Mikrobázi hobby**. Protože jsou k dispozici slovníky anglické i německé, zapůjčil jsem si pro jistotu oba.

A jak dopadl test? Tak to se dozvíte příště. Ale abych tolik nenapínal - přes malé nedostatky v návodu a v menu na obrazovce je to nesporně užitečný pomocník. Slovníky jsou rozsáhlé, což však poněkud komplikuje život těm, kteří nemají Winchester. Při provozu z floppydisku se musí slovníky rozdělit. Na to však autor (LUSA software) pamatoval.

Program je dodáván pod poněkud exotickým názvem **YAP**. Je to vlastně balík programů, které ošetří češtinu z klávesnice (možno konfigurovat), nahraží češtinu do EGA, nebo VGA a umožňují také správu slovníků, jejich doplňování, spojování a třídění.

Balík obsahuje tyto programy:

ANGL	YAP	405256	LINK	EXE	18954
CONFIG	EXE	18382	NEM	YAP	383271
CSEGA	EXE	6617	README	EXE	8484
CSVGA	EXE	7129	YAP	CFG	41
ERASED	EXE	18868	YAP	EXE	27248
FILTER	EXE	12810	YAP	TMP	1174

Přesněji řečeno, dostanete vždy buď **ANGL.YAP**, nebo **NEM.YAP** a zbytek balíku se opakuje.

Rezidentní slovník YAP (a řadu dalších zajímavých programů) dodává firma **Computer Sapiens**, pošt. příhr. 105, 160 41 Praha 6

Další zajímavý program, který jsem objevil na PC Salonu nabízel další sympatický podnikatel, tentokrát přímo autor programu. Program se jmenuje **Forte** a jeho autor Jaroslav Kočandrle. Forte je program, po kterém jistě rád sáhne každý, kdo je zvyklý pracovat s obvykle kořistním Norton Commanderem. Jedná se totiž o původní český program, který nahrazuje Norton Commander v3.0.

A nejen nahrazuje - má dokonce něco navíc. Komunikuje s uživatelem v českém jazyce, umí inverzní výběr souboru, dokáže naformátovat disketu a změnit její volume. Umožňuje tisk souborů na pozadí s možností odfiltrování diakritiky a přesměrování výstupu. Vestavěný editor odpovídá ovládním i funkcí oblíbeným Turbo editorům. A ještě jedna zajímavost - jeho rezidentní část má jen 1099 bajtů. Samozřejmě, že jsem na něm našel zase nějaké mouchy, ale autor již pracuje na jejich vychytání. Podrobnosti se dozvíte příště v testu. Poslední zajímavostí tohoto programu je jeho cena: 997,-Kčs za 1. instalaci a 299,-Kčs za každou další. Sympatické je i to, že pro zdravotnictví, školství a soukromé podnikatele je až 50% slevy.

Adresa autora je: **fa Jaroslav Kočandrle, Komenského alej 1532, 438 01 Žatec**

V našich testech a recenzích programů se budeme zaměřovat převážně na původní české a slovenské programy, hlavně na tzv. *utility*, které zpřijemňují práci s počítačem. Proto vyzýváme autory takovýchto programů, aby sami nabídli svá díla k recenzi. Uveřejnění výsledků testu, či recenze je (u dobrých produktů) ta nejlepší reklama. A vlastně i nejlacinější. Plošná inzerce v podobném rozsahu by stála několik tisíc Kčs. Neváhejte - my máme zájem své čtenáře informovat.

Uděláme si katalog

Už se vám někdy stalo, že v nějakém přístroji byl poměrně obyčejný číslicový IO, ale vy jste nemohli nikde nalézt jeho zapojení? Třeba takový 7426 může člověka docela potrápit. U nás je to zatím s katalogy dost špatné. Tesla do svých katalogů sice občas zařadila IO, které nevyráběla, ale většinou tam spíš nebyly ani ty, které měli běžné v obchodech. Knižně nevyšlo téměř nic, už jen proto, že výrobní lhůta kolem pěti let nedává mnoho nadějí na aktuálnost. Tak vlastně vše dosud zachraňovalo jen AR. Jenže ani tam nebylo všechno a v poslední době vlastně nic. Záměrně se nzmínuji o zahraničních katalogích, kde je sice (skoro) všechno, ale z cenových důvodů jsou naprosto mimo soutěž.

Zde tedy přicházím se svojí troškou do mlýna. Myslím, že mnoha čtenářům by mohl takový katalog číslicových IO docela dobře posloužit. Tím se také předplatné každému značně zhodnotí (cena obdobného katalogu v SRN je od 25DM nahoru).

Jenže - jak získat podklady pro co nejkompletnější katalog číslicových TTL a CMOS? Za léta své

amatérské i profesionální činnosti se mi povedlo dát dohromady docela slušný základ. Trochu jsem ho doplnil s pomocí redakčních spolupracovníků a s ostatním se obracím na vás. Představa je taková: V každém čísle **Mikrobáze hobby** bude vždy nejméně na dvou stránkách uveřejněna část katalogu. Když u některého IO nebudu znát zapojení a přitom se budu domnívat, že by mohl existovat, nechám prostě příslušný obdélníček volný. Když někdo ze čtenářů (a není to málo lidí) bude zapojení znát, pošle ho redakci. Tyto údaje od čtenářů uveřejníme nakonec samostatně. Stačí si pak volná místa přelepit novými údaji a máte nejkompletnější katalog. Co vy na to?

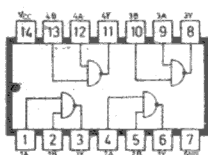
A ještě něco. Pokud někdo má k dispozici nějaké dobré katalogy, ozvěte se raději hned. Já vám zašlu obratem seznam IO, jejichž zapojení zatím nemám k dispozici. Tak můžeme mnoho obdélníků zaplnit rovnou. Také bych uvítal seznam neexistujících čísel IO, abych pro ně zbytečně nenechával místo. Děkuji předem za spolupráci.

Daniel Meca 15

Číslicové obvody TTL

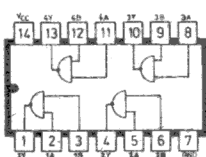
© Mikrobáze hobby

SN 7400



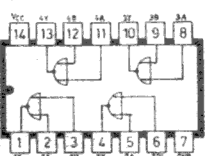
Čtyři dvouvstupová hradla NAND

SN 7401



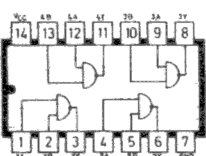
Čtyři dvouvstupová hradla NAND (o.K.)

SN 7402



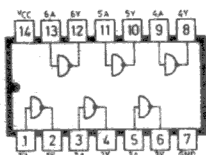
Čtyři dvouvstupová hradla NOR

SN 7403



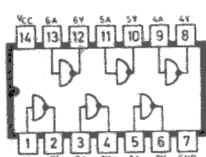
Čtyři dvouvstupová hradla NAND (o.K.)

SN 7404



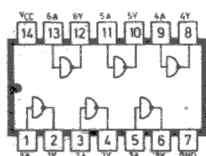
Šest invertorů

SN 7405



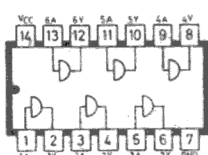
Šest invertorů (o.K.)

SN 7406



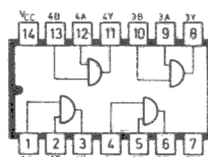
Šest invertujících budičů (o.K. 30V)

SN 7407



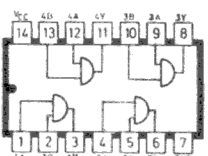
Šest budičů (o.K. 30V)

SN 7408



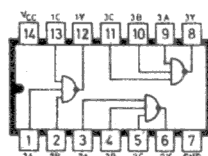
Čtyři dvouvstupová hradla AND

SN 7409



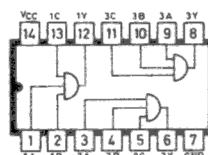
Čtyři dvouvstupová hradla AND (o.K.)

SN 7410



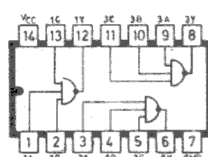
Tři trojvstupová hradla NAND

SN 74 LS 11



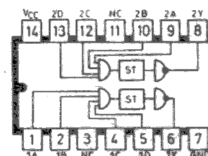
Tři trojvstupová hradla AND

SN 7412



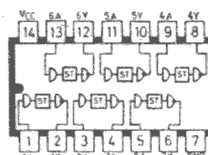
Tři trojvstupová hradla NAND (o.K.)

SN 7413



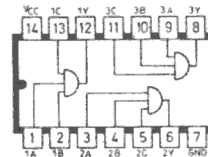
Dvě čtyřvstupová hradla NAND se Schmitt. obvodem

SN 7414



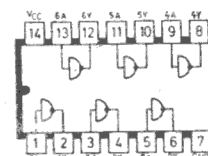
Šest invertujících Schmittových obvodů

SN 74 LS 15



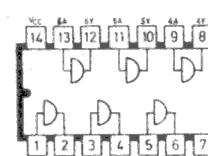
Tři trojvstupová hradla AND (o.K.)

SN 7416



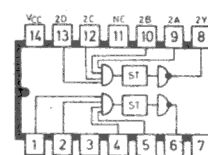
Šest invertujících budičů (o.K.)

SN 7417



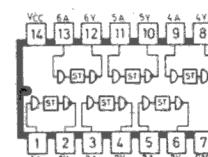
Šest budičů (o.K. 15V)

SN 7418



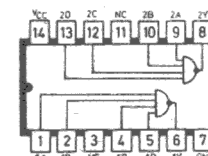
Dvě čtyřvstupová hradla NAND se Schmitt. obvodem

SN 7419



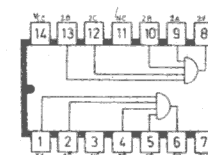
Šest invertujících Schmittových obvodů

SN 7420



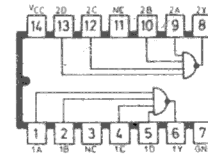
Dvě čtyřvstupová hradla NAND

SN 74 LS 21



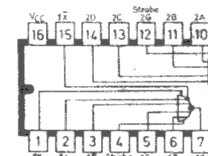
Dvě čtyřvstupová hradla AND

SN 7422



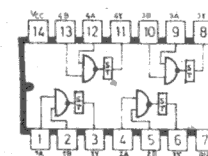
Dvě čtyřvstupová hradla NAND

SN 7423



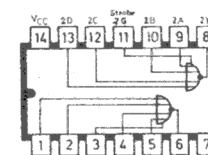
Dvě čtyřvstupová hradla NOR

SN 7424



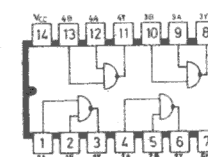
Čtyři dvouvstupová hradla NAND se Schmitt. obvodem

SN 7425



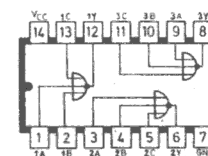
Dvě čtyřvstupová hradla NOR

SN 7426



Čtyři dvouvstupová hradla NAND

SN 7427

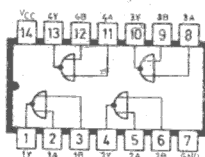


Tři trojvstupová hradla NOR

Číslicové obvody TTL

© Mikrobáze hobby

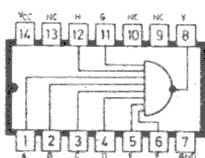
SN 7428



Čtyři dvouvstupová
výkonová hradla NOR

???

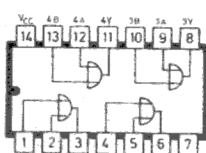
SN 7430



Osmivstupové hradlo NAND

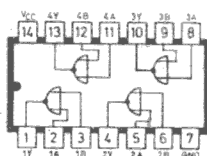
???

SN 7432



Čtyři dvouvstupová hradla
OR

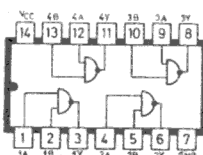
SN 7433



Čtyři dvouvstupová
výkonová hradla NOR

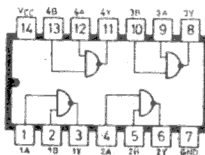
???

SN 7437



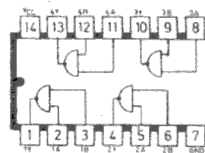
Čtyři dvouvstupová
výkonová hradla NAND

SN 7438



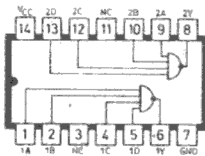
Čtyři dvouvstupová
výkonová hradla NAND

SN 7439



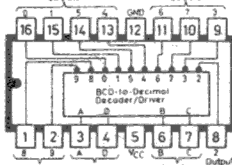
Čtyři dvouvstupové budiče
NAND (o.K.)

SN 7440



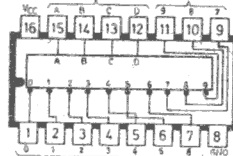
Dvě čtyřvstupová
výkonová hradla NAND

SN 7441



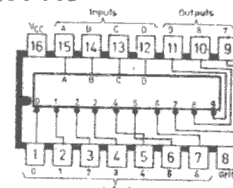
BCD -> Dec. dekodér
(o.K. 70V)

SN 7442



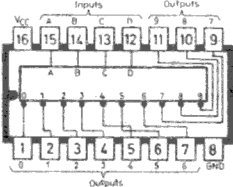
BCD -> Dec. dekodér

SN 7443



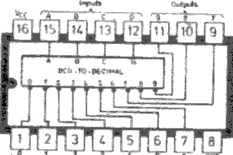
Excess-3 -> Dec. dekodér

SN 7444



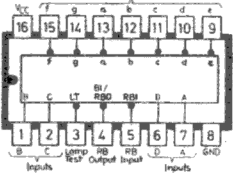
Excess-3-Gray -> Dec.

SN 7445



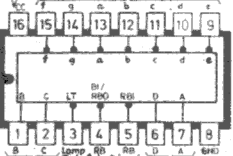
BCD -> Dec. dekodér
/ budič

SN 7446



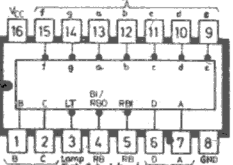
BCD -> 7-segment dekodér
/ budič

SN 7447



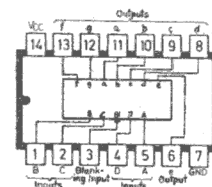
BCD -> 7-segment dekodér
/ budič

SN 7448



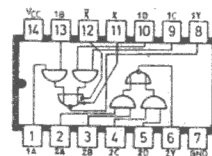
BCD -> 7-segment dekodér
/ budič

SN 74 LS 49



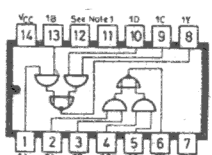
BCD -> 7-segment dekodér/b

SN 7450



Dvě hradla AND-OR-INVERT

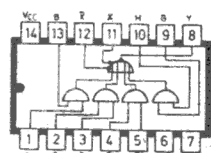
SN 7451



Dvě hradla AND-OR-INVERT

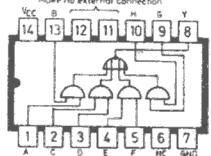
???

SN 7453



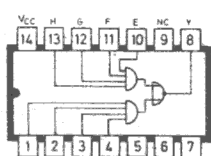
Hradlo AND-OR-INVERT
s expandérem

SN 7454



Hradlo AND-OR-INVERT

SN 74 LS 55



Hradlo AND-OR-INVERT
s 2x4 vstupy

Opět programátor EPROM!

Ing. Jan Hrejša

Úvodem

Článek o programování a programátorech paměti již bylo napsáno mnoho - lepších i horších - tak jako zapojení v nich publikovaná. Přesto bych se chtěl pokusit o další popis programátoru z hlediska některých vlastností, které byly mnohdy zjištěny až dlouhodobým používáním jednotlivých typů programátorů, jichž jsme vytvořili zatím pět.

Nejprve by bylo dobré se alespoň stručně zmínit o typech programovatelných obvodů a o specifikách jejich programování. Programovatelné obvody lze rozdělit v principu na paměti a logická pole. Logická pole jsou v současné době velmi moderním prvkem elektroniky, umožňujícím nahradit složitou hradlovou strukturu jedním nebo několika málo obvody. Jejich struktura je programována maskou, a zákazník tedy nemá možnost změnit jejich strukturu, nebo jsou jednou programovatelná (PAL, PLS) případně reprogramovatelná (GAL) tak jako paměti EPROM. Programování těchto polí je poměrně složité - mimo jiné i díky jejich značnému počtu a různým programovacím metodám a je spojeno i s problémem návrhu vnitřní struktury, takže je námětem pro samostatný článek.

Programovatelné paměti lze rozdělit do několika hlavních typů. Nejzákladnějším typem jsou paměti ROM, programovatelné maskou která se tvoří při výrobě poslední vrstvy integrovaného obvodu. Tyto paměti může programovat pouze výrobce při výrobě čipu a tato metoda je obvykle použitelná od 100 kusů výše. Výhodou paměti ROM je vyšší spolehlivost (nedochází ke zpětné migraci kovu z přepálených spojek případně k vybití elementárních kondenzátorů).

Dalším, pro malospotřebitele zajímavějším, typem programovatelných pamětí jsou paměti typu PROM. Při jejich programování dochází k přepalování Ni-Cr spojek na čipu, čímž je do obvodu zaznamenávána informace. Přepálené spojky však již nelze spojit, čímž je výrazně omezena možnost opravy chyb v naprogramovaných pamětech. Dalším nebezpečím je zpětná migrace kovu z přepálených spojek, nebylo-li programování provedeno vhodným impulsem. Této chyby se dopuští řada dříve uveřejněných programátorů. Zpětná migrace se projeví tím, že propálená buňka se po určité době (někdy i po několika měsících) vrátí na původní hodnotu nebo do neurčitěho stavu.

Třetím typem programovatelných pamětí jsou paměti reprogramovatelné, to znamená že jejich obsah lze v omezeném počtu cyklů měnit. Výrobce obvykle garantuje u EPROM 10 - 100 a u EEPROM až několik tisíc cyklů. Podle způsobu vymazání paměti rozlišujeme dva typy - EPROM mazatelné ultrafialovým zářením a EEPROM mazatelné elektricky.

Princip paměti EPROM spočívá ve vytvoření unipolárních tranzistorů, s hradlem izolovaným oxidem křemičitým, které tvoří elementární kondenzátor. Tento kondenzátor ve vybitém stavu prezentuje základní stav paměti EPROM. Jeho nabitím (pomocí ladinování náboje přes izolační bariéru při zvýšeném napětí) dochází k zaznamenání hodnoty do paměti

(obvykle to bývá log. 0). Mazání EPROM se děje ultrafialovým zářením s určitou doporučenou vlnovou délkou. Tím se stane izolace, která obklopuje hradlo tranzistoru vodivou (fotoelektrický efekt) a elementární kondenzátor se vybije. Značně rozšířeným nešvarem je používání klasických osvětlovacích výbojek nebo "horského slunce" pro vymazávání EPROM. Jediným správným postupem je však použití speciálních mazacích výbojek se studeným světlem, neboť zahřívání paměti klasickou výbojkou při mazání výrazně zkracuje životnost paměti. Také před mazáním paměti v zapnutém stavu je nutno důrazně varovat. Některé dřívější návody (např. v AR) doporučovaly během mazání paměť čist a až bude prázdná s vymazáváním přestat. Mělo to zkrátit dobu expozice EPROM a zvýšit její životnost. Většina pamětí se dá tímto způsobem pouze spolehlivě zničit, neboť fotoelektrický efekt nepostihuje pouze vybité kondenzátory, ale projevuje se i na ostatních tranzistorech např. ve vnitřním dekodéru adres, které se tak dostávají do zkratu. To bylo experimentálně ověřeno a nevěřící Tomášové si to mohou klidně zkusit. Po naprogramování je zvykem u EPROM přelepit okénko neprůhledným materiálem, neboť paměť se vymazává nejen působením záření mazací výbojky, ale i např. slunečním zářením a jinými zdroji UV záření - i když samozřejmě podstatně pomaleji (vymazání pomocí výbojky obvykle trvá 15 - 45 minut, slunečním světlem několik týdnů).

Paměti EEPROM se vymazávají elektrickým impulsem, a to buď celé najednou, nebo jen jedna definovaná adresa (vlastně se přepíše novými daty). Posledním hitem mezi EEPROM jsou tzv. pomalé RAM. Jedná se o EEPROM, která pro svoji činnost nepotřebuje zvýšené programovací napětí, stačí pouze zapisovací impuls, trvající jednotky ms.

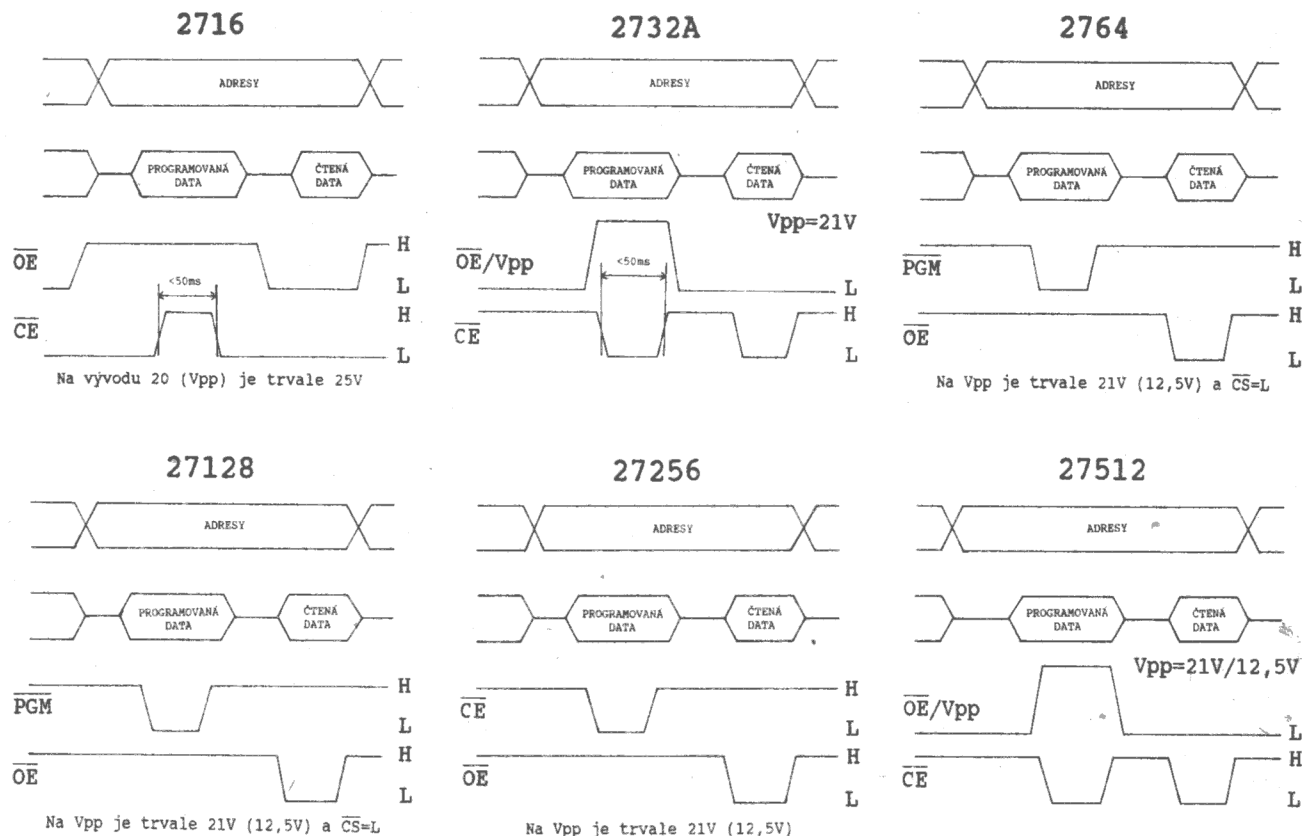
Stejně jako klasické EPROM se chovají EPROM jednočipových mikropočítačů, rozdíl je pouze ve složitějším programování, které je způsobeno vazbou paměti na vnitřní mikroprocesor.

EPROM lze z hlediska jejich historie rozdělit do tří základních kategorií. Nejstarším typem byly paměti se třemi napájecími napětími (pokud pomineme úplně původní paměti vyráběné technologií PMOS, které se však nikdy příliš nerozšířily). Vyráběly se v organizaci 256x8 případně 1kx8 a výjimečně i 2kx8. Novější byly paměti s jedním napájecím napětím bez možnosti automatické identifikace. Typická kapacita 2kx8 a 4kx8. V současné době se vyrábějí skoro výhradně paměti, umožňující automatickou identifikaci s kapacitami 8kx8 - 64kx8. Automatická identifikace spočívá v tom, že přivedením napětí 11V na adresu A9 se na datové sběrnici objeví 2 byte, které určují typ paměti, výrobce a programovací napětí. Paměti obvykle existují ve dvojím provedení - s vyšším a nižším programovacím napětím. Toto napětí nesmíme při programování splést, zejména směrem nahoru, neboť by došlo ke zničení paměti.

Vlastní programování se může provádět podle různých algoritmů. Základní algoritmus doporučuje programování pomocí programovacího impulsu kons-

tantní délky. Je to nejčastější způsob, používaný většinou u nás publikovaných programátorů. Ve světě se k němu kloní zejména japonští výrobci pamětí. Jeho výhodou je jednodušší hardware programátoru i jednodušší softwarové vybavení, nedostatkem je delší programovací doba a zejména obtížnější mazání naprogramovaných pamětí. Druhou možností je použití rychlejších algoritmů, např. systému fy Intel označovaného jako **Intelligent programming**. Tato metoda spočívá v přivedení krátkého programovacího impulsu po jehož skončení se čte obsah dané buňky. Pokud obsah není správně zaznamenán, impuls

se prodlouží a postup se opakuje. Pokud jsou data správná, opakuje se záznam impulsem trojnásobné délky, než která byla dosud použita a programování daného byte se ukončí. Programování se provádí při zvýšeném napájecím napětí. Tato metoda, zejména u kvalitních pamětí, výrazně snižuje dobu programování a šetří paměti. Není vhodná pro paměti horší kvality (zejména sovětské), které pak snadno ztrácejí obsah. Programování jednotlivých typů pamětí se od sebe liší, jak je zřejmo z následujících programovacích diagramů:



Konstrukce programátoru

Na základě uvedených poznatků byly zkonstruovány dva programátory. První z nich je jednoduchá verze, určená k přímému připojení k počítači třídy PC (ale lze ji připojit prakticky k libovolnému mikropočítači) a je určena pro programování pouze EPROM od 2716 do 27512 s oběma programovacími napětími v provedení klasickém i CMOS. Programátor neumožňuje automatickou identifikaci ani programování jednočipových mikropočítačů.

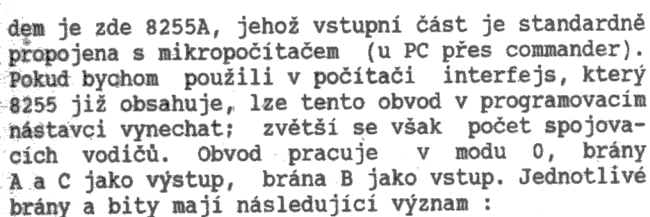
Druhý je profesionální varianta s vlastním mikroprocesorem, která umí programovat PROM, EPROM, EEPROM, jednočipové mikropočítače a používá i systém automatické identifikace. K počítači je připojena přes rozhraní RS232C, ale může pracovat i samostatně (má vlastní display a klávesnici a lze k ní připojit i jednotku pružných disků).

Popíšeme si zde první, jednodušší variantu - jednoduchý programátor. Ten sestává ze dvou samostatných částí - interfejsové jednotky (dále označované commander) a z vlastního programovacího zařízení. Tato koncepce byla zvolena především z hlediska univerzality. Programovací jednotka zůstává stále stejná, pouze commander se mění v závislosti na typu počítače. Protože commander je v podstatě jenom zesilovač sběrnice s dekodérem adres, dá se po drobných úpravách jeho zapojení použít i pro jiné typy počítačů. Kromě toho je commander velmi příjemnou pomůckou při vývojových pracích na počítači, neboť funguje jako oddělovač

sběrnice, takže můžeme experimentovat daleko klidněji. Jeho zapojení je na obr. 1.

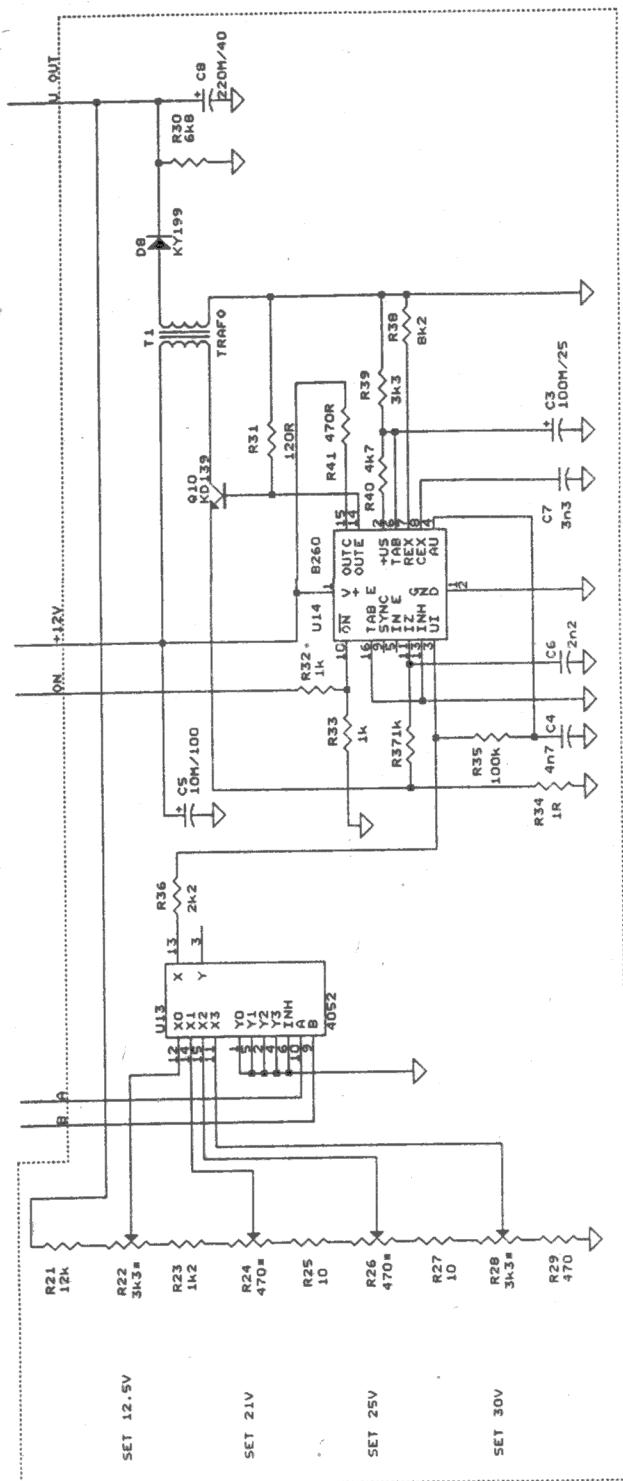
Základem je oddělovač datové, části adresové a řídicí sběrnice, realizovaný dvěma obvody 8286. Další částí je dekodér adresy, který je tvořen dvojicí komparátorů 74LS85, které porovnávají aktuální adresu s adresou nastavenou pomocí přepínačů DIL. Toto řešení umožňuje snadno nastavit adresy, na kterých bude commander fungovat. Z komparátoru je ovládán dekodér 74LS138, který poskytuje 4 signály CS pro vnější zařízení. Commander zabírá v adresovém prostoru 16 adres. Obvody jsou doplněny o generátor WAIT se 74123, který je pomocí DIL spínače vypínatelný a slouží ke generaci signálu nepřipravenosti při spolupráci s pomalejšími obvody (pro 8255 CS provenience není nutný, ale je vhodný; pro cizí 8255, pokud nejsou označeny jako "rychlé", obvykle nutný bývá). Dalším prodloužením tohoto signálu je ovládána svítivá dioda, indikující, že počítač naadresoval desku commanderu (LED při každém přístupu blikne). Commander je opatřen 25 pinovým konektorem CANNON pro spojení s vnějšími obvody. Na tomto konektoru jsou vyvedeny takové signály, které umožňují zejména snadné připojení periferních obvodů z rodiny 8080.

Druhou částí programátoru je vlastní programovací zařízení (obr. 2), které je umístěno mimo počítač na propojovacím kabelu, takže umožňuje snadný přístup k programované paměti. Základním obvo-



- Na výstupu 8255 jsou použity zesilovače a střadače. V mnoha zapojeních vidíme programátory, kde paměť je spojena přímo s 8255. V těchto případech není paměť buzena z dostatečně malých impedancí, což vede často ke špatnému programování. Proto je vhodné výstupní signály zesílit. Kromě toho v době, kdy se paměť zakládá do programovací objímky jsou tyto oddělovače ve třetím stavu, takže součástka není zakládána do programátoru v aktivním stavu, kdy hrozí její poškození. Oddělovače čas-
tečně tvoří i záchytné registry, abychom vystačili s počtem signálů poskytovaných 8255. Zapojení s takto rozděleným vstupem a výstupem je nutné pro programování v "Intelligent programming". Na výstup oddělovačů je připojena jediná objímka s nulovou zasouvací silou, s počtem vývodů 28, do které lze vložit všechny uvedené typy EPROM. K tomuto řešení nás vedly zejména ekonomické úvahy a poměrně špat-

Pro přepínání typů paměti bylo u tohoto programátoru využito miniaturních relátek typu RM2, která se u nás běžně vyrábějí. I když toto řešení vypadá zastarale, odpovídá řešení některých programátorů světových výrobců, kromě toho je toto řešení nejekonomičtější, neboť elektronická náhrada, která je částečně použita ve velkém profesionálním programátoru - pomocí J-FET - je poměrně drahá. Kromě toho uvedená relé odpovídají půdorysné pouzdu DIL16, pouze jsou mírně vyšší. Přepínací logika umožňuje přepínání typů paměti a rozvedení programovacích signálů na náležité piny. Pro ovládání relé a dalších vnitřních funkcí programátoru je použit stavový registr, tvořený dvojicí obvodů 74LS175. Pro generování programovacích impulsů a pro přepínání Vpp je použit tranzistorův, ovládaných hradly s vysokonapětovým výstupem (7407). Pro napájení programované paměti potřebujeme jednak +5V a při "Intelligent programming" +6V. Toto napětí nelze získat přímo z napájecího zdroje počítače, proto bylo získáno pomocí tříbodového stabilizátoru z 12V. Výhodou tohoto zapojení je kromě možnosti získání 6V posunutím země stabilizátoru i proudová pojistka pro případ zkratu na paměti. Zdroj je doplněn tranzistorovým spínačem pro vypnutí napájecího napětí při zakládání paměti. Poslední částí programátoru je zdroj programovacího napětí. Tento zdroj musí poskytovat stabilizované napětí 12.5V, 21V, 25V, a přibližně 30V. Poslední napětí slouží k programování v režimu "Brutal", kdy se pokoušíme naprogramovat paměť i za cenu



překročení mezních parametrů (toto řešení vychází z filozofie, že pokud se paměť nepovede naprogramovat, tak je stejně určena k zahození).

Programovací zdroj je řešen jako měnič s obvodem B260, jehož zpětná vazba je přepínána pomocí 4052 na dělič, kterým se nastavuje velikost jednotlivých napětí. Pro napájení měniče je opět použito 12V.

Při pečlivé práci nedělá uvedení do chodu menší problémy (ověřeno na 200-kusové sérii). Před osazením je vhodné pečlivě zkontrolovat plošné spoje, případně proměřit použité součástky. Po osazení informativně zkontrolujeme odběr celého zařízení (5V max 1.2A, 12V max 0.4A). Pokud je vše v pořádku, nastavíme adresu a zasuneme commander do počítače. Při programovém zápisu nebo čtení musí na desce bliknout světlová dioda. Pokud je vše v pořádku, nastavíme programové 8255 a zkusíme ovládat jednotlivé funkce programátoru (řízení

směru dat, spínání relé, ovládání zdrojů, data, adresy atd.) Je-li opět všechno v pořádku, zapneme Vcc do režimu 6V a trimrem u 7805 toto napětí nastavíme. Poté nastavíme i jednotlivá napětí programovacího impulsu (nesmíme zapomenout vždy nastavit vhodné přepínání zdroje). Pokud by některé z napětí nešlo nastavit (což se stává při mimotolerantní citlivosti vstupu B260), je nutno poznamenat odpory označené * a nastavení opakovat. Pak již lze nejlépe na paměti se známým obsahem zkontrolovat čtení a posléze odzkoušet i zápis. Pro snadné uvedení do provozu byl napsán program pro IBM PC, který umožňuje všechny popsané kroky provést rychle a efektivně.

Programové vybavení

Programové vybavení bylo vytvořeno pro počítače třídy IBM PC a je prakticky (z hlediska uživatele) shodné pro velký i malý programátor. Napsáno je převážně v jazyce C, pouze části s vysokými nároky na rychlost jsou napsány v assembleru. Podrobný popis programu bude zřejmě zajímavý hlavně pro uživatele PC, ostatním uživatelům však dá představu o tom, co by podle našeho názoru měl programátor umět. Verzi programu pro IBM PC na disketě je možno získat od autorů prostřednictvím redakce *Mikrobáze hobby*. Pro ostatní počítače bude nutno programové vybavení teprve napsat a následující popis použít jen jako inspiraci.

Počáteční menu programu vypadá následovně:

Programming from:	Programming voltage:
Memory type:	Number of memory chips:
Programming mode:	F5 Set checksum:
Data Bus size:	F6 Production number:
Data cross: Address cross:	Load Save
Command: Memory Clear test	Edit commentary
Write to memory	F2 Write configuration
Read memory to editor	F3 Read configuration Quit
Quit Verify memory	

EDITAČNÍ ČÁST

Toto menu je rozděleno na tři horizontální části. V nejvyšší části jsou příkazy zadávající parametry programování, střední část obsahuje výkonné příkazy a část nejnižší je určena k vytváření dokumentačního souboru k programovaným datům. Výběr činnosti se provádí stiskem klávesy odpovídající prosvětlenému znaku zvoleného příkazu.

První třetina úvodního menu umožňuje zadat parametry a způsob programování. Obsahuje následující příkazy:

PROGRAMMING FROM

Klávesou P se přepíná odkud se programuje. Režim programování z editoru je funkčně omezený a umožní programovat pouze do jedné paměti. Nedovoluje používat kontrolní součty a výrobní čísla. Obsah datového editoru si musí uživatel před použitím dalších příkazů (např. Write) zajistit mimo programovací okno v samotném editoru. Jakákoliv operace v režimu programování ze souboru obsah editoru bez náhrady zničí.

Režim programování ze souboru umožňuje použít všechny možnosti nabízené programátorem. Soubor ze kterého se budou brát data pro programování zvolíme příkazem [Load].

MEMORY TYPE

Volba typu paměti. Volba je totožná s výběrem v programovacích funkcích. Má význam především pro

velikost editoru a hodnotu propalovacího napětí. To je určeno koncovým písmenem uvedeným za typem paměti. L udává nižší a H vyšší propalovací napětí. Jeho hodnota je po navolení vypsána v položce menu [PROGRAMMING VOLTAGE:].

PROGRAMMING MODE

Volba režimu programování. Je možné zadat 3 režimy programování:

1. Režim **NORMAL** používá pevné programovací časy předepsané výrobcem.
2. Režim **INTELLIGENT** použije kratší programovací časy a tedy programuje rychleji.
3. Režim **BRUTAL** je poslední možnost jak naprogramovat nekvalitní paměť. Postupně zvyšuje čas zápisu do paměti.

DATA BUS SIZE

Volba šířky datové sběrnice. Kolik paměti bude vedle sebe. Je-li zvoleno 16, 32 nebo 64 bitů, je vstupní soubor do jednotlivých pamětí patřičně upraven (první byte souboru je D0 - D7, druhý byte souboru je D8 - D15 ...)

DATA CROSS

Volba zakřížení dat v paměti. Logickým datovým bitům Dn (tak jak jsou ve vstupním souboru) přiřazujeme fyzické WIREn bity paměti.

ADRESS CROSS

Volba zakřížení adres v paměti. Logickým adresovým bitům An (tak jak jsou ve vstupním souboru) přiřazujeme fyzické adresové bity WIREn paměti.

PROGRAMMING VOLTAGE

Programovací napětí. Je závislé na zvoleném typu paměti a má pouze informativní charakter.

NUMBER OF MEMORY CHIPS

Počet paměťových čipů potřebných k propálení datového souboru.

SET CHECKSUM

Volba kontrolních součtů v pamětech. Zobrazená informace ON, OFF určuje zda jsou kontrolní součty vytvářeny.

Pohybem šipkami nahoru a dolů nastavíme položku kterou chceme změnit a klávesou ENTER provedeme změnu parametru. Máme možnost výběru z následujících typů generátorů kontrolních součtů:

Check type	Suma-Xor
Check type	Positive-Negative
Check type	Byte-Word-Dword
Check type	Low,High-High,Low
Check type	All-One
Check type	Suma-Xor

CHECK TYPE - SUM

Kontrolní součet se vypočítává jako součet jed-

notlivých položek (byte, word, dword) a ukládá se dvojkový doplněk součtu (negace součtu +1).

CHECK TYPE - XOR

Kontrolní součet se vypočítává jako bitová non-ekvivalence (součet modulo 2, výhradní součet) jednotlivých položek (byte, word, dword) a ukládá se také v této šířce.

CHECK TYPE POSITIVE - NEGATIVE

Určuje jestli se výsledný vypočítaný součet uloží tak, jak byl vypočten (positive), nebo je před uložením bitově znegován (negative).

CHECK TYPE BYT - WORD - DWORD

Určuje z jakých položek je kontrolní součet vypočítáván a také v jaké šířce je ukládán.

Byte = 8 bitu (0-255) (0-FFH)
Word = 16 bitu (0-65535) (0-FFFFH)
Dword = 32 bitu (Double word, Long)

CHECK TYPE - LOW, HIGH

Kontrolní součet je ukládán běžným způsobem - na nižší adrese leží byte s menším významem a na vyšší adrese leží řádově významnější byte. Uplatňuje se pro typy Word, Dword.

CHECK TYPE - HIGH, LOW

Kontrolní součet je ukládán opačným způsobem - na nižší adrese leží byte s řádově větším významem a na vyšší adrese leží méně významný byte. Uplatňuje se pro typy Word, Dword.

CHECK TYPE - IN ALL CHIPS

Do každé paměti která je programována (programuje-li se jich najednou více) bude uložen její vlastní kontrolní součet.

CHECK TYPE - ONLY ONE

Při programování více pamětí je vypočítán pouze jeden kontrolní součet v šířce všech pamětí a uložen do některé z pamětí podle zadané adresy.

SET ADDRESS

Nastavení adresy pro kontrolní součet. Počítá-li se pouze jeden kontrolní součet zadáváme jeho adresu jako hexadecimální long word (32 bitů, delší soubor se nepředpokládá).

Je-li počítán součet každé paměti, zadáváme adresy pro každou paměť jednotlivě jako hexadecimální word (16 bitů). Paměti jsou číslovány v pořadí v jakém budou požadovány při programování. Při použití součtu v každé paměti je počet omezen na 69.

SET PRODUCTION NUMBER

Nastavení výrobního čísla v paměti. Zobrazená informace ON, OFF určuje zda je výrobní číslo použito.

Pohybem šipkami nahoru a dolů nastavíme položku kterou chceme změnit a klávesou ENTER provedeme změnu parametru. Generaci výrobního čísla lze ovlivnit následujícími příkazy:

Number size
Number type
Autoincrement
Set address for number
Set number value

NUMBER SIZE

Určuje kolik místa je vyhrazeno pro výrobní číslo v paměti.

Byte = 8 bitů (0-255) (0-FFH)
Word = 16 bitů (0-65535) (0-FFFFH)
Dword = 32 bitů (Double word, Long)

NUMBER TYPE

Určuje formát výrobního čísla:

Binary : Binární číslo zadáváme hexadecimálně a je zvětšované běžným způsobem.
BCD : Číslo ve formátu BCD, tak je také zvětšováno.
Ascii : 1, 2 nebo 4 ASCII znaky, jde-li o číslo je zobrazeno decimálně a také tak zvětšováno. Volba formátu má význam hlavně současně s funkcí [Autoincrement].

AUTOINCREMENT

Zapnutí/vypnutí automatického zvyšování výrobního čísla. Po každém úspěšné dokončeném programování je číslo zvětšeno o jedničku. Výrobní číslo je zvyšováno způsobem odpovídajícím jeho typu a šířce. U typu BCD a Ascii, neodpovídá-li hodnota čísla typu, je výsledek po zvýšení o 1 nezaručen. Hodnota výrobního čísla je ukládána v parametrech paměti a je dobré před ukončením práce konfiguraci uložit aby byla zachována hodnota posledního výrobního čísla.

SET ADDRESS FOR NUMBER

Nastavení adresy pro výrobní číslo. Adresu na kterou se uloží výrobní číslo do paměti zadáváme jako hexadecimální long word (32 bitu, 8 hexadecimálních číslic).

SET NUMBER VALUE

Nastavíme počáteční hodnotu výrobního čísla způsobem odpovídajícím jeho typu a ve zvolené šířce.

ČTENÍ A ZÁPIS DATOVÉHO SOUBORU

LOAD

Otevření souboru pro programování. Zadáme jméno souboru ze kterého se bude programovat do paměti. Zadáme-li wildcards (znaky *, ?) otevře se okno s nabídkou souboru. Implicitně je nabízen adresář zvolený v menu Option [Set directory]. Implicitní jméno souboru je *.ROM. Existuje-li konfigurace parametrů k souboru (jméno.RCF) je automaticky použita a zobrazena.

Mikrobáze hobby 1/91

SAVE

Načtení paměti a uložení do souboru. Zadáme jméno souboru do kterého se uloží přečtené paměti a podle pokynů programu vkládáme jednotlivé kusy paměti (je-li jich více). Konec načítání oznámíme stlačením klávesy ESCAPE (program to sám nabídne). Při čtení se uplatní volba zakřížení dat i adres a volba šířky datové zbernice. Uplatní se také volba formátu [File filter].

DOKUMENTAČNÍ SOUBOR

Dokumentační soubor vytváříme pomocí příkazu [EDIT COMMENTARY]. V plnohodnotném textovém editoru píšeme vlastní pomocné informace k paměti. Text se ukládá v konfiguraci paměti viz. [Write configuration]. Ovládání je shodné jako u editorů typu WS. Při psaní textu popřípadě při jeho načtení s konfiguračním souborem jsou zobrazovány ve spodní části obrazovky 3 řádky textu.

ČTENÍ/ZÁPIS KONFIGURACE

WRITE CONFIGURATION

Uložení všech parametrů programování. Do souboru jsou ukládány všechny parametry programování a komentářový text napsaný v editoru. Soubor parametrů má pevnou příponu .RCF. Je-li otevřen nějaký soubor (příkaz [Load]), jsou parametry uloženy do souboru téhož jména (i adresáře) s příponou .RCF. Není-li soubor otevřen je proveden dotaz na jméno souboru do něhož se mají parametry uložit.

READ CONFIGURATION

Načtení parametrů programování ze souboru. Je dán dotaz na jméno souboru (povinná přípona .RCF) a parametry jsou z něj načteny. Zobrazí se textový komentář a všechny parametry. Je-li otevřen soubor, který měl parametry uloženy v souboru stejného jména, ale s extenzí .RCF, pak byly tyto parametry již automaticky přečteny příkazem [Load].

OPERACE S PAMĚTÍ

MEMORY CLEAR TEST

Kontrola smazání paměti. Výsledek je oznámen :

empty = paměť je prázdná
not empty = paměť není smazaná

Důsledky plynoucí z nesmazané paměti můžeme vyhodnotit příkazem [Verify memory-editor]. V režimu programování ze souboru jsou obě kontroly prováděny automaticky.

WRITE TO MEMORY

Zápis do paměti. V režimu datového editoru je zahájeno programování. Programovány jsou pouze ty byty, které jsou odlišné od propalovaného data.

V režimu programování ze souboru podle pokynů programu vkládáme jednotlivé kusy paměti. Proběhne automaticky otestování na smazání a případně i porovnání s programovaným obsahem. Záporný výsledek je uživateli oznámen. Potom je paměť naprogramována a program případně požádá o další paměť.

READ MEMORY TO EDITOR

Načtení obsahu paměti do editoru. Obsah již vložené paměti je přečten do datového editoru. Nelze použít v režimu programování ze souboru.

VERIFY MEMORY - EDITOR

Porovnání obsahu paměti s datovým editorem. Jsou-li nalezeny chyby je nabídnuto jejich výsá-ní.

Warning - paměť lze přeprogramovat na požadovaný obsah

Error - paměť nelze programovat na požadovaný obsah.

V režimu programování ze souboru jsou postupně kontrolovány všechny paměti tak jak byly programo-vány.

REŽIM DATOVÉ EDITACE

Režim datové editace vyvoláme z úvodního menu příkazem [QUIT]. Po zadání tohoto příkazu se na obrazovce objeví editační režim. Obrazovka editač-ního režimu zobrazuje v horním řádku základní pří-kazy, v dolním řádku jsou příkazy, které lze kdy-koliv vyvolat a střední část obrazovky je vyhraze-na pro zobrazení datového okna. Po navolení příka-zu dochází u některých k zobrazení podmenu. Příka-zy můžeme vyvolat nastavením prosvětlení (směrový-mi šipkami) na požadovaný příkaz a stisknutím klá-vesy ENTER nebo současným stiskem ALT klávesy a klávesy odpovídající zvýrazněnému znaku příkazu.

Výjimku tvoří příkaz QUIT, u kterého je nutné spo-lu s klávesou ALT stisknout klávesu X. Horní řádka nabízí následující volby:

File, Edit, Memory, Option, Programming, Change color, Quit

FILE

Příkaz umožňující práci ze soubory, vymazání obsahu datového okna, změnu adresáře pro zápis resp. čtení souborů, dočasné opuštění programu a ukončení činnosti programu. Položky podmenu pří-kazu FILE:

LOAD - F3

Příkaz LOAD načte zadaný datový soubor do dato-vého bufferu velikosti odpovídajícího zvolené pa-měti. V případě, že datový bufer je velikostí od-lišný od načítaného souboru, je o tom dáno hláše-ní. Vstupní soubor je "oříznut" pokud je větší, a doplněn hexadecimální hodnotou FF je-li menší než je velikost bufferu.

PICK ALT-F3

Při vyvolání příkazu je podána informace o třech naposledy načtených datových souborech a je obsluha umožněno si z této nabídky vybrat da-tový soubor, který se načte.

NEW

Maže obsah datového bufferu. Po vykonání příka-zu je v celém bufferu hodnota OFFH.

SAVE F2

Ukládá obsah datového bufferu do souboru stej-ného jména jako má soubor, který byl do datového buferu načten příkazem LOAD.

WRITE TO

Zapisuje obsah bufferu do souboru jehož jméno musíme zadat.

CHANGE DIR

Mění adresář ve kterém se budou realizovat pří-kazy LOAD, SAVE a WRITE TO.

OS SHELL

Program je dočasně přerušen. Uživatelé jsou k dispozici všechny příkazy Operačního Systému. Návrat do programu je proveden příkazem EXIT. Při návratu jsou všechna data i parametry zachovány.

QUIT ALT-X

Ukončení činnosti programu a odchod do Operač-ního Systému. Podle potřeby zároveň uloží barevné instalace a změny v parametrech.

EDIT

Datový editor umožňuje vytvářet popřípadě měnit obsah datového bufferu.

PŘÍKAZY PRO OVLÁDÁNÍ KURZORU

Cursor Left	Left
Cursor Right	Right
Cursor Up	Up
Cursor Down	Dn
Byte (word) right	Ctrl Right
Page up	PgUp
Page down	PgDn
Top of editor	Ctrl PgUp
Bottom of editor	Ctrl PgDn

Funkční klávesou F9 lze vyvolat podmenu edito-ru. Jeho příkazy vykonávají tyto pomocné funkce:

Pomocne funkce editoru - LOCAL MENU

Goto	Ctrl G
Search expression	Ctrl S
Next search	Ctrl N
Insert mode	Ctrl I
Display mode	Ctrl D
Block commands	Ctrl B

GOTO

Nová adresa kurzoru. Adresu zadáváme hexadeci-málně nebo dekadicky podle zvoleného typu editoru.

SEARCH

Search expression. Zadej výraz který se bude hledat od polohy kurzoru dále. Výraz může být za-dán v následujících tvarech:

číslo - byte (0-255 nebo 0-FF)

číslo - word (0-65535 nebo 0-FFFF)

řetězec - text uzavřený v uvozovkách

Zadávané-li sekvenci znaků můžeme v řetězci použít znak *. Hvězdičkou označujeme naprosto libovolný byte nebo znak.

Hledáme-li sekvenci více bytů nebo řetězců, oděluje je jednotlivé výrazy znakem mezera. Číslo zadáváme hexadecimálně nebo dekadicky, podle zvoleného typu editoru.

NEXT

Next search. Pokračuj ve vyhledávání dříve zadaného řetězce od pozice kursoru dále.

INSERT

Insert mode. Přepne editor do insert režimu. Je aktivní jen v režimu byt-ascii. V tomto režimu je možné vkládat ASCII znaky nebo nibble (4 bity, jedna hexadecimální číslice). Klávesou DEL provádíme opačnou funkci. Data posunutá za konec bufferu "přetékají" do vyrovnávací paměti velikosti 1 KB a funkcí DEL se opět navrací do editoru.

DISPLAY

Display mode. Přepínání režimu editoru.

BYTE - ASCII = Hexadecimální byte editor s Ascii editací znaku.

WORD = Hexadecimální word editor.

BINARY = Binární editor. Vhodný např. pro vytváření znakových generátorů.

DEC = Dekadický byte editor.

Umožňuje-li jeden editor více typů editace, přepínají se klávesou TAB.

BLOCK

Block commands. Blokované funkce editoru:

Fill - Plnění paměti konstantou.
Copy - Přenesení bloku paměti.
Exchange - Výměna dvou bloků paměti.
Read - Načtení bloku ze souboru.
Write - Uložení bloku do souboru.

FILL

Plnění paměti konstantou. Začátek a konec bloku paměti, který bude vyplněn (včetně těchto mezí) zadáváme hexadecimálně nebo dekadicky podle typu režimu editoru. Výraz, kterým se bude paměť vyplňovat, zadáváme stejně jako u funkce Search. Včetně znaku '*', který značí, že v editoru bude ponechán původní obsah.

COPY

Přenesení bloku paměti. Adresové meze bloku a adresu od které se bude blok ukládat zadáváme hexadecimálně nebo dekadicky, podle typu editoru. Blok je přenášán včetně mezí. Zdrojová a cílová oblast se mohou překrývat.

EXCHANGE

Výměna obsahu dvou bloků paměti. Adresové meze bloků zadáváme hexadecimálně nebo dekadicky podle typu editoru. Výměna proběhne včetně hranic bloků. Tyto paměťové oblasti se nesmějí překrývat.

READ

Načtení bloku ze souboru. Zadáme jméno souboru ze kterého se bude načítat celá délka souboru. Obsah bufferu bude obsahem souboru přepsán. Soubor se ukládá od pozice kursoru. Přesahuje-li délka souboru konec paměti, je to oznámeno. Neuplatňují se vstupní konverze souboru.

WRITE

Uložení bloku do souboru. Zadáváme hexadecimálně nebo dekadicky, podle typu editoru, začátek a konec bloku paměti, který bude uložen (včetně těchto mezí). Dále musíme určit jméno souboru do něhož se vymezený blok запиše. Neuplatňují se vstupní konverze souboru.

FUNCTION

Vyplňování paměti funkcí. Velmi účinný nástroj pro vytváření různých dekodérů. Výstupní stav Yn paměti lze definovat pomocí předcházejícího stavu výstupu Dn (vhodné pro negování a pod.) a stavu vstupních adres An. Se všemi uvedenými operandy lze provádět základní operace uvedené v tabulce.

Parametry: D0-D7 předcházející stav bitu v editoru.
Y0-Y7 nové spočítané bity (musí být už vypočteny)
A0-A15 adresové bity počítané buňky

Operátory: '- negace, nejvyšší priorita
'* log. součin, nižší priorita
'+' log. součet,
'^ operace neekvivalence (xor)
'(' závorky na změnu priority. Počet úrovní závorek není omezen.

MEMORY

Volba typu paměti.

OPTION

Určení parametrů programu. Umožňuje nastavit následující parametry:

DIRECTORY

Volba implicitně nabízeného adresáře.

Input directory :

Adresář pro načítání a ukládání souborů paměti. Je nabídnut při zadávání jmen souborů. Není-li zadán žádný adresář, je použit implicitní adresář DOSu.

Working directory :

Adresář ve kterém jsou vytvářeny pomocné soubory. Vhodné lze použít např. ramdisk.

FILE FORMAT

Volba formátu načítaných a ukládaných dat.

BYTE - Soubor se ukládá resp. čte beze změn, tak jak je zobrazen v editoru, nebo jak je uložen v ROM paměti.

INTEL HEX - Soubor je ukládán ve formátu firmy INTEL, předepsaném pro zadávání obsahu paměti. Délka souboru je omezena na 64KB. (Používá se také pro data na děrné pásce).

SET MARK

Nastavení ukazatelů v binárním editoru. Volba Mark on - off určí zda se značky zobrazují. Length mark je vzdálenost mezi značkami. Vhodné při vytváření znakových generátorů k označení délky znaku - zvyšuje přehlednost editoru.

CONFIG

Zajištění automatického uložení parametrů. Volba ON způsobí, že při opuštění programovacích oken dojde k automatickému uložení programovacích parametrů (viz. [Write configuration]) v případě, že byly měněny. Za změnu parametru se považuje

i nárůst výrobního čísla. Není-li známo jméno souboru s parametry, provede se dotaz na toto jméno.

HELP LANGUAGE

Přepínač pro jazykovou verzi nápovědy. Lze volit mezi angličtinou a češtinou.

PROGRAMMING

Uživateli je nabídnuto úvodní okno programovacích funkcí. Není-li připojen programovací přípravek, je uživatel upozorněn hlášením chyby.

CHANGE COLOR

Změna barevné instalace programu. Barevná instalace je uchovávána spolu s parametry [Option] v souboru jména "PROGRAM.CFG", který leží ve stejném adresáři jako program a je pouze jeden. V případě potřeby jej lze smazat. Program potom nastartuje se základní barevnou instalací.

QUIT

Ukončení činnosti programu a odchod do operačního systému. Je-li uplatněna volba [Config] - On, dochází k uložení parametrů do souboru.

SEZNAM SOUČÁSTEK

COMMANDER			MALÝ PROGRAMÁTOR EPROM			D8	KY199	1ks	R34	1R	1ks
						D9	LEDR	1ks	R35	100k	1ks
U1,U2	8286	2ks	U1	8255	1ks	R1,R10	330	2ks	R38	8k2	1ks
U3	74LS138	1ks	U2,U3,U7	8216	3ks	R2,R3,R4,R5,R6,R7	1k2	6ks	R39	3k3	1ks
U4,U5	74LS85	2ks	U4,U5	8282	2ks	R8,R43	150	2ks	R40	4k7	1ks
U6	74LS00	1ks	U6	LM7805	2ks	R9	68	1ks	R41	470R	1ks
U7	74123	1ks	U8,U9	74LS175	2ks	R11,R12,R14,R15,			C1	50M	1ks
D1	KAS21	1ks	U10	74LS04	1ks	R16,R18,R42	1k5	7ks	C2	M1	1ks
D2	LED	1ks	U11	7442	1ks	R13,R32,R33,R37	1k	4ks	C3	100M/25	1ks
R1	1k5	1ks	U12	7407	1ks	R17,R19,R20,R36	2k2	4ks	C4,C9,C10	4n7	3ks
R2,R3	15k	2ks	U13	4052	1ks	R21	12k	1ks	C5	10M/100	1ks
R4	150	1ks	U14	B260	1ks	R22,R28	3k3*	2ks	C6	2n2	1ks
RPACK1	8*560	1ks	Q1,Q2,Q3,Q8	KC635	4ks	R23	1k2	1ks	C7	3n3	1ks
C1	220	1ks	Q4,Q6,Q7,Q9	KC636	4ks	R24,R26	470*	2ks	C8	220M/40	1ks
C2	22M	1ks	Q5	KC237	1ks	R25,R27	10	2ks	T1	TRAFO	1ks
SW1	SW DIP-4	1ks	Q10	KD139	1ks	R29	470	1ks	K1,K2,K3	RELAY 12V	3ks
SW2	SW DIP-8	1ks	D1,D2,D3,D5,D6,D7	KA207	6ks	R30	6k8	1ks			
			D4	LEDG	1ks	R31	120R	1ks			

NAVÍJECÍ PŘEDPIS TRAFU MALÉHO PROGRAMÁTORU

Ferit. hrníček ϕ 24 mm, materiál H 21.

Primár: 30z. D=0.3 až 0.5 mm v případě nedělené kostičky se vine dospod a jednotlivé vrstvy se neprokládají.

Sekundár: 120z. D=0.3 až 0.5 mm v případě nedělené kostičky se vine na primární vinutí.

Primární a sekundární vinutí je nutné oddělit dvěma vrstvami transformátorového papíru.

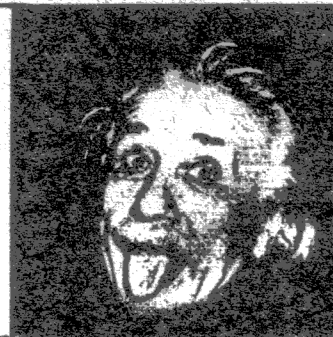
Obě půlky feritů distancujem podložkou 0,7 mm z izolačního materiálu.

Celý transformátor nakonec vyvaříme v parafinu. Pro upevnění je nutno použít mosazný šroubek.

Tento článek nemá být jen pokusem o jeden z mnoha dalších programátorů, šlo spíše o to seznámit čtenáře s problematikou programování pamětí EPROM a na příkladu ukázat jedno z možných řešení programátoru. Je brán zvláštní zřetel na parametry, ve kterých jiné programátory nevyhověly. V případě zájmu zajistí redakce pro své čtenáře dodávku desek plošných spojů. Pro ty, kteří mají o uvedený programátor zájem a nehodlají se zabývat jeho stavbou, může *Mikrobáze hobby* dodat i hotové výrobky. Podle posledního jednání s výrobcem by cena kompletního programátoru včetně luxusního programového vybavení pro IBM PC neměla přesáhnout

4100,-Kčs. Samotné programové vybavení pro IBM PC je možno získat na dobírku hotové na disketě (výpis programu zde nemůže být uveden, protože program v kódu procesoru 8088 má přes 200KB, takže jeho přepsání do počítače je nereálné, a program ve zdrojovém jazyce potřebuje speciální knihovny, které nejsou běžné k dispozici). Součástí programového vybavení je i program pro oživení a vyzkoušení programátoru. Pro zájemce je možno zajistit i dodávku hotového profesionálního typu programátoru, o kterém je v článku zmínka. Ten je ovšem určen pro profesionální využití, čemuž také odpovídá cena. Pište na adresu redakce.

Programovací jazyky a jejich porovnání (1)



ing. Rudolf Pecinovský CSC.

Úvod

S nástupem nové generace počítačů se opět začala na mnohých pracovištích přetřásat otázka "nejlepšího" programovacího jazyka. Všichni jsou zajedno v tom, že by bylo ideální, kdyby všichni programovali ve stejném jazyce, nejsou se však většinou schopni dohodnout na tom, který jazyk by to měl být. Pokusím se vám proto ukázat výhody a nevýhody některých nejznámějších programovacích jazyků a možná tak někomu pomoci v nesnadném rozhodování.

Celé naše povídání si rozdělíme na dvě části: v první si stručně povíme nejzákladnější charakteristiku většiny jazyků, na něž se často odvolávají autoři knih a článků o programování a v druhé části si potom na jednoduchém programu ukážeme konkrétní vlastnosti těch z nich, které jsou v současné době na osobních počítačích nejpoužívanější.

Na počátku úvah o nejvhodnějším programovacím jazyce by si měl každý nejprve odpovědět na následující otázky:

1. Na jaké typy úloh bude daný programovací jazyk používán, neboli:

a) Zda se bude jednat o:

- **Programy vědecko-technického rázu.**
Očekáváme mnoho výpočtů s relativně malým počtem dat. Přivítáme zejména bohaté knihovny numerických metod. Častou podmínkou bývá možnost práce s reálnými čísly ve dvojnásobné přesnosti.
- **Programy na hromadné zpracování dat.**
Očekáváme nepříliš složité operace se spoustou dat. Oceníme zejména bohaté možnosti práce se soubory a zvláště pak možnosti práce se soubory ve formátu některé rozšířené databáze. Často potřebujeme takový formát zobrazení čísel, aby se nám při práci s miliardami neztrácelo halíře.
- **Programy systémového zaměření.**
Lidově řečeno, žádná "špatnost" nám nebude cizí. Budeme potřebovat možnost práce s absolutními adresami, přímý přístup k jednotlivým bitům, možnost přímé spolupráce s hardwarem, bezproblémovou spolupráci s podprogramy ve strojovém kódu. Na druhou stranu většinou nepotřebujeme pracovat s reálnými čísly, tím méně s čísly ve dvojitě přesnosti.
- **Programy pro práci v reálném čase.**
Typickým rysem těchto programů je potřeba co nejrychlejší odezvy na vnější podněty. Nesmí nastat situace, že by počítač, zatímco zpracovává nějaký podnět, propásl podnět jiný, možná daleko významnější. Bu-

deme proto hledat jazyk, který nám umožní pokud možno bezpečně naprogramovat paralelní zpracování několika úloh: vlastních reakcí na podněty a současného hlídání podnětů dalších.

- **Měření, regulace, řízení.**

Vrcholně náročná záležitost. Nejen že nás čekají nepříjemnosti předchozích dvou bodů (práce s bity, absolutní adresace, spolupráce s hardwarem a assemblerem, práce v reálném čase, ...) a že asi nezůstaneme ušetření ani práce s reálnými čísly (samozřejmě s maximalistickými požadavky na rychlost), ale navíc se velmi často stává, že náš program je určen pro zcela jiný počítač, než je ten, na němž jej připravujeme. Potřebujeme proto takové prostředky, které nám kromě všech ostatních služeb umožní převést náš program do nového prostředí. A pokud se liší nejenom počítač, ale i použitý procesor, můžeme si vybírat pouze z těch jazyků, pro něž umíme sehnat tzv. křížový překladač.

b) Zda se bude jednat o programy jednodušší (stovky příkazů), středně složité (tisíce příkazů), složité (do desetitisíců příkazů) a nebo velmi složité.

c) Zda v našich úlohách vystačíme se základními datovými typy (jednoduché proměnné, vektory, matice - sem patří např. většina vědecko-technických úloh) nebo zda budeme potřebovat i typ záznam (většina úloh hromadného zpracování dat) a nebo zda budeme dokonce nuceni používat dynamických datových struktur (např. při práci v reálném čase).

2. Zda bude program vytvářet jednotlivce nebo úzká skupina zasvěcených a nebo zda se bude jednat o dílo širokého okruhu tvůrců.

3. Jaká bude úroveň programátorů, kteří budou ve vybírání jazyka programovat a jaké budou jejich předchozí zkušenosti.

4. Na kolik je volba programovacího jazyka svazována nutností navázat na dříve naprogramované úlohy a zůstat proto kompatibilní s jazyky, v nichž byly tyto úlohy naprogramovány. Tuto povinnou kompatibilitu můžeme zařadit do jedné z následujících hladin:

a) Nic nám nesvazuje ruce, naše volba je naprosto svobodná.

b) Programátoři jsou zvyklí na nějaký jazyk, ale na žádné dříve zpracované projekty navazovat nemusíme (i kdyby snad něco bylo dříve

naprogramováno, budeme stejné dělat vše úplně znovu).

c) Doposud zpracované úlohy lze používat buď ve formě knihoven a nebo ve formě modulů, které lze přisestavit k vytvářenému programu. Neplánujeme žádné závažnější zásahy do zdrojového tvaru dříve napsaných modulů.

d) Nelze vyloučit větší zásahy do dříve napsaných programů.

5. Jaké jsou požadavky na přenositelnost výsledného produktu na jiné počítače.

Pokusím se vám nyní nastínit charakteristiku nejnámějších (všimněte si - neříkám nejpoužívanějších) programovacích jazyků a zároveň se pokusím naznačit při jakých kombinacích odpovědi na předchozí otázky by byla volba daného jazyka výhodná. Abych neupadl v podezření, že nějakému jazyku nadřazuji (stejně upadnu), budeme probírat jednotlivé jazyky podle abecedy.

Ada

Největší veledílo v dosavadní historii překladačů. Jazyk, v němž má být možno naprogramovat naprosto všechno. Návrh jazyka vyhrál konkurs na universální programovací jazyk, vypsaný ministerstvem obrany USA. To však ještě neznamená, že jazyk byl ihned k dispozici. V žádném případě! K dispozici byl pouze onen vítězný návrh normy. A trvalo několik let, než se objevily prvé plody následného usilovného vývoje mnoha navzájem si konkurujících firem. Tvůrci Ady se totiž dopustili stejné chyby, jaké se 15 let před nimi dopustili autoři jazyka PL/1 - chtěli, aby v jejich jazyku nechyběla pokud možno žádná z vymožeností kteréhokoliv z ostatních jazyků, aby podporoval veškeré zásady moderního programování, aby co největší procento chyb odhalil již překladač, aby, aby, aby... Zkrátka kladli si příliš maximalistické cíle, takže vzniklo monstrum, o jehož některých vlastnostech si programátoři dokonce z počátku ani nebyli jisti, že je dokáží naprogramovat. Proto také trvalo tak dlouho, než první plnohodnotná Ada spatřila světlo světa.

Nad čistotou jednotlivých implementací Ady bdí zvláštní komise a žádná firma nesmí svůj produkt prohlásit za Adu dříve, dokud jej komise nepodrobí souboru několika tisíc testů a neprohlásí, že je to skutečná, pravá a nefalšovaná Ada. Tím je samozřejmě zaručena naprostá kompatibilita jednotlivých implementací a tedy i následných produktů.

Ada vychází (ostatně jako většina moderních programovacích jazyků) z programovacího jazyka Pascal. Rozšiřuje jej o relativně jednoduché, avšak mocné prostředky programování synchronizace paralelních procesů - tzv. *rendesvous* (čti randevu), o řadu nejrůznějších "cingrlátek", na jejichž užitečnost není jednotný názor a zejména o možnost odděleného ladění mezimodulového rozhraní a následné samostatné ladění jednotlivých modulů. To je velice důležité při týmové práci, zejména u týmu velmi rozsáhlých, např. právě těch, které řeší mamutí projekty ministerstva obrany USA. Chyby v mezimodulovém rozhraní jsou u takovýchto obřích programů jedny z nejzávažnějších. Tváří se nenápadně, ale dokáží natropit velikou spoušť. Připočteme-li k tomu, že americké ministerstvo obrany, které je největším zákazníkem softwarových firem, prohlásilo, že programy v žádném jiném programovacím jazyce brát nebude, nelze se divit rychlému rozšíření Ady mezi jeho dodavatele.

Adě je vytýkána řada věcí. Především její obrovitost, která nutně vede k tomu, že ji většina programátorů bude znát jenom zčásti. Další výhrady jsou k mechanismu *rendesvous*, který má z hlediska implementace některé velmi nepříjemné vlastnosti. Třetí výhrada vytýká Adě, že nepodporuje tzv. **objektové programování**, které se začalo ve světě silně prosazovat záhy po uvedení prvních verzí Ady na trh. V době vzniku Ady byly sice myšlenky objektového programování již známy, ale živořily na pokraji zájmu a mezi programátorskou veřejnost začaly prosakovat až po širším zavedení jazyka Smalltalk a zejména pak jazyka C++.

Jak vyplývá z předchozího povídání, po Adě se možná vyplatí shánět, máme-li před sebou obrovský projekt. Upozorňuji však, že se jedná o monstrum, které lze ve své kompletní, certifikované podobě získat pouze za velmi vysokou cenu a kromě toho je na něj embargo. Proto nepředpokládám, že by se čtenáři začali v nejbližší době houfně vrhat na programování v Adě (to, co koluje po republice, jsou jen slabé odvary nehodné toho jména). Berte proto zmínku o Adě jako pouhou zmínku o jazyku, který je v současné době na popředí zájmu a možná některé z vás ve vaší budoucí praxi čeká (koluji zprávy o pracech na československé verzi plnohodnotného překladače).

Algol 60

Dalo by se říci, že Algol byl prvním výkřikem moderního programování. A výkřikem silným. Algol byl první z jazyků, jejichž tvůrci zastávali názor, že je rozumné navrhnout jazyk trochu "upovídanější" a tím poskytnout překladači informace, na jejichž podkladě pak dokáže odhalit řadu v programech často se vyskytujících chyb.

Původní podoba jazyka vznikla v roce 1956 jako společné dílo evropských a amerických matematiků. Po publikaci jeho definice se objevila řada připomínek. Reakcí na ně byla revidovaná zpráva o jazyku Algol, která spatřila světlo světa v roce 1958. Ani ta však nebyla dokonale a tak v roce 1960 vychází druhá revidovaná zpráva. Touto zprávou definovaný jazyk dostal jméno Algol 60 (píše-li někdo samotné jméno Algol, mívá tím většinou na mysli právě Algol 60).

Algol 60 začal novou kapitolu v historii programovacích jazyků a ovlivnil i další vývoj programování jako takového. Stal se praotcem většiny moderních programovacích jazyků protože z něj buď přímo vycházely (Simula 67, Algol 68, Pascal,...), nebo vycházely z některého z jeho potomků, většinou z Pascalské větve (Ada, Chill, Edison, ...) a nebo jim alespoň byly výrazně ovlivněny (PL/1, C,...).

Ačkoliv se tento jazyk stále ještě na některých vysokých školách vyučuje, musíme konstatovat, že doba jeho slávy již nenávratně skončila. Jeho místo v současné době převzal jeho přímý potomek - Pascal.

Algol 68

Jazyk vznikl - jak ostatně napovídá číslo ve jméně - v roce 1968. Přestože vznikl obdobným způsobem jako Algol 60, nedoznal ani vzdálené jeho úspěchů o rozšíření ani nemluvě. Proč? Protože byl tak rafinovaný, že si skoro nikdo netroufl napsat na něj překladač. Takže v současné době se s ním setkává spíše prostřednictvím odvolávek v učebnicích než prostřednictvím chodivých programů, které by byly v tomto jazyku napsány a odladěny.

APL

Název jazyka je akronymem z anglického A Programming Language ((nějaký) programovací jazyk). Byl navržen pro použití v dialogových systémech. Při práci v tomto jazyku musíte používat upravenou klávesnici, protože zavádí spoustu nových symbolů, které na běžných klávesnicích nejsou (kolečka, čtverečky, nabla, ...). Díky jim je zápis programů nesmírně kompaktní. Oblíbeným sportem vyznavačů tohoto jazyka je vymýšlení problémů, které by bylo třeba v jiných jazycích napsat přinejmenším na celou stránku a které lze v APL zapsat na jediný řádek. Tato kompaktnost ovšem zákonitě vede k tomu, že za tři dny nechápe program ani autor sám.

Jazyk se zpočátku poměrně rychle šířil, ale v současné době se již používá pouze omezeně, i když jej zdaleka nemůžeme považovat za odepsaný. Jeho přívrženci se stále ještě domnívají, že dobu vrcholného rozkvětu má tento jazyk terpve před sebou a že čeká pouze na dostatečně výkonnou malou výpočetní techniku. Považují jej totiž za ideální jazyk vědeckých kalkulaček.

Assembler

Programování v assembleru považuje řada programátorů za jediné opravdové programování, protože jediné tak mohou z počítače vyždímat vše, čeho je schopen. Nedejte však na jejich silácké řeči. To, že oni umí nějaký assembler, ještě neznamená, že s ním stojí a padá programátorský svět. Opak je pravdou. V assembleru v současné době programují již pouze zoufalci. Buďto jsou to zoufalci z přesvědčení, kteří v něm programují ze zásady, a nebo jsou to zoufalci z donucení, kteří v něm programují proto, že jejich technické a programové vybavení jim danou úlohu jinak naprogramovat nedovoluje. (mezi tyto zoufalce z donucení je nutno počítat majitele ZX Spectra a dalších osmibitových počítačů se značně omezeným rozsahem operační paměti a s nevelkou rychlostí - pozn. red.)

Na důkaz toho, že v celém světě se od programování v assembleru pokud možno ustupuje, mohu uvést, že např. ze 13 000 řádků zdrojového textu slavného operačního systému UNIX je jich v assembleru napsáno pouze 800, což by mohlo odpovídat přibližně 1% až 2% programu (nezapomínejte, že každý řádek ve vyšším programovacím jazyce se přeloží do několika strojových instrukcí). Pro všechny tvůrce programového vybavení je nejdůležitější být na trhu se svým výtvozem první. A to by se jim při programování v assembleru nemuselo podařit. A tak se k němu všichni uchylují pouze v těch nejnutnějších případech.

Typickými třídami úloh, programovaných prakticky výhradně v assembleru, je programování jednočipových mikropočítačů pro řídicí, měřicí a regulační účely a programování klíčových, časově kritických rutin operačních systémů a systémů práce v reálném čase. V případě časově kritických rutin se situace ještě velmi dlouho nezlepší, protože ve chvíli, kdy záleží na každém "clocku" chce mít každý programátor celý proces plně ve svých rukou. V případě jednočipových mikropočítačů sice dodávna většinou ani nic jiného než křížový assembler neexistovalo. V poslední době se však situace lepší a rychle přibývají mikropočítače, které lze programovat v jazyce C (např. 8051), případně některém dalším.

Každý profesionální programátor by měl assembler svého počítače znát stejně, jako by měl každý automobilista umět vyměnit třeba svíčky. Tato znalost mu může za prvé pomoci mnohem rychleji odhalit některé obzvláště záluďné chyby a za druhé některé speciální rutiny je někdy přeci jenom lepší napsat v assembleru. To jsou ty rutiny, kdy při

malém zvýšení úsilí můžeme dosáhnout velký efekt (např. přesun bloků paměti, základní ošetření velmi častých přerušení apod.).

Jestliže používáte assembler, mějte jej vždy pouze jako doplňkový jazyk a ne jako hlavní jazyk celé aplikace. Doba, kdy čas jsou peníze, začíná pronikat i k nám. Uvažujte občas o hesle:

Program pracuje pomalu?

Kupte si rychlejší počítač!

Kromě toho mají programy v assembleru ještě jednu nevýhodu: přejdete-li na nový počítač, většinou je musíte celé přepsat.

Basic

Basic vznikl v polovině šedesátých let jako jazyk pro laické uživatele počítačů (např. pro inženýry z jiných oborů), kteří si byli svoje jednoduché problémy ochotni naprogramovat sami. Vznikl jako jazyk, kterým se mělo zvýšit využití terminálové sítě velkého počítače a který měl sloužit řádovým inženýrům jako lepší kalkulačka. Základním cílem bylo, aby práce s ním byla co nejjednodušší a nevyžadovala předchozí studium.

S postupným zlevňováním výpočetní techniky a zejména pak s nástupem minipočítačů, počet lidí, kteří si chtěli řešit na počítači svoje problémy, rychle rostl. Všichni ti, jejichž problémy nebyly příliš složité a kteří se nechťeli zatěžovat učebním nějakého rafinovaného programovacího jazyka, vitali Basic téměř jako svoji spásu. Zejména když se začaly objevovat jeho dokonalejší verze, které již uměly pracovat s řetězci, s celými maticemi a vektory a měly ještě řadu dalších užitečných funkcí. A počet vyznavačů Basicu se ještě mohutně rozrostl s nástupem mikropočítačů, které byly jeden čas bez zabudovaného interpretu jazyka Basic prakticky neprodejně.

Jednou z velkých předností Basicu byla možnost relativně komfortního ladění u jiných jazyků prakticky nedosažitelná. Uživatel si mohl program krokodovat, sledovat průběžné stav jednotlivých proměnných, měnit za chodu jejich obsah, spouštět program prakticky z libovolného místa - to byla aktiva, která Basicu nemohl nikdo upřít. Prostředky, které by umožňovaly odladit ve zdrojovém tvaru i programy napsané v jiných jazycích, tzv. zdrojové debugery, se na mikropočítačích objevily až v posledních několika letech.

Počátkem osmdesátých let byl Basic pravděpodobně nejpoužívanějším programovacím jazykem na světě. Mikropočítač, který neměl Basic, byl prakticky neprodejný. To pochopila i firma IBM a její počítače PC mají Basic pro jistotu také zabudován (počítač se ale do Basicu přepne, až když si ověří, že uživatel nemá k dispozici žádný lepší prostředky). Dokonce se objevily i mikroprocesory přímo programovatelné v Basicu!

Opojení Basicem však trvalo pouze do chvíle, než si uživatelé začali uvědomovat nekonečné možnosti i těch nejmenších počítačů a chtěli je pověřit složitějšími úkoly. Tu se najednou začalo ukazovat, že na ně Basic nestačí. Prvním omezením byla rychlost - vznikly tedy překladače. Pak přišlo na řadu formátování výstupu, spolupráce s hardware, práce se soubory, strukturované příkazy, modularita, práce v reálném čase. Nastala druhá éra vylepšování Basicu. Ten přidal to, ten přidal ono a začalo vznikat něco na způsob Velorexu s bulldozrovou radlicí a elektrickým stahováním oken. A aby to nebylo tak jednoduché, každý samozřejmě přidával po svém. Vznikla záplava navzájem nekompatibilních implementací, pro něž je téměř nemožné vymyslet prakticky použitelný program tak, aby fungoval na všech verzích beze změny.

(pokračování příště)



NABÍDKA



Na tomto místě našeho časopisu budete pravidelně nacházet nabídku programů, součástek, plošných spojů a služeb, které redakce **Mikrobáze hobby** zajišťuje pro své čtenáře. Doufám, že naše nabídka bude pro vás dost pestrá a atraktivní. Budou se zde postupně objevovat i programy, součástky a služby jiných organizací, ovšem vždy až po ověření serióznosti nabídky některým z našich spolupracovníků. Kdo by měl tip na něco zajímavého pro naše čtenáře, dejte vědět. Autoři zajímavých programů, nabídněte své výtvary naší redakci k šíření.

Velice mě mrzí, že nabídka součástek, plošných spojů a dílů byla původně daleko zajímavější, především v cenách. Po dvojí devalvaci koruny však bylo nutno celou nabídku přepracovat a řadu položek vypustit.

Vzniklé torzo dostalo další citelný zásah, když vypukla živelná liberalizace cen. Cenová smršť, která se rozpoutala mne donutila, abych celou

dvustranu ještě v tiskárně vyměnil za novou, ze které řada položek vypadla a ve které musely být upraveny ceny. Týká se to především nabídky integrovaných obvodů. V původní nabídce totiž například byl řadič WD 1793 za 650,- Kčs, dnes to však vypadá téměř na dvojnásobek. Uvidíme, co bude možno nabídnout ve druhém čísle.

Doba, vymknuta z kloubů, šílí - říká klasik. Věřím však, že až toto číslo vyjde (měl by to být přelom ledna a února) bude cenová hladina už poněkud klidnější. Zatím jsem byl například výrobcí plošných spojů informován, že Gumon Bratislava, jakožto monopolní výrobce Cuprexitu, zvýšil cenu na více než trojnásobek. Jak se to promítne do ceny plošných spojů, to lze tušit. A jsou jiné a jiné věci...

Ale konec nařikání - čtěte dál, snad tam přeci jen něco zůstalo.

D.M.

1. Programy

Mikrobáze hobby nabízí svým předplatitelům programy pro různé typy počítačů a to na kazetě či disketě. Ceny pro předplatitele jsou stlačené obvykle na nejnižší možnou úroveň, prakticky jen na režijní náklady. Pokud to druh programu vyžaduje, je doplněn brožurkou s návodem. Některé programy však návod vůbec nepotřebují, případně je v nich návod obsažen. Pak jsou vybaveny jen základní informací ke spuštění, či instalaci. Ke každému programu dostanete číslovaný ústřížek, který slouží jako potvrzení o legálním užívání. K ceně každého programu dodaného poštou bude připočítáno poštovné. Předpokládáme, že ceny programů jsou natolik přijatelné, aby už nedocházelo k nelegálnímu kopírování programů, které je nejen trestné podle platných zákonů, ale především je hluboce nemorální a bezohledné k autorům. U většiny programů je stanoven poplatek, který by měl redakci **Mikrobáze hobby** poslat každý, kdo si program sám nakopíruje. Je to odměna pro autora. Za to obdrží obratem potvrzení o tom, že je legálním uživatelem. Bude to mít mimo jiné význam při tzv. *update*, tedy výměně staré verze programu za novější.

Programy mohou dostat i ostatní čtenáři, kteří nejsou předplatiteli, ovšem cena bude poněkud vyšší. Také v pořadí vyřizování objednávek mají předplatitelé přednost. Protože si čtenáři neradi své výtisky stříhají, budeme identifikovat předplatitele podle kartotéky a ostatní čtenáři se mohou prokázat horním označeným růžkem této stránky.

Diskové verze programů je možné nakopírovat i na zaslanou disketu. Ta však musí být prázdná, bezchybná a naformátovaná ve formátu DD 40 stop SS, nebo DS (max. 360KB) pro 5 $\frac{1}{4}$ ", případně totéž, ale 80 stop (max. 720KB) pro 3 $\frac{1}{2}$ ". Na dodané kazety programy nenahráváme, vzhledem k problémům s kvalitou kazet a zdoluhavému verifikování.

TeletextZX

- program pro příjem teletextového vysílání pomocí adaptéru popsaného ve zpravodaji **Mikrobáze** č. 7+9/89. Popis programu je v tomto čísle **Mikrobáze hobby**.

ZX Spectrum - kazeta+brožurka.

Cena pro předplatitele 80,-Kčs
Cena pro ostatní čtenáře 90,-Kčs
Samostatný registrační poplatek 20,-Kčs

ZX Spectrum - disketa 5 $\frac{1}{4}$ " (Betadisk)+brožurka.

Cena pro předplat. .. (podle diskety) max. 70,-Kčs
Ostatním zatím nedistribujeme.

ČeštinaPC

- úplná luxusní programová podpora českého a slovenského jazyka na IBM PC kompatibilních. Driver klávesnice, download a font designer pro EGA a VGA, editor rozložení kláves, dva definiční soubory klávesnice a český COUNTRY.SYS. Je možno použít kódů Kamenických, Latin 2, KOI8CS a HPGG. Součástí je i program pro vzájemný převod mezi jednotlivými kódy, případně na ASCII. Originální řešení tvorby rámečků a zadávání původních ASCII znaků. Podrobný popis naleznete v příštím čísle **Mikrobáze hobby**.

Cena pro neprofesionálního uživatele 250,-Kčs

Cena pro profesionálního uživatele 500,-Kčs

Cena za multilicenci pro jeden podnik .. 2000,-Kčs

Cena za licenci pro další prodej jako součást vlastního progr. vybavení 10000,-Kčs

Cena za licenci k dalšímu volnému prodeji 20000,-Kčs



ČeštinaPC - minimální verze

- programová podpora českého jazyka na IBM PC kompatibilních. Driver klávesnice, download pro EGA a VGA. Čeština pouze v kódu Kamenických. Část programového balíku z naší nabídky - ČeštinaPC.

Cena pro předplat. ... (podle diskety) max. 70,-Kčs
Cena pro ostatní (podle diskety) max. 80,-Kčs
Samostatný registrační poplatek 30,-Kčs

ProgramátorPC

- luxusní program na IBM PC pro obsluhu programátoru, popsáno v tomto čísle *Mikrobáze hobby*. Doplněn programem pro oživení hardvéru.

IBM PC/XT/AT - disketa 5¼" (360KB)

Cena pro předplat. ... (podle diskety) max. 70,-Kčs
Cena pro ostatní (podle diskety) max. 80,-Kčs
Samostatný registrační poplatek 30,-Kčs

Připravujeme:

CP/M pro ZX Spectrum, univerzální formátovací program pro Betadisk, testovací program pro Betadisk, Shareware a Public domain programy pro IBM PC, utility pro Betadisk, utility pro CP/M.

2. Plošné spoje

Vzhledem k cenové liberalizaci zde došlo k velkým změnám, takže ceny jsou jen orientační a nemůžeme v současné době ani zaručit dodávku.

Teletext

- pro adaptér popsán ve zpravodaji *Mikrobáze* č. 7+9/89. Oboustranná neprokováná deska. Obslužný program, verze pro ZX Spectrum je popsán v tomto čísle.

Orientační cena pro předplatitele 85,-Kčs
Ostatním zatím nedistribuuje.

Programátor EPROM

- na programátor popsán v tomto čísle. Předlohy jsou na obálce tohoto čísla. Tři oboustranné prokované desky.

Orientační cena pro předplatitele 400,-Kčs
Ostatním zatím nedistribuuje.

Betadisk

- dvě oboustranné prokované desky na řadič pružných disků pro ZX Spectrum. Popis začal ve zpravodaji *Mikrobáze* č. 5/89 a pokračuje již sedmým dílem v tomto čísle *Mikrobáze hobby*.

Orientační cena pro předplatitele do 680,-Kčs
Ostatním zatím nedistribuuje.

BOBO 64KB (256KB)

- jedna oboustranná prokována deska na počítač kompatibilní se ZX Spectrem. Popis byl ve zpravodaji *Mikrobáze* č. 6+8/89. Návod na rozšíření paměti je v tomto čísle *Mikrobáze hobby*.

Orientační cena pro předplatitele 400,-Kčs
Ostatním zatím nedistribuuje.

3. Součástky

Tato část nabídky musela být z výše uvedených důvodů zatím stažena. Velice se omlouváme, není naší vinou, že se ceny tak rozhoupaly. Přesto se s požadavky na součástky můžete obrátit na naši redakci, která je předá k vyřízení některé z firem, se kterými jsme jednali. Možná už příště budeme schopni dodat ceník s alespoň přibližnými cenami.

Mikrobáze hobby 1/91

Pokud někdo ví o spolehlivé firmě, dejte vědět. Máme zájem o co nejpestřejší nabídku a nejnižší ceny. Chceme čtenářům sloužit, ne je odírat.

4. Kompletní doplňky

Programátor EPROM pro IBM PC

- kompletní fungující programátor EPROM, popsán v tomto čísle. Foto je na zadní straně obálky. Funguje pro typy EPROM od 2716 do 27512. Určený k připojení na počítače IBM PC/XT/AT a kompatibilní. Součástí dodávky je luxusní programové vybavení na disketě. Prodej i na fakturu.

Cena pro všechny čtenáře do 4200,-Kčs
Vzhledem k omezenému množství mají předplatitelé přednost.

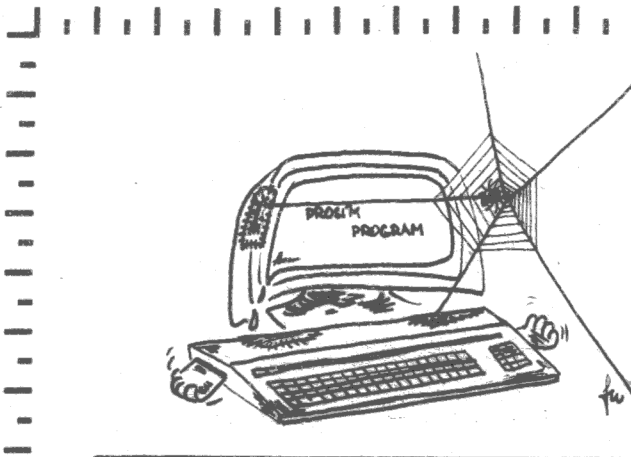
5. Služby

Programování EPROM

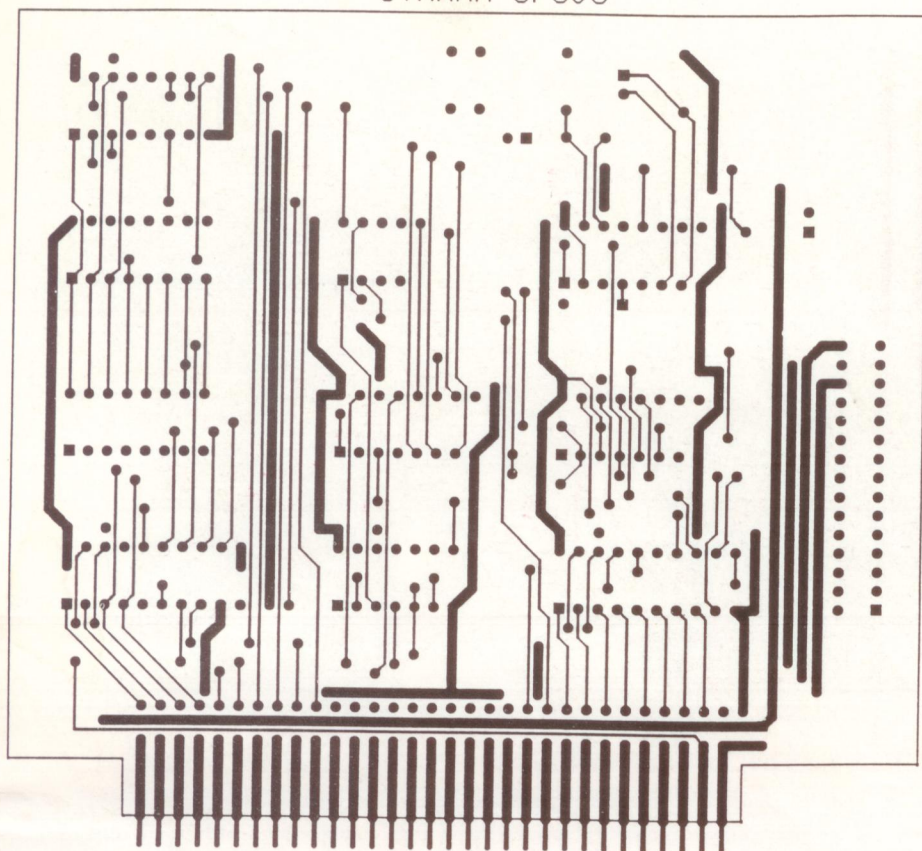
- pro své čtenáře zajišťujeme naprogramování různých EPROM do konstrukcí popsanych v *Mikrobázi hobby*, ale i do jiných. Je možné naprogramovat EPROM od 2716 do 27512, 27513 a 27011. V případě požadavku naprogramování jiného obsahu, než podle našich konstrukcí, je nutno se předem informovat na naše možnosti, nebo dodat požadovaný obsah EPROM na disketě pro IBM PC 5¼" 360KB, případně 3¼" 720KB, CP/M Lecsoft, Betadisk, nebo na kazetě pro ZX Spectrum. Vlastní záznam musí být ve formátu HEX INTEL, nebo BIN (u Betadisku a kazety nejlépe CODE). Přesný popis typu záznamu a žádaný způsob programování nutno uvést v průvodním dopise. Paměť dodaná k naprogramování by měla být smazaná (smažeme za příplatek), nepoškozená a s nezkrácenými vývody. Cena podle pracnosti a délky. Orientační cena (27128 - Betadisk) 50,-Kčs

Všechny ceny uvedené v této nabídce se budeme snažit podržet alespoň do výjití příštího čísla *Mikrobáze hobby*. Zatím se nám nezdá příliš pravděpodobné, že by jejich tendence byla klesající. Neváhejte proto s nákupem. Kdo zaváhá ...

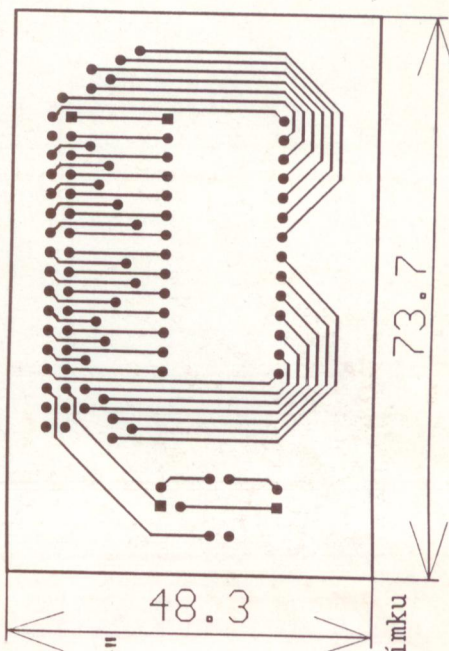
Ve svých objednávkách prosím uvádějte přesnou adresu a PSČ, druh a množství objednávaného zboží, případně (u orientačních cen) jaká nejvyšší cena by pro vás ještě byla přijatelná (ne že bychom toho chtěli zneužít, ale pro zrychlené vyřízení a omezení zbytečné korespondence).



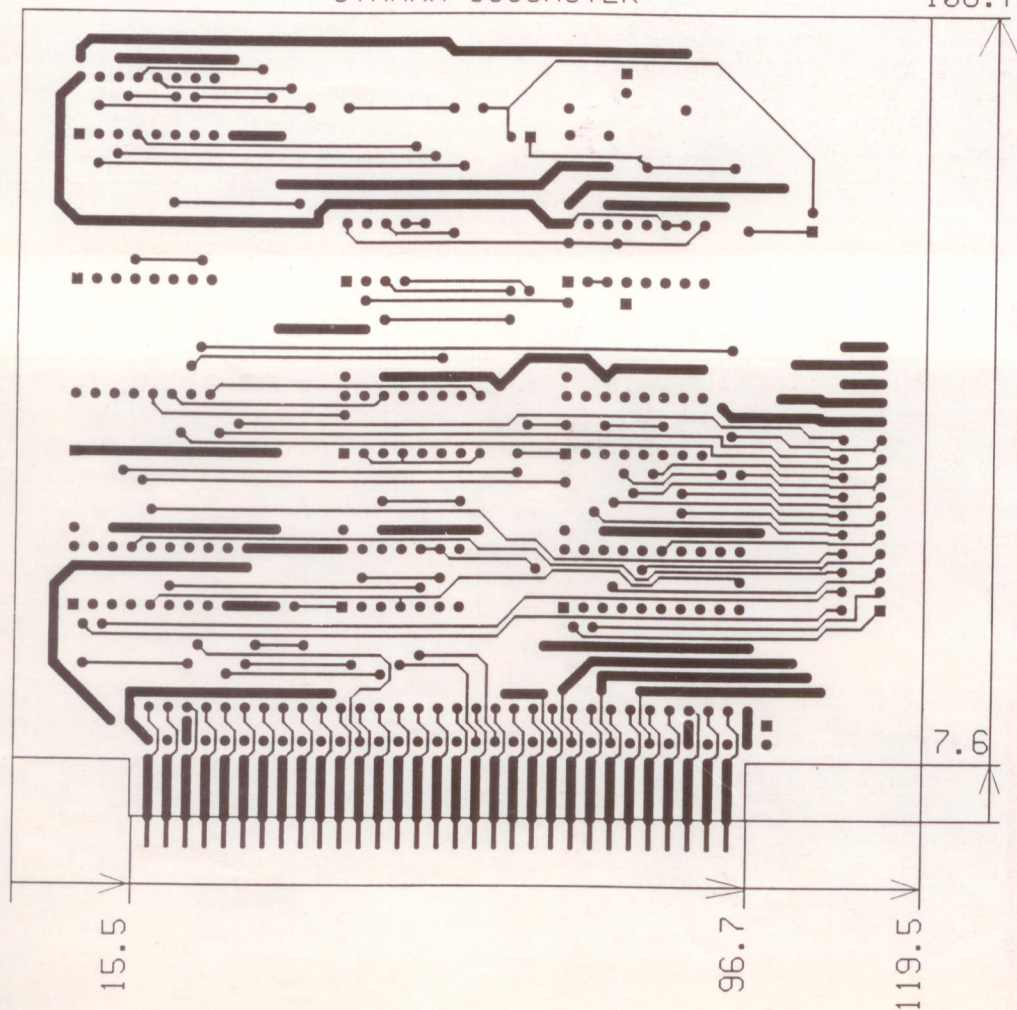
STRANA SPOJU



POHLED
ZE STRANY SOUC.



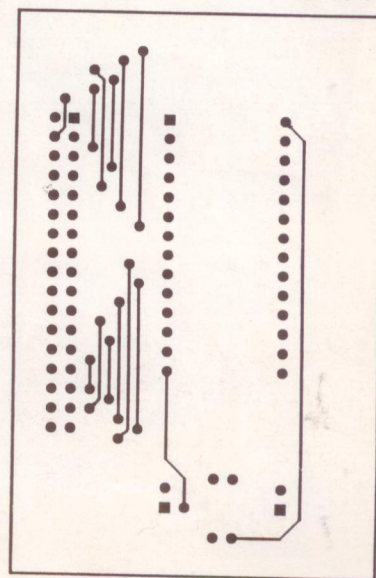
STRANA SOUCASTEK

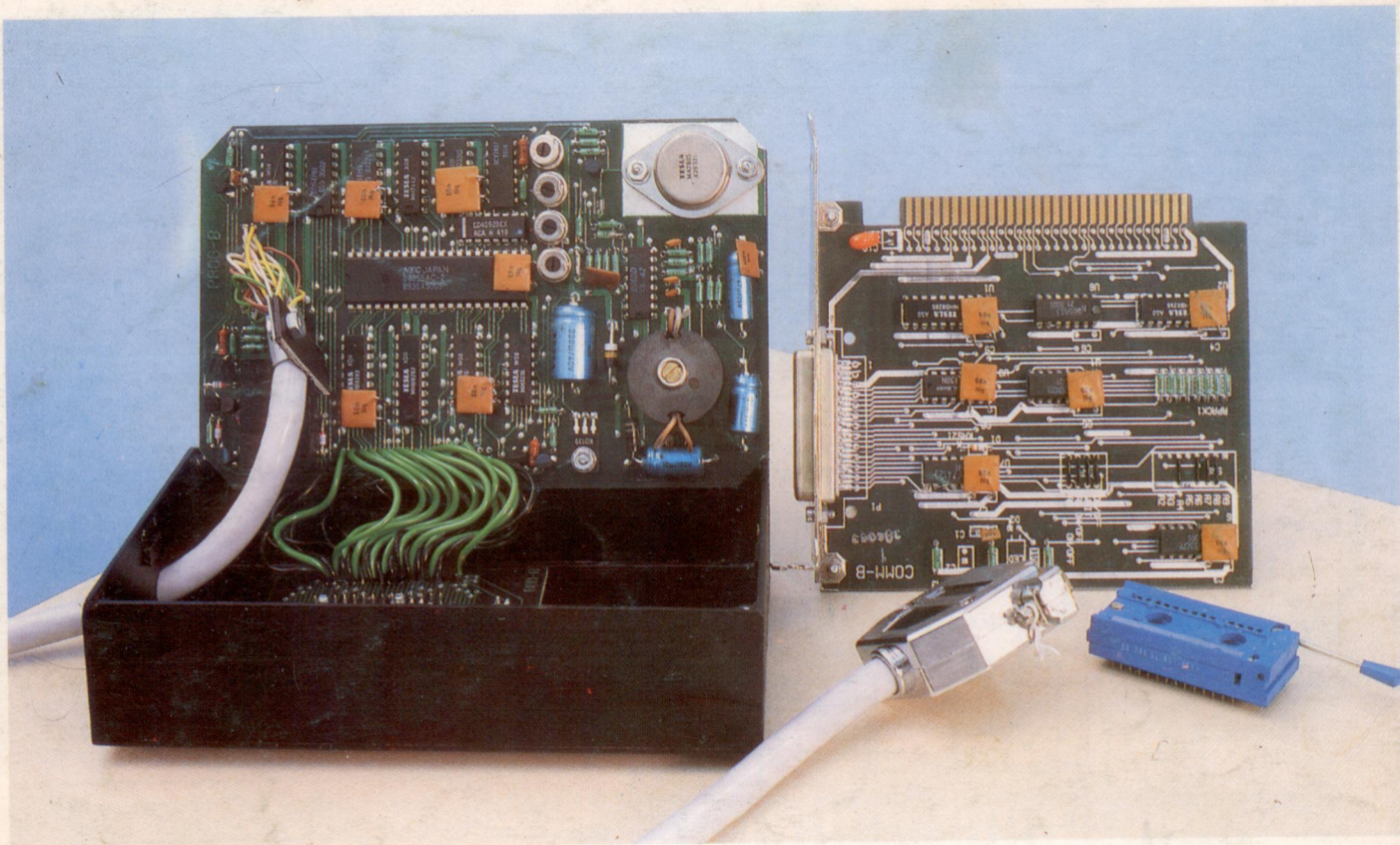


Obrazec plošných spojů desky "Commander"
k programátoru EPROM,
podle popisu v tomto čísle.

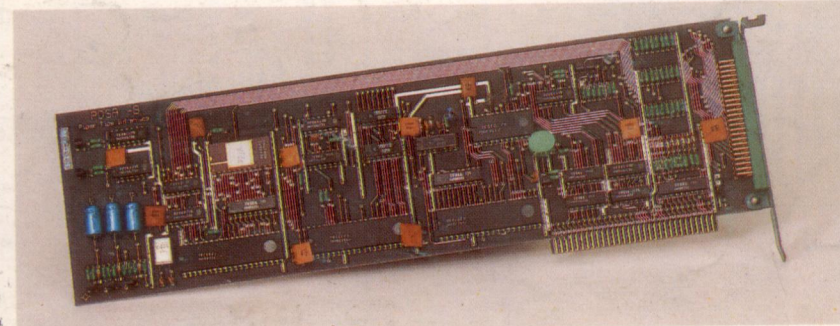
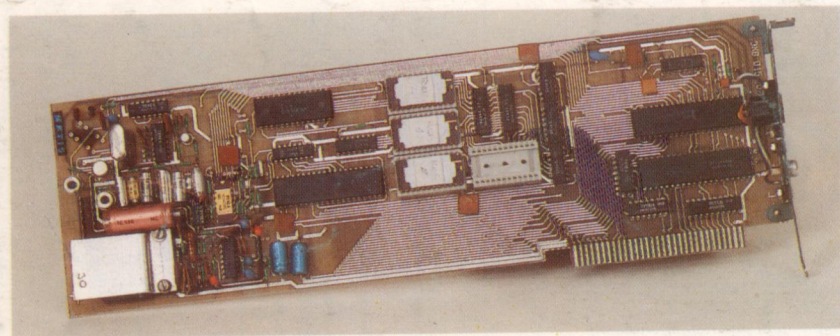
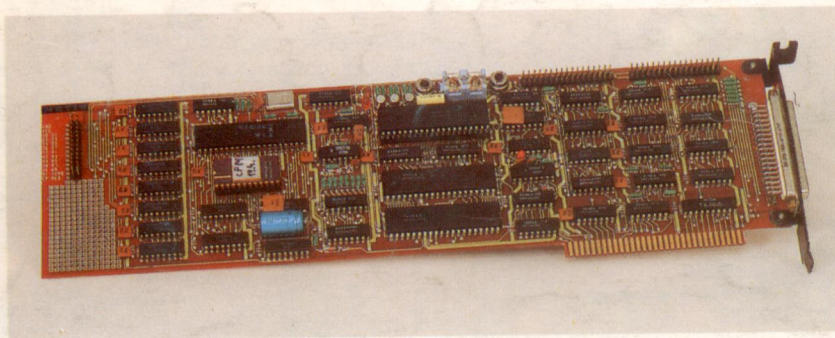
Mikrobáze hobby 1/91

Obrazec plošných spojů destičky pod objímku
na EPROM. (súčást programátoru EPROM,
podle popisu v tomto čísle)





NAŠE REKLAMNÍ LETÁČKY VÁM NELHALY!



V reklamních letáčcích jsme vám slibovali řadu zajímavých ověřených konstrukcí pro osmi, šestnácti a vícebitové počítače. Na připojených obrázcích můžete vidět některé z konstrukcí, připravených do letošního ročníku. Tentokrát vám představujeme část konstrukcí určených k rozšíření IBM PC a kompatibilních.

Nahoře je programátor paměti EPROM, použitelný s malou úpravou i pro osmibitové počítače. Úplný stavební návod je již v tomto čísle. Na dalších obrázcích je shora dolů k vidění: Karta CP/M, karta formantového hlasového výstupu a karta pro připojení děrovače, vybavená vlastní inteligencí.

A ještě něco, podařilo se nám zajistit výrobu a dodávky většiny důležitých dílů a v některých případech také finálních výrobků za zajímavé ceny. Proto čtěte naši nabídku na straně 31! Předplatitelé budou při objednávkách zvýhodněni. Můžeme dodávat i podnikům na fakturu.