

ŘÍJEN

FUN

1993

with Commodore

časopis uživatelů Commodore 64/128

8. číslo

BASIC

C64 A VIDEO

Text. editor ČStext

Vektorový font editor

Hardcopy pre plotter

Zajímavé

produkty

CMD

OBSAH

Úvodní slovo	1
Basic	2
Assembler na C64	6
Tipy a triky	9
C64 a video	11
Představujeme:	
Textový editor ČStext	12
Vektorový font editor	13
Hardcopy pre plotter	15
Zajímavé produkty CMD	17
Dopisy čtenářů	20
Comotronic News	23

CO BUDE V PŘÍŠTÍM ČÍSLE?

Basic – pokračování
Assembler – pokračování
Tipy a triky
C64 a video
Programujeme DEMO

Představujeme:
Geos – Gateway
Postavte si scanner
Comotronic News
Dopisy čtenářů
Inzerce

Adresa redakce: Dolnomlýnská 2, 787 01 Šumperk * Autoři čísla: Jaroslav Vančura a Jiří Kouřil
Podávání novinových zásilek povoleno Oblastní správou pošt v Ostravě pod č. j. 288 / 92 – P1
ze dne 14. 12. 1992

Podávání novinových zásilek povolené SP š. p. ZsRP Bratislava č. j. 613 – PD – 1993 zo dňa 1. apríla 1993
FUN with Commodore – časopis Comotronic klubu pro uživatele počítačů Commodore 64/128.
5 / 93 – 8. číslo
Fotosazba REPROtisk J. Kotinský * Tisk: Vegaprint Šumperk

Vážený čtenáři,

v souladu s naší informací z posledního čísla FUNu zahajujeme naši novou rubriku C64 a video, ve které určitě najdete mnoho nových a zajímavých informací o možnostech Vašeho C64.

Do konečné fáze se dostává i příprava programů pana Billy a proto jsme zařadili do tohoto čísla jeho informace o programech ČStext, vektorovém editoru fontů a grafických hardcopy pro plotry. Tyto informace jsou superzajímavé, neboť se jedná o profesionální, slovenští čtenáři mohou říci tuzemský software, který bude dostupný za vynikající ceny (srovnáme-li s cenami některých českých programů pro C64, prodávaných v Praze).

CMD je zkratka Computer Micro Devices, jinak je to americká firma, dodávající pro počítače C64/128 velice promyšlenou a vzájemně se doplňující řadu výrobků, z nichž některé jsme Vám nabídli v našem katalogu 93 a některé Vám představujeme v tomto čísle.

Díky těmto dodatečně zařazeným článkům jsme nedokázali do čísla vměstnat původně připravené pokračování článku programujeme DEMO. Budeme se snažit Vám tento nedostatek vynahradit v příštím čísle.

Jiří Kouřil
redaktor FUNu

ZÁHADY TISKU

V tomto příspěvku rubriky Basic se budeme zabývat problematikou týkající se tiskáren a tisknutí. Ukážeme, jak si s nimi rozumět a na závěr uvedeme malý program pro hardcopy.

Nejprve se budeme zabývat tiskem textu. Tiskárny patří mezi tzv. inteligentní přístroje, neboť mají vlastní mikroprocesor, paměť a operační systém, což nezřídka vede k problémům, jak se s nimi domluvit.

Stejně jako s počítači dá se i s tiskárnami komunikovat pomocí příkazů. Kterým příkazům tiskárna rozumí a jaké má schopnosti se dá přirovnat k operačnímu systému C64. Většina tiskáren umí více standardů. Pro jednotlivé emulace se dají tiskárny přepnout (např. EPSON, IBM nebo NEC).

U 9 jehličkových tiskáren, ponejvíce rozšířených v domácích výpočetních střediscích, převládá standard ESC/P (Epson Standard Code for Printers). Jím se tedy budeme zabývat. Příklady se ovšem dají lehce převést na jiné emulace, nezbytné příkazy najdete v příručce k tiskárně.

Vedle Epona se týká následující text také tiskáren Star, Seikosha, Mannesmann a dalších.

Datové soubory

Musím připomenout, že C64 posílá všechna data na kterýkoli přístroj jako tzv. datové soubory. Počítači je potom jedno, zda se jedná o magnetofon nebo tiskárnu.

Datový soubor musí být nejprve otevřen prostřednictvím příkazu:

OPEN 1,4

Prvé číslo představuje adresu souboru, pod níž je v dalším průběhu programu datový soubor volán. Druhé číslo je vyhrazeno pro tzv. sekundární adresu. Každý přístroj připojený na seriový port má svou nezaměnitelnou adresu. Tak přísluší tiskárně zpravidla číslo 4. Uvedeným příkazem zaznamená počítač, že k datovému souboru jedná přísluší tiskárna. Se všemi daty uvedeného datového souboru posílá počítač automaticky 4, aby je tiskárna akceptovala. Analogicky by se dalo otevřít až 10 souborů najednou.

Nyní následuje otázka, jak sdělit tiskárně co má dělat. Pro tištění znaků (písmena, čísla) je to velmi jednoduché:

PRINT #1, "COMOTRONIC"

Příkaz PRINT# posílá následující data do datového souboru označeného adresou (zde číslo 1). Soubor jde na tiskárnu.

Pokud nyní tiskárna takový, tisknutelný znak přijme, zapíše jej do vnitřní paměti (buffer) a čeká na další. Pokud je buffer plný nebo pokud je ukončen řádek, znaky vytiskne.

V našem případě se na papíře objeví slovo COMOTRONIC. Chcete-li nyní bezprostředně navázat na tento text a vytisknout další znaky, můžete potlačit řádkový posuv, tedy přepnutí na další řádek. Připojte za příkaz PRINT # značku pro sloupec (;). Pokračujete-li nyní:

PRINT #1, " s.r.o.", objeví se na papíře COMOTRONIC s.r.o.

Doposud to bylo velmi jednoduché. Hovořili jsme rovněž o tisknutelných znacích. Tato formulace předpokládá i existenci znaků netisknutelných. Skutečně jsou takové. Jak možná víte, je každému znaku přiřazeno kódové číslo mezi 0 a 255. V počítači se tato hodnota ukládá a zpracovává ve formě bajtu. Tiskárny využívají taktéž tohoto kódování. Ostatně (nebo naštěstí) obsahuje počítačové písmo podstatně méně znaků (po 26 velkých a malých písmenech, 10 cifer a zvláštní národní znaky). Mimo to přicházejí ještě některé symboly (matematické znaky, řecká písmena apod.). Pro tiskárnu jsou zajímavé kódy 0 až 31. Zde jsou uvedeny řídicí znaky pro tiskárnu, např. ty, které zapínají široké nebo úzké písmo (viz tab.). Tyto znaky ale nenajdete na klávesnici, tiskárně je musíte sdělit jinak:

PRINT #1, CHR\$(X)

X nahradíte číselnou hodnotou. Tiskárna okamžitě zareaguje. Široké písmo zapnete například pomocí: **PRINT #1, CHR\$(14)**

Ještě před několika lety znamenaly tyto řídicí znaky všechno, čemu tiskárny rozuměly. Přístroje jsou však stále modernější a čím dál výkonnější, mají více možností. To vedlo k tomu, že tiskárny se dají přepnout do programového módu. K tomu slouží kód 27 (Escape). Pokud tiskárna rozpozná tento znak, čeká, že následující bajt bude představovat řídicí znak (tzv. Escape sekvence). Pokud chcete tisknout např. pouze 40 znaků na řádek:

PRINT #1, CHR\$(27); "Q", CHR\$(40)

Nejdůležitější ESC kódy jsou připojeny v tabulce níže. Pokud tiskárna přijme kompletní příkaz ESC, přepne sama na tiskový mód. Vyzkoušejte zadat po posledním příkazu:

PRINT #1, "ABCDEFGHJKLMNOPQRSTUVWXYZ- ZABCDEFGHIJKL"

Písmeno L se nyní vytiskne na novém řádku.

Důležité je dodržet velká a malá písmena v rámci jedné ESC sekvence. Jinak by mohlo dojít k nežádoucím efektům, posléze až ke zhroucení tiskárny.

Přejdeme k prvému užití nových poznatků: program, který vytiskne obsah textové obrazovky. Program začíná otevřením datového souboru. Dále je připojena smyčka, která čte znak po znaku obsah videopaměti. Paměť začíná u adresy 1024, má délku 1000 bajtů a je rozložen do 25 řádků po 40 bajtech.

První smyčka počítá řádky (od 0 do 24), druhá znaky ve sloupcích (od 0 do 39). Z těchto údajů se potom vypočítává adresa v obrazovkové paměti:

$X = 1024 + (A * 40) + BA$ je řádek, B sloupec obrazovky. Paměť se čte jako obvykle příkazy PEEK v řádku 10130.

Co nyní následuje je převod kódů. Commodore, žel nepracuje v ASCII kódu, nýbrž používá vlastní kódové tabulky. Např. v ASCII je A přiřazen kód 65, u Commodore 1. Naštěstí pořadí kódů zůstává zachováno. Proto postačuje ke kódu přičíst jen hodnoty. Pro obrazovkové kódy pod 96 musíme přidat 64. Na hodnoty nad 128 nebereme zřetel (grafické znaky), proto stiskneme space (v módu Epson nejsou grafické znaky k dispozici). Převod kódů zajistí smyčka IF v řádku 50.

V proměnné X máme nyní hotový znak připravený k vytisknutí pomocí příkazu PRINT#. Důležitá je značka (;) na konci řádku, potlačuje řádkový posuv.

Pokud bylo vytištěno všech 40 znaků jednoho řádku, musíme přejít na řádek následující. To znamená, že musíme zařídit řádkový posuv. K tomuto účelu existuje speciální příkaz (ASCII kód 13), který po každém zpracování smyčky pošleme na tiskárnu. Alternativně se dá použít i příkaz:

PRINT #1, " " Nic tím sice nevytiskneme, ale pošleme díky chybějícímu středníku řádkový posuv.

Stejným způsobem se pokračuje dál, dokud není zpracováno všech 25 řádků. Nakonec přijde ještě jeden důležitý příkaz. Pomocí CLOSE uzavřeme datový soubor. C64 vyšle na tiskárnu kód k ukončení práce. Tiskárna reaguje tak, že vytiskne ještě všechny znaky, které jsou v bufferu.

Program Hardcopy textu

```
10 OPEN 1,4 <089>
20 FOR A=0 TO 24 <203>
30 : FOR B=0 TO 39 <143>
40 : L=PEEK(1024+A*40+B) <240>
50 : IF L<32 THEN L=L+64 <125>
60 : PRINT#1,CHR$(L); <015>
70 : NEXT B <102>
80 : PRINT#1,CHR$(13) <212>
```

```
90 : NEXT A <110>
100 CLOSE 1 <111>
109 END <111>
```

Tiskové příkazy

ASCII kód	název funkce
07	BEL signál na bzučák tiskárny
08	BS o jeden znak zpět
09	HT horizontálně o jeden tabulátor dále
10	LF řádkový posuv
11	VT vertikálně o jeden tabulátor dále
12	FF stránkový posuv
13	CR vrací tiskovou hlavu zpět na začátek řádku
14	SO široké písmo pro jeden řádek zapnuto
15	SI úzké písmo pro jeden řádek zapnuto
17	DC1 přepíná tiskárnu na ON LINE
18	DC2 vypíná úzké písmo
19	DC3 přepíná tiskárnu na OFF LINE
20	DC4 vypíná široké písmo
24	CAN vymaže buffer
27	ESC následují znak nebo posloupnost má zvláštní význam
127	DEL vymaže naposled přijatý znak

sekvence ESC

27,14	ESC SO	široké písmo zapnuto
27,15	ESC SI	úzké písmo zapnuto
27,45	ESC -	podtrhávání zapnuto
27,14,n	ESC/n	nastavit vertikální tabulátor
27,49	ESC 1	vzdálenost řádků 7/72 inch
27,51,n	ESC 3 n	vzdálenost řádků n/216 inch
27,54	ESC 6	rozšiřuje rozsah tisknutelných znaků
27,56	ESC 8	hlídač konce papíru vypnut
27,57	ESC 9	hlídač konce papíru zapnut
27,60	ESC <	jednosměrný tisk
27,64	ESC @	reset tiskárny
27,65,CHR\$(n)	ESC A	vzdálenost řádků n/72 inch
27,67,CHR\$(n)	ESC C	délka stránky určena počtem řádků
27,67,48,CHR\$(n)	ESC C (0)	délka stránky určena v inch
27,69	ESC E	tlusté písmo zapnuto
27,70	ESC F	tlusté písmo vypnuto
27,77	ESC M	písmo Elite
27,80	ESC P	písmo Pica
27,81,CHR\$(n)	ESC Q	nastavit pravý okraj

Zatímco tisk textu se jeví jako relativně jednoduchý,

jde v případě obrázků, které se mají přenést na papír, o podstatně složitější věc.

V dalším uvažujme rozšířené 9 jehličkové tiskárny kompatibilní s Epson, které využívají standardu ESC/P. Protože tyto tiskárny pracují s hlavičkou v níž je uloženo 9 jehliček, mohou přenést na papír najednou 9 bodů.

Při tisku grafiky se přenáší na tiskárnu pro každý bod přesně jeden bit. 0 odpovídá bodu, který se nevytiskne (mezeře), 1 bodu vytištěnému. Protože devítka je pro počet bitů nepříjemné číslo, využívá se pouze 8 jehel. Proto se v jednom bajtu dají přenášet všechna data pro jeden krok tiskárny. Každému bitu v bajtu odpovídá potom přesně jedna jehla. Bit s nejnižším významem patří k nejspodnější, nejvýznamnější bit potom k jehle zcela nahoře. Pokud se tedy na tiskárnu pošle hodnota 129, vytisknou se dva protilehlé body s šestibodovou mezerou.

Tiskárně se však musí nejprve sdělit, že následující bajt nemá představovat žádné písmeno nebo číslo, nýbrž grafiku. Sekvence ESC má potom následující stavbu:

```
CHR$(27);"***";CHR$(MOD);CHR$(n1);CHR$(n2);
```

Příkaz, ač vyhlíží komplikovaně, je v podstatě velmi jednoduchý: CHR\$(27) je uvádějící ESC příkaz. Hvězdička označuje mód „BIT-MAP“. Sděluje tedy tiskárně, že má tisknout vysoce rozlišitelnou grafiku.

Na tomto místě si dovoluji poznámku k tomu, že tiskárny rozeznávají několik druhů grafických módů. Přehled uvádí tabulka.

Grafické módy 9 jehličkových tiskáren.

m	módus	rozlišení (bod/inch)
0	normal	60
1	dual	120
2	dual, double speed	120
3	quadruple	240
4	CRT-grafika	80
5	plotr	72
6	CRT-grafika 2	90

Jednotlivé módy se liší v hustotě bodů. Pracuje-li tiskárna v grafickém módu, rozumí každému z 256 možných bajtů (%00000000 až %11111111). Nebere proto na hodnotu 27 žádný ohled a chápe ji jako grafický znak. Protože neexistuje možnost tiskárnu nějakým ESC příkazem vypnout, musí se na tuto skutečnost pamatovat dopředu. Tiskárně se musí sdělit údaj o počtu grafických dat, která počítač vyšle. Po zpracování celého plánovaného balíku dat se přepne zpět na textový mód a opět rozumí všem příkazům.

Čísla n1 a n2 jsou teď Low- a High- bajty tohoto počtu. Jejich hodnota se vypočítá podle vztahu:

$$n2 = \text{INT}(\text{pocet}/256)$$

$$n1 = \text{pocet} - (n2 * 256)$$

Při experimentování tuto skutečnost respektujte. Záměna obou hodnot přináší neočekávané efekty (tiskárna čeká grafická data přičemž její systém se zhroutí...).

Proto abyste odeslali na tiskárnu data, musíte nejprve otevřít odpovídající soubor:

OPEN 1,4,X

X použijte pro sekundární adresu lineárního kanálu. Zpravidla se dává 1 nebo 4, někdy též 7. To způsobí, že všechna data postoupí z počítače na tiskárnu aniž by prošla konverzí. To má u grafických dat enormní vliv na vizáž obrázku.

Dále připojte příkaz pro tiskárnu:

```
PRINT#1,CHR$(27);"***";CHR$(1);CHR$(8);CHR$(0);
```

Důležitý je středník na konci. Chybí-li vyše počítač ještě jeden řádkový posuv (ASCII kód 13), který je pochopen jako prvý grafický bajt.

Tiskárna se nyní nachází v grafickém módu a čeká na osm bajtů. Vyzkoušejte zadat:

```
PRINT#1,CHR$(1);CHR$(2);CHR$(4);CHR$(8);CHR$(16);CHR$(32);CHR$(64);CHR$(128)
```

Na papíru se objeví šikmá čára ze spodu zleva nahoru doprava. Pokud snad připojíte ještě jeden bajt, např. 65, nevytisknou se žádné body nýbrž znak A. Elementární základy tisku grafiky jste tímto doufám zvládli. Zbývá tedy pojednat o vzdálenosti řádků.

Na rozdíl od tisku textů se nesmějí objevit mezi jednotlivými řádky žádné mezery. Vzdálenost řádků je u tiskáren proměnná a může být přestavitelná pomocí ESC sekvence. Uvažujme. Normální velikost bodů činí 1/72 inch. Osm bodů bezprostředně vedle sebe zabere 8/72 inch. Přesně o tuto vzdálenost se musí papír posunout.

Příslušný příkaz má formát:

```
CHR$(27);"A";CHR$(n)
```

Přitom „n“ označuje vzdálenost v n/72 inch. Zadejte tedy 8 a vzdálenost bude sedět.

Tato vzdálenost zůstane zachována i při zpětném přepnutí do textového módu do té doby, než ji znovu změníte.

Popsaným příkazem dosáhnete zajímavých efektů. Nastavíte-li posuv např. na 1, pohne se papír o šíři jediného bodu. Pokud přejedete stejný řádek ještě jedenkrát dostanete druh tlustého tisku. Vydělají na tom zvláště obrázky s šikmými čarami, neboť schůdky a proluky se zaplní.

Abych vás nezatěžoval jen teorií, umožním vám čerstvě nabyté poznatky aplikovat prakticky. Programem hardcopy můžete tisknout jednoduché jednobarevné obrázky.

Program začíná definováním videoadresy (řádek 10). Byla použita hodnota 8192 (\$2000), používá se v 90 % všech případech. Je-li vaše grafika položena v jiné

oblasti, změňte tuto hodnotu. V dalším kroku se otevře soubor se sekundární adresou 1 (lineární kanál). Jako první se vyšle signál k resetování tiskárny. Povel zruší všechna předchozí nastavení (řádek 30). Následuje nastavení odstupe řádků na 8/72 inch (řádek 40). Tímto krokem byly ukončeny všechny přípravy a tiskárna očekává grafická data. Ta musíme číst z videoram bajt po bajtu. C 64 nedodrží pořadí bajtů jako tiskárna, uchýlíme se proto k malému triku. Obrázek se nevytiskne napříč přes papír jak je běžné, ale pootočený o 90°. Tím se podstatně přiblíží formáty tiskárny a C64. RAM se ovšem nebude číst po řádcích nýbrž po sloupcích počínaje na levém spodním okraji obrázku. Vnější smyčka proto běží od 0 do 39, čítá sloupce. Na začátku přepne tiskárna na grafiku na 200 datových bajtů (řádek 70). Jako další následuje smyčka, jíž jsou čítány řádky (řádek 80). Pro správný začátek tisku je směr čítání opačný. Tím teď můžeme určit pozici znaku na obrazovce. Každý znak je kódován osmi bajty, proto potřebujeme ještě třetí smyčku, která čítá zpět od 7 do 0. Zpětně proto, že bajt 7 odpovídá spodním osmi bitům znaku.

Dále následuje přepočítání adresy videopaměti a čtení RAM (řádek 100). Aby bylo vidět jak daleko je počítač s prací, necháme změnit kurzor na aktuálním místě v grafické obrazovce. K tomu jednoduše nastavíme na 1 všechny bity, do příkazu POKE přijde hodnota 255 a v zápětí stará, už jednou čtená hodnota (řádky 110 a 120).

Přečtené bajty se nyní mohou posílat na tiskárnu (řádek 130). Středník na konci příkazu je důležitý, jinak by byl ještě poslán řádkový posuv. Následují dvakrát návěští NEXT, která způsobí další pokračování bajtu a sloupců. Po kompletním zpracování jednoho sloupce se v řádku 160 papír posune a začne se zpracovávat další sloupec. Po vytištění všech sloupců se ještě uzavře datový soubor.

Listing "HARDCOPY". Důležité řádky mají na konci uveden kontrolní součet (Checksum).

```

5 REM V - počáteční adresa video-
paměti
6 REM zde nastavena na $2000
7 REM můžete ji přizpůsobit svému
programu
10 V=8192 <008>
15 REM otevření lineárního kanálu
na tiskárnu
20 OPEN 1,4,1 <006>
25 REM inicializace tiskárny způ-
sobí předání do
26 REM definovaného stavu
30 PRINT#1,CHR$(27);CHR$(64) <130>
35 REM nyní musí být odstup řádků

```

```

nastaven na
36 REM 8/72 inch
40 PRINT#1,CHR$(27); "A";CHR$(8)
<187>
45 REM dohromady máme 40 sloupců
50 FOR A=0 TO 39 <110>
60 REM tiskárna se nastaví pro tisk
61 REM 200 bajtů grafiky
70 PRINT#1,CHR$(27); "***";
CHR$(0);CHR$(200);CHR$(0) <112>
75 REM dále se sloupci a to nazpět
80 FOR B=24 TO 0 STEP-1 <242>
85 REM každá pozice obsahuje osm
bajtů
90 FOR C=7 TO 0 STEP-1 <144>
95 REM díky pootočenému tisku je
přepočítání adresy
96 REM podstatně jednodušší
100 X=PEEK(V+A*8+B*320+C) <189>
105 REM necháme blikat kurzor
106 REM tím poznáme jak daleko je
počítač s prací
110 POKE(V+A*8+B*320+C),255 <017>
120 POKE(V+A*8+B*320+C),X <083>
125 REM bajt se pošle na tiskárnu
130 PRINT#1,CHR$(X); <252>
135 REM osm bajtů hotovo?
140 NEXT C <176>
145 REM ano, potom sloupec hotov?
150 NEXT B <178>
155 REM pokud ano, odeslat řádkový
posuv
160 PRINT#1,CHR$(10);CHR$(13)
<155>
165 REM obrázek komplet?
170 NEXT A <190>
175 REM ano, potom uzavřít soubor
180 CLOSE 1 <191>

```

pozn.

Videokontrolér C64 zapisuje vysoce rozlišitelnou grafiku do paměti v neobvyklém formátu. Nejprve je 8 bodů horního řádku, potom 8 bodů druhého řádku, potom třetího atd. až po osmý. Dále následují body 9 až 16 horního řádku atd. Stavba silně připomíná textový módus, kdy písmena a znaky jsou zobrazeny shodně.

VIC se tím práce ušetří, ovšem programátoři při přepočítávání adres grafických bodů zrovna nadšení nejsou.

V programu Hardcopy bylo použito triku, kdy byl obrázek pootočen o 90°, tiskl se tedy na ležato. Rozdělení paměti se tím přiblížilo tisku grafiky a nároky na přepočítání jsou nyní minimální. (JV)

ASSEMBLER NA C64

(7. pokračování)

POPIS PŘÍKAZŮ PRO STACK

PHA (Push Accumulator)

Pomocí příkazu PHA je možné přenést obsah akumulátoru do stacku. Hodnota se přitom zapíše na adresu \$0100 + SP. Nakonec se ukazatel stacku (SP) sníží o jednu. Obsah akumulátoru není tímto příkazem ovlivněn. Ani příznaky tento příkaz neovlivňuje.

PLA (Pull Accumulator)

Tento příkaz představuje opačný příkaz k právě uvedenému příkazu PHA. Způsobuje nejdříve zvětšení SP o jednu a potom je hodnota uložená na adrese \$0100 + SP zkopírována do akumulátoru. Podle toho, jaká hodnota je do akumulátoru zkopírována, mění se záporný nebo nulový příznak.

PHP (Push Processor status)

Pomocí tohoto příkazu se obsah stavového registru uloží do Stacku. Uložení funguje tak, jak je výše uvedeno.

Tento příkaz má smysl zvláště tehdy, když mají být volány rutiny, aniž se jimi má měnit obsah stavového registru. K tomuto účelu se hodnota SR uloží do Stacku, vyvolá se rutina a po jejím provedení se obsah SR vrátí zpět.

PLP (Pull Processor status)

Pomocí tohoto příkazu se vytáhne bajt ze Stacku a uloží se do stavového registru. Použití příkazu odpovídá příkazu PHP, ale s obrácenou funkcí.

Nyní přicházíme k funkcím, které jsme popisovali zvláště v kapitole o přenosech. Jedná se zvláště o dva příkazy pro manipulaci s ukazatelem Stacku, což je nepostradatelné zvláště při zapnutí počítače.

TXS

Tento příkaz slouží k k přenosu obsahu registru X do ukazatele Stacku. Opačný proces je proveditelný pomocí příkazu

TSX

Ten slouží k přenosu obsahu ukazatele Stacku do X-registru.

Abychom si použití příkazů pro Stack ukázali názorně, uvedeme si několik příkladů.

K tomu účelu prosím aktivujte Váš monitor strojového kódu. Monitor se obvykle hlásí nápisem MONITOR a výpisem obsahu registrů procesoru nebo jen výpisem obsahu registrů:

MONITOR

```
PC SR AC XR YR SP  
; FF00 00 00 FF 00 F8
```

Zapište nyní následující program:

```
A 3000 PHA  
A 3001 BRK
```

Program odstartujte příkazem G 3000 a stiskněte RETURN.

Monitor se teď hlásí:

BREAK

```
PC SR AC XR YR SP  
; 3003 30 00 FF 00 F7
```

Pokud si všimnete stavu SP před a po průběhu programu, zjistíte, že se liší o 1. Obsah SP je po průběhu programu o 1 nižší než před. Jak se to dá vysvětlit?

V našem programu jsme uložili obsah akumulátoru prostřednictvím příkazu PHA do stacku. Již dříve jsme si vysvětlili, že uložení hodnoty do stacku vede ke snížení SP o 1, v našem případě tedy z hodnoty \$F8 na \$F7.

Vyzkoušejte si také obrácený proces. Změňte nejdříve obsah akumulátoru takto:

```
; 3003 30 FF FF 00 F7
```

Změnili jsme tedy obsah akumulátoru a nyní se přesvědčíme, že tomu tak opravdu je. K tomu použijeme následující program.

```
A 3000 PLA  
A 3001 BRK
```


Program odstartujte jako dříve příkazem G 3000

<RETURN>

Ohlásí se nám monitor:

BREAK

```
PC SR AC XR YR SP  
; 3003 32 00 FF 00 F8
```

Ukazatel stacku má tedy opět hodnotu \$F8 a také akumulátor má svůj starý obsah \$00.

K vysvětlení příkazu TXS je tu dále malý výřez z rutiny NMI operačního systému.

```
. F2A4 A2 FF LDX #$FF  
. F2A6 78 SEI  
. F2A7 9A TXS  
. F2A8 D8 CLD
```

Jak vidíte, je nejdříve do registru X uložena hodnota FF. Potom se pomocí příkazu SEI zabrání přerušení, což nás však v našem případě příliš nemusí zajímat. Nyní následuje příkaz, o který nám zvláště jde. Jak jsme si popsali, bude pomocí příkazu TXS obsah registru X použit jako nový ukazatel stacku. Protože jsme předtím uložili do akumulátoru hodnotu \$FF, bude nová hodnota SP také \$FF. Obsah registru X se přitom nezmění.

Nyní ještě kódy probíraných příkazů:

```
PHA: $48 PLA: $68 PHP: $08  
PLP: $28 TXS: $9A TSX: $8A
```

6.14 Příkazy podprogramů JSR a RTS

Na tomto místě jsme se dostali k technice programování podprogramů. Příkazy pro podprogramy jsou zcela jistě dva nejdůležitější příkazy strojového jazyka.

Z Basicu známe dva příkazy – GOSUB a RETURN. Příkaz GOSUB slouží k volání podprogramu, zatímco příkaz RETURN slouží k ukončení podprogramu a návratu na programový řádek, následující za příkazem GOSUB, kde průběh hlavního programu pokračuje. V zásadě se to ve strojovém kódu dělá stejně. Máme k dispozici příkaz k volání podprogramu a další k uzavření podprogramu.

Příkaz k volání podprogramu se jmenuje JSR (Jump SubRoutine) – skoč na podprogram.

Příkaz k ukončení podprogramu se nazývá RTS (Re-

Turn from Subroutine) – návrat z podprogramu.

Aby bylo možno příkazem JSR vyvolat podprogram, musí se k vlastnímu příkazu přidat adresa, na které podprogram začíná. Také adresa následující za příkazem JSR a na kterou se musí program po provedení podprogramu vrátit, aby mohl pokračovat, se musí někam uložit. Jak z dosud diskutované problematiky vyplývá, bude pro uložení adresy nevhodnější stack. Protože stack pracuje principem LIFO, může být do sebe vnořen téměř libovolný počet podprogramů. Jediným omezením je zde velikost stacku.

Nyní si ještě probereme podrobněji funkci příkazů JSR a RTS.

Pokud procesor při provádění programu narazí na kód příkazu JSR, stane se následující:

1. Procesor vezme okamžitou pozici programového čítače, který stojí právě na kódu pro příkaz RTS. K této adrese přidá potom procesor ještě hodnotu 2. Programový čítač stojí nyní na HIGH bajtu adresy podprogramu. (Tato adresa je jak víme rozdělena na HIGH a LOW část jen kvůli procesoru.)
2. Procesor uloží HIGH-bajt adresy do stacku a sníží ukazatel stacku o 1.
3. Potom uloží do stacku i LOW-bajt adresy a sníží hodnotu ukazatele stacku ještě o 1. Dohromady je teď ukazatel stacku o 2 menší, než před voláním podprogramu. Ve stacku je teď adresa na které je příkaz JSR plus 2.
4. Nyní si procesor vytáhne adresu, která stojí za kódem příkazu JSR. Tato adresa udává, kde začíná podprogram. Je uložena do paměti ve formě LOW-HIGH bajt a bude nyní převzata jako nový obsah programového čítače. Procesor nyní může zpracovávat podprogram.
5. Když procesor při provádění podprogramu narazí na kód příkazu RTS, proběhne dříve popsany proces obráceně. Obsah ukazatele stacku se zvýší o 1 a ze stacku bude načten LOW-bajt adresy, na kterou se má program vracet.
6. Potom se ukazatel stacku zvýší ještě o 1 a ze stacku se přečte HIGH-bajt adresy. Ukazatel stacku má nyní stejnou hodnotu, jakou měl před voláním podprogramu. Jak již bylo uvedeno, ukazuje adresa, která byla do stacku uložena na HIGH-bajt adresy za příkazem JSR. Na tomto místě pro-

gram přirozeně pokračovat nemůže. K tomu se musí adresa, která již byla ze stacku vytažena, zvýšit ještě o 1 tak, aby ukazovala na další strojový příkaz, který je uložen za příkazem JSR.

7. Nakonec se ještě programový čítač nastaví na adresu, která byla právě vytažena ze stacku (plus 1!). Teď už může program zcela normálně pokračovat za příkazem JSR.

K tomu ještě přehled, který uvedené ještě jednou ozřejmí. Příkaz JSR v tomto příkladu volá podprogram na adrese \$1000.

Adresa \$3000 \$3001 \$3002 \$3003 \$3004

Obsah Prg. JSR Adr. Adr. Prg.
text kód LOW HIGH text

Progr. ^čítač: \$3000

S T A C K

ukazatel stacku \$XX

Nyní narazí procesor na kód příkazu JSR. Uloží aktuální obsah programového čítače plus 2 do stacku. Celé to vypadá takto:

Adresa \$3000 \$3001 \$3002 \$3003 \$3004

Obsah Prg. JSR Adr. Adr. Prg.
text kód LOW HIGH text

Progr. ^čítač: \$3001

S T A C K

\$03 \$30

ukazatel stacku \$XX-2

Nyní zpracovává procesor podprogram na adrese \$1000, dokud nenarazí na příkaz RTS. Potom si opět vytáhne ze stacku LOW a HIGH-bajt adresy zpětného skoku. Stack pak vypadá takto:

S T A C K

ukazatel stacku \$XX

Jeho stav je tedy shodný s tím, který byl před skokem do podprogramu.

Adresa, která byla ze stacku vytažena, byla adresa \$3003. K této adrese přidá procesor ještě hodnotu jedna. Konečná platná adresa pak je \$3004. Přesně na této adrese procesor pokračuje ve zpracování hlavního programu:

Adresa \$3000 \$3001 \$3002 \$3003 \$3004

Obsah Prg. JSR Adr. Adr. Prg.
text kód LOW HIGH text

Progr. ^čítač: \$3004

Aby byl program vytvářený program co nejpružnější, musíte se snažit rozložit jej na co nejvíce částí – podprogramů. Tyto pak je vhodné tvořit tak, aby je bylo možno volat s libovolnými parametry. Pokud chcete například napsat rutinu pro invertování určité části obrazovky, natahujte adresy, které k tomu potřebujete ne jako pevné hodnoty, zabudované do programu, ale určete podprogramu, na kterých adresách má potřebné hodnoty hledat. Tak je později možné bez velkých komplikací invertovat i jiné části obrazovky.

Takový způsob programování se nazývá strukturované programování. To je základní idea vyšších programovacích jazyků jako je C nebo pascal, nikoliv však BASIC. V konečném efektu vede strukturované programování k tomu, že váš program vypadá asi takto:

Podprogram 1

Podprogram 2

Podprogram 3

(JK)

PŘEČÍSLOVÁNÍ V BASICU

Následující řádek se sice nedá srovnávat s profesionálním programem na přečíslování řádků, nicméně funguje.

```
1 FORA=2049 TO
PEEK(45)+PEEK(46)*256-3:
POKEA+2,Z:
POKEA+3,0:
A=PEEK(A+1)*256-1:
Z=Z+1:
NEXT
10:
20 REM RENUMBER
```

Rutina přečísluje čísla řádků programu v Basicu. Program smí být max. 255 řádků dlouhý. Prvý řádek opatří číslem 0 další v odstupu 1. Skokové adresy ovšem nedokáže změnit, ty se musí opravit dodatečně ručně (GOTO/GOSUB). Při opisování rutiny použijte zkratky příkazů (viz. manuál C64). Tím se vejde celý program do jednoho řádku.

Aby se program dal přečíslovat, musí se nejprve vědět, na kterém místě v paměti se nachází. Každý basicovský program obsazuje u C64 v normálním stavu paměť 2048 až $PEEK(45)+256*PEEK(46)-3$. $PEEK(2049)+256*PEEK(2050)$ udává, na které adrese první řádek končí. Hodnoty na adresách 2051 a 2052, určují první číslo řádku. ($PEEK(1051)+256*PEEK(2052)$) Druhé číslo řádku se najde v paměti na adresách $PEEK(2049)+PEEK(2050)*256+2$ a $PEEK(2049)+PEEK(2050)+3$.

Zpátky k programu pro přečíslování. Čísla řádků se nacházejí právě na adresách A+2 (Low-bajt) a A+3 (High-bajt). Z úsporných důvodů poukne na místo A+2 hodnotu z ($z=0,1,2,\dots,n-1$; n představuje počet řádků Basicovského programu, který má být přečíslován, včetně jednořádkové rutiny) a na místo A+3 hodnotu 0. Program v Basicu může mít maximálně 255 řádků (max. hranice pro 1 bajt). Jinak bychom museli rutinu rozepsat do dvou řádků:

```
1 FOR A=2049 TO PEEK(45)+PEEK(46)*256-
3:POKE A+3,z/256:POKE A+2,
z-INT(z/256)*256
2 A=PEEK(A)+PEEK(A+1)*256-1:z=z+1:NEXT
```

AUTOMATICKÝ LOADING A START

Flopyspeedery mají docela užitečnou funkci, která ušetří čas. Automatický loading a start programu vyřídí jedním příkazem.

Ostatní potřebují pro loading a start dva oddělené příkazy:

```
LOAD „program“,8 <RETURN>
RUN <RETURN>
```

Zadání příkazu

```
LOAD „program“,8:RUN <RETURN>
```

moc nepomůže. Všechny další příkazy stojící za příkazem LOAD jsou ignorovány. Přesto řešení existuje. Zadejte nejprve příkaz pro loading aniž byste stiskli RETURN.

```
LOAD „program“,8:
```

Teď přijde vlastní trik. Stiskněte zároveň klávesy <SHIFT> a <RUN/STOP>. Výsledek bude vypadat takto

```
LOAD „program“,8:LOAD
SEARCHING FOR PROGRAM
LOADING
READY.
RUN
```

Program se jedním průchodem natáhne a odstartuje. Ušetřili jste něco málo klapnutí kláves a nemusíte celou dobu natahování sedět u počítače a čekat kdy už...

Zdůvodnění funkce je až překvapivě prosté. Rozhodně při něm nezapadneme do tajů programování strojového kódu. Nesmíte zapomenout, že kombinace kláves <SHIFT> + <RUN/STOP> není nic jiného než postupné stisknutí kláves:

```
<L><O><A><D><RETURN><R><U><N><RETURN>
```

Po prvním RETURN se C64 pokusí provést příkaz: `LOAD„program“,8:LOAD`

První LOAD pochází z našeho ručního zadání, druhý z kombinace kláves <SHIFT> + <RUN/STOP>. Vzpomínáte si ještě? C64 zapomene při provádění příkazu LOAD všechno, co stojí za dvojtečkou. Druhý příkaz LOAD je plně irelevantní, zbude z něj pouze <RETURN>. Po loadingu je C64 připraven pro další

zadávání. Tady najdou své uplatnění další klávesy z textu SHIFT-RUN/STOP.

Příkazy

RUN <RETURN>

nebudou tedy pochopeny jinak než byste je napsali znak po znaku.

RUN NA DVA STISKY KLÁVES

Za předpokladu, že se v paměti počítače nachází nějaký program můžeme provést jeho odstartování s minimální námahou. Nejvhodnější je zřejmě obsazení funkčních kláves cartridgi, ovšem účinné je i stisknutí kláves:

K <SHIFT> + <RUN/STOP>

Obrazovka bude vypadat poněkud chaoticky:

KLOAD

?SYNTAX ERROR

READY.

RUN

Příkaz LOAD bude ignorován, protože ve spojení s klávesou K (podle libosti jinou) nedává smysl, resp. dává syntaktický nesmysl. Existuje jen příkaz LOAD nikoli KLOAD nebo jiný. Po hlášení SYNTAX ERROR je tedy proveden kýžený RUN.

Příručky pro Basic V2.0 nechávají otevřenu spoustu otázek. Ukážeme řešení některých z nich.

PRVÝ ROZMĚR

V commodorském Basicu je u dimensovaných proměnných 11 standardních prvků. Aniž byste dopředu dimenzovali pole A, můžete použít například příkaz PRINT A(10) nebo (A10)=19 a přesto nedostanete hlášení? BAD SUBSCRIPT ERROR. Funguje to proto, že Basic při startu všechna pole, jedno zda numerická či řetězcová, bere s číslem 10 (počítání začíná nulou, dohromady jich tedy může být 10).

Jako kuriozita působí, že po zpracování jednoho z obou příkazů způsobí DIM A(10) ?REDIM'D ARRAY ERROR přesto, že žádné pole dosud dimenzováno nebylo.

NEXT PLEASE !

Příkaz NEXT uzavírající smyčku FOR...NEXT bývá zpravidla opatřen jen jednou proměnnou. Můžete jej však využít i tak, aby uzavíral více proměnných. Pořadí NEXTA: NEXTB: NEXTC se dá jednoduše stáh-

nout do jednoho NEXTA,B,C. Vedle časové úspory je devízou tohoto způsobu zápisu přehlednost.

CO JE CMD?

Zapomeňte na známou americkou firmu. Asi znáte příkaz CMD, který slouží k vysílání dat na periferní přístroje jako třeba tiskárnu. Abyste vypsali program tiskárnou, zadejte jí:

OPEN4,4: CMD4: LIST

Otázka pak stojí: Co znamená zkratka CMD? Vznikla z anglického „CHANGE MAIN DEVICE“, tedy přepnout hlavní přístroj.

Dále, jen dále.

Příkaz CONT může být po přerušení programu klávesou RUN/STOP nebo po příkazu END nebo STOP použit k tomu, aby se program dal spustit od místa přerušení. Co se však stane, narazí-li počítač na CONT během programu. Potom CONT vede na nekonečnou smyčku. Mimochodem, příkazy END a STOP se skutečně liší jen tím, že v případě END se objeví hlášení BREAK IN XXXX. Jinak jsou oba příkazy identické a zaměnitelné.

Jakou číselnou hodnotu má tečka ?No, kolik? Nic! Lépe řečeno 0. Vždy, když se má použít cifra 0, je dobré ji nahradit tečkou. Basic provádí příkazy s tečkou rychleji než s nulou. Jinými slovy:

```
10 POKE 53281,.: POKE 53281,1: GOTO 10
```

běží rychleji než významově stejný řádek:

```
10 POKE 53281,0: POKE 53281,1: GOTO 10
```

DATA řádek

Nuly v DATA řádcích mohou být jednoduše vynechány. Ušetří se tím místo v paměti a prsty při psaní. Namísto:

```
10 DATA 34,0,2,45,0,0,23,0,0,0,2
```

se může také napsat

```
10 DATA 34,,2,45,,,23,,,2
```

Rovněž tak se dají ušetřit uvozovky v „prázdných řetězcích“ v datech. U řetězců je pozoruhodná ještě jedna zvláštnost. Pokud mají v datech nalézt uplatnění znaky vytvářené SHIFTEM, musí být texty vsazeny do uvozevek, sic je interpreter převede do tzv. token.

(JV)

Nákup jednoúčelových titulkovacích zařízení pro videa se musí majitelům C64 nutně jevit jako mrhání finančními prostředky. Jejich miláček totiž zvládne otitulkování videofilmu také. Záleží jen na tom jaká je představa o titulcích.

Naznačená úloha je v zásadě řešitelná dvěma způsoby. Prvý nevyžaduje žádné zvláštní zařízení. Postačuje základní konfigurace počítače tj. C64, disketová jednotka či dataset a vhodný program. Titulky se vytvoří do grafických obrazovek. Na videopásku se při záznamu nechá vhodně dlouhá mezera a titulky se do ní vloží.

Druhý, profesionální přístup, předpokládá nasazení digitálního genlocku, který umožňuje titulkování do obrazu.

Začněme však od začátku, od toho nejjednoduššího. První, důležitá otázka, která každého okamžitě napadne zní: A jak vlastně dostanu titulky na pásek? Nejprve musí být C64 připojena k videorekordéru. Principiálně lze spojení obou přístrojů provést dvěma způsoby. Přes anténní zdířku (HF cestou) nebo z videovýstupu počítače.

K propojení přes anténní výstup jen pár poznámek. Koaxiální kabel přijde připojit do videa na místo anténního vstupu. Potom zvolte na videorekordéru kanál 36. Signál z C64 můžete zaznamenat jako každý jiný televizní signál.

Tento postup má pár nevýhod. Kvalita obrazu je poněkud horší než při použití videosignálu. Také ozvučení ze stereozařízení či magnetofonu je komplikované, neboť videorekordér nemá možnost současného nebo následného ozvučení. Audiosignálem musíte totiž napájet pin 5 audio/video zdířky C64. Pro tuto aplikaci neexistuje žádný hotový kabel, musíte si jej tedy zhotovit sami. Důležité je, že mimo piny 5 a 2 nejsou obsazeny žádné další.

Přejdeme ke spojení přes audio/video vstupy. Problematika je komplikovanější o to, že existuje celá řada rozdílných norem pro konektory. Většina videorekordérů je opatřena oddělenými koaxiálními vstupy cinch nebo BNC pro video (jeden) a audio (dva) signál. K propojení použijte kabel na jehož jednom konci je pětikolíkový konektor a na druhém konci dva konektory cinch (BNC), z nichž jeden vede video a druhý audio. Konektor pro videosignál bývá červené barvy. Propojte jej se zdířkou VIDEO IN na videorekordéru. Na audio vstup videorekordéru můžete přivést signál ze stereozařízení nebo magnetofonu a podmalovat titulky hudbou. Konektor cinch na kabelu k monitoru zůstane nevyužit.

Kabely k monitorům 1084S zásadně nepoužívejte. Nevedou totiž kombinovaný videosignál, nýbrž oddělené signály pro jas a barvy.

Pokud váš videorekordér nemá oddělené audio a videovstupy, musíte použít kabel s dvacetikolíkovým Scartem nebo šestikolíkovým DIN, podle toho jakými vstupy je váš přístroj osazen. Tyto kabely mají však nevýhodu v nemožnosti současného ozvučení titulků. Pak nezbude než kabel si zhotovit (nechat zhotovit) sami.

K záznamu samotnému není potřeba dodávat snad nic. Ponechávám prostor plně vaší fantazii.

Pro tvorbu vlastních titulků lze využít nejrůznějších programů. Do skupiny programů pro vytváření počítačových dopisů, pozvánek a reklam patří:

Omega Writer

Program existuje i ve verzi pro dataset. Má několik módů. Pro vytváření nápisů slouží mód edit. Titulky běží po obrazovce ve formě horizontálně plynoucího textu zprava doleva či obráceně. Rychlost skrolování je možno dvakrát změnit (rychle, pomalu), popřípadě úplně zastavit. Program umožňuje doprovodit plynoucí text i hudbou z počítače.

Super Writer Classic

Na rozdíl od předchozího programu lze při psaní textu měnit znakové sady. Stejně jako u Omega Writer je text doprovázen hudbou. Obrazovku postupně zaplní písmena, která být doplněna různými efekty (text na pozadí hvězdné oblohy ...). Plná obrazovka skroluje nahoru.

Podobných programů existuje celá řada.

Dále je možno použít grafické programy jejichž součástí je funkce TEXT. Takovým programem je ART Studio. Protože je program určen především pro malování na obrazovku, dají se titulky doplnit celou řadou obrázků a drobných grafik. Ke stejnému účelu se dá použít i program Supergrafik 2 (možnost změny velikosti písmen, nemožnost skrolingu).

Zvláště vhodné se jeví použití programů z dílny Scantronik počínaje populárním Printfoxem přes Eddison k Videofoxu.

O Printfoxu bylo již pár odstavců ve FUNu napsáno, zmíním se tedy o programu pro kreslení Eddison. Program vychází z HiEddi a HiEddi+. Vyznačuje se snadností obsluhy. Aktivaci funkcí je možno provést buď z menu na spodním okraji obrazovky nebo z klávesnice. Všechny funkce pro kreslení lze odzkoušet

na nečisto pružným obrazcem sledujúcim koncový bod kurzoru. Týká sa to i textu, ktorý sa dá do obrázků doplnit. Napsaný text môže byť libovolně polohován a nakonec fixován.

- program využívá grafickú paměť 640*400 bodů.
- má všechny standardní funkce dobrého malovačihó programu přímky, obdélníky, kružnice, elipsy, funkce sprej, vymazávání a vyplňování ohraničených ploch.
- kreslení a mazání různě tlustými štětci.
- volně definované sprajty.

na bod přesně definované oblasti mohou být přesunovány, kopírovány, otáčeny, zrcadleny a logicky spojovány. Mohou být i stupňovitě zvětšovány a zmenšovány.

- 20 nejružnějších vzorků pro vyplnění obrazců.
- funkce TEXT s různými možnostmi (tlusté, široké, vysoké písmo). Měnitelné znakové sady.
- funkce ZOOM pro přesné editování
- zobrazení souřadnic v bodech či milimetrech. Volně definovatelný počátek souřadnic.
- výstup na tiskárny kompatibilní Epson a Commodore jak 9 tak i 24 jehličkové.

(pokračování)
(JV)

PŘEDSTAVUJEME . . .

TEXTOVÝ EDITOR ČSTEXT

Najčastejšie využitie osobného počítača po hrách je pravdepodobne na spracovanie textu. Oproti lacným písacím strojom má spracovanie textu pomocou počítača mnohé výhody. Jednoduché opravovanie chýb, možnosť kedykoľvek texty opäť vytlačiť či archivovať. Commodore 64 má potrebný výkon na spracovanie textov, pokiaľ nemáme zvýšené nároky na rozsah, či špeciálne funkcie. Predstavovaný textový editor umožní aj užívateľom C64 plnohodnotne pracovať s textom, využijú svoj počítač pri práci.

DIAKRITIKA

Textový editor ČStext disponuje úplnou českou, aj slovenskou diakritikou na úrovni vstupu aj výstupu. Textové okno na obrazovke je riešené graficky, nakoľko zabudovaný textový mód C64 neumožňuje vytvárať väčšie znaky ako 8x8 bodov. Základný znak pri editovaní na obrazovke bude mať rozmer 8x12 bodov, na obrazovke sa dá naraz zobraziť 16 textových riadkov. České aj slovenské znaky môžu existovať súčasne v texte, ako aj na obrazovke, prepína sa iba klávesnica.

KLÁVESNICA

Pre slovenské a české znaky je klávesnica predefinovaná podľa vzoru českého, slovenského písacieho stroja. Počas písania textu je možné ľubovoľne prepínať medzi českou, slovenskou a pôvodnou klá-

vesnicou C64. Tak je možné do textu vpísať všetky existujúce znaky. Spoločným problémom pre českú aj slovenskú klávesnicu zostáva vhodné preznačenie kláves. Obsluha klávesnice je riešená tak, že bude možné inštalovať aj driver pre externú klávesnicu. Pripojením bezkontaktnéj klávesnice z PC AT/XT získate stabilný systém na spracovanie textu. V súčasnej dobe je možné si veľmi lacno obstaráť klávesnicu z počítačov PP06. Počítač nestál za veľa, ale klávesnica je kvalitná, vydrží podstatne viac ako na C64. Klávesnice počítačov PC je možné pekne preznačiť pomocou nálepiek, ktoré sú bežne v predaji. Na pripojenie externých klávesníc bude využitý joystickový port č.1, pretože USER port využíva veľa majiteľov C64 na pripojenie paralelnej tlačiarne.

KAZETA ČI DISKETA?

Všetky vlastnosti editoru môžu využiť iba tí, ktorí vlastní disketovú jednotku. Ovšem pre majiteľov datasetov pripravujeme kazetové TURBO rutiny, ktoré umožnia s editorom dobre pracovať. S kazetou nebude možné nahrávať fonty počas tlače, bude ale možné postupne doťahovať z kazety texty na tlačenie (jedná sa o texty, ktoré sa naraz nezmestia do pamäti). V každom prípade budú môcť editor používať aj tí, ktorí nevlastnia floppy. Malé obmedzenia, ktoré im vzniknú budú skôr stratou časti komfortu, ako podstatnými prekážkami.

OBRAZOVKA

Základná obrazovka editoru je rozdelená na pracovnú plochu textu a dialógový riadok. Pomocou dialógového riadku zadáva užívateľ príkazy pre editor, prípadne obdrží chybové, či iné hlásenia a výsledky. Editovanie textu prebieha bežnými editovacími operáciami, pomocou kurzoru. Znaková sada, ktorá je použitá pre zobrazenie na obrazovke sa dá jednoducho vymeniť za inú. Užívateľ má možnosť pomocou FONT EDITORU pre bodové znaky vytvoriť si vlastné znakové sady pre obrazovku. Písmo, pomocou ktorého sa edituje text na obrazovke má veľkosť 8x12 bodov, aby bolo dobre čitateľné aj na bežných televízoroch, alebo kompozitných farebných monitoroch.

VÝSTUPNÉ ZARIADENIE

Hlavným úmyslom pri tvorbe textového editoru pre C64 je snaha umožniť majiteľom tejto lacnej techniky jej plnohodnotné využitie. Preto sa chceme zamerať hlavne na vývoj driverov pre rôzne zariadenia, plottery, tlačiarne, prípadne aj elektronické písacie

stroje. S tým všetkým bude môcť nový textový editor pracovať. Tvorba driverov pre tlač musí zohľadňovať záujem a potreby zo strany užívateľov. Základné zariadenia, o ktorých si myslíme, že sú najviac rozšírené, budú podporované už pri uvedení editoru, ďalšie budú priebežne doplnené podľa záujmu majiteľov.

ZÁVER

Textový editor, ktorý stručne popisuje tento článok, bude možné si objednať u firmy Comotronic v kaze-tovej i disketovej verzii. Cena editoru bude nízka, aby si mohli editor zakúpiť všetci záujemcovia, ktorí potrebujú na svojom C64 spracovávať text. Cena editoru sa bude mierne odlišovať podľa driveru pre tlač textu. Pre každú tlačiareň či plotter nie je napísanie obslužného programu rovnako náročné. Prípadní záujemcovia o bližšie informácie môžu písať na adresu:

Branislav Billa
Za poštou 9/A
920 01 HLOHOVEC

VEKTOROVÝ FONT EDITOR

Program FONT EDITOR slúži na vytváranie vektorových znakových súborov, ktoré môžu ďalej využívať rôzne programy pre plottre (znakové sady vytvorené v tomto programe používa aj PLOTTBASIC II a editor ĆStext).

ZAKÓDOVANIE ZNAKU

Znak je určený relatívnymi súradnicami svojich vrcholov vzhľadom na ľavý dolný roh zvoleného znakového poľa. Jednotlivé vrcholy sú, alebo nie sú pospájané úsečkami. Na vykreslenie takto definovaného znaku potrebujeme len príkazy MOVE (presun so zdvihnutým perom) a DRAW (presun so spusteným perom), za ktorými nasledujú súradnice bodu. Aby kódovanie bolo úspornejšie, neuvádza sa pred každou dvojicou súradníc príkaz DRAW. Použije sa len vtedy, ak sa začína nová lomená čiara. Za DRAW je teda možné zapísať niekoľko dvojíc súradníc, pričom pero sa bude medzi nimi pohybovať spustené (pospája ich úsečkami), pokiaľ kresliaci program nenarazí v kóde znaku na príkaz MOVE. Ďalším zlepšením je zavedenie príkazu JUMP – skok na ľubovoľný znak sady. Napríklad, ak sa už v sade na-

chádza písmeno „a“, zakódovanie písmena „á“ bude pozostávať z vykreslenia dĺžna a skoku na kód písmena „a“. Kresliaci program najprv vykreslí dĺžňu a po príkaze JUMP pokračuje kreslením znaku „a“. Používaním príkazu skoku sa podstatne znižuje pamäťový rozsah znakovkej sady, aj práca pri jej tvorbe. Koniec definície znaku je určený kódom END.

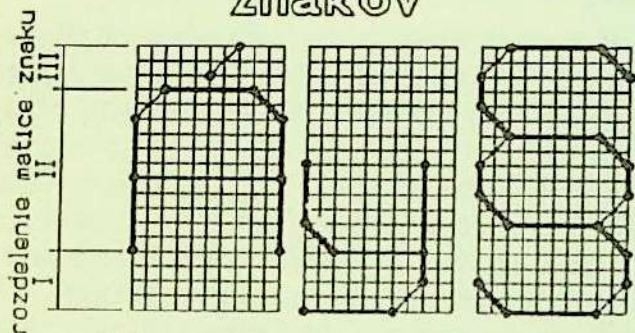
POPIS PROGRAMU

Pracovná obrazovka FONT EDITORu je rozdelená na dve časti, farebne odlišené. Ľavú časť tvorí oblasť 200 x 200 bodov, na ktorej sa zobrazujú znaky. Pravá časť obrazovky slúži na komunikáciu s užívateľom, zadávanie a editovanie znakov. Základné menu poskytuje tieto možnosti:

- | | |
|--------------|-----------------------------------|
| NEW CHAR | - vytváranie nového znaku |
| DELETE CHAR | - vymazanie znaku z pamäti |
| EDIT CHAR | - editovanie už vytvoreného znaku |
| JUMP TO CHAR | - skok na znak |
| LOAD FONT | - nahrať fontu z diskety |
| SAVE FONT | - uložiť fontu na disketu |

- GRID - zapnutie/vypnutie mriežky
- INVERT - inverzia pracovnej plochy
- RESET - inicializácia pracovnej pamäti

Tvorba vektorových znakov



Tabuľka kódov pre FONT EDITOR

kódy	odpovedajúce znaky
0-13	" # \$ % & ' () * + , -
14-27	. / 0 1 2 3 4 5 6 7 8 9 : ;
28-41	< = > ? @ a b c d e f g h i
42-55	j k l m n o p q r s t u v w
56-69	x y z [\] ^ _ ` a b c d e
70-83	f g h i j k l m n o p q r s
84-97	t u v w x y z → ↓ ♂ ♀ € ₤
98-111	đ ē ē i i i ñ ó o o r r š š
112-125	ú ú ú ý z á á č ě ě ě ě ě ě
126-138	ň ó ô õ ř ř š ť ú ů ů ý ž

V hornom riadku je zobrazený číselný kód aktuálneho znaku. Vytvorenie nového znaku je jednoduché. Najprv sa klávesami F1-F7 presunieme na taký číselný kód, ktorý chceme vytváranému symbolu priradiť. Funkcia JUMP TO CHAR slúži na rýchly presun k požadovanému kódu. Po voľbe funkcie NEW CHAR zadávame postupnosť príkazov MOVE, DRAW a JUMP s príslušnými parametrami. Editor je čiastočne samoopravný - nedovoľuje zadanie nesprávnych súradníc (mimo rozsahu) alebo nedefinovaných príkazov. Po každom zadání súradníc sa príslušný príkaz vykoná v grafickej časti obrazovky, znak sa postupne vykresľuje už počas editácie. Zadáním kódu ukončenia znaku alebo prerušením zadávania sa vrátíme do menu. Korekcie a úpravy sa potom môžu vykonať zvolením funkcie EDIT CHAR. Umožňuje mazanie riadkov, vkladanie nových riadkov a zmeny súradníc vrcholov. Pri pohybe kurzora po riadkoch sa analogicky pohybuje grafický kurzor

(biela bodka) po editovanom znaku. Pre lepšiu orientáciu v grafickom poli sa dá funkciou GRID vyvolať mriežka s delením po 10 bodoch. Zmenu súradníc je možné vykonať priamo - zadaním nových číselných hodnôt, alebo v špeciálnom interaktívnom režime. Po zapnutí interaktívneho režimu riadkový kurzor zostane stáť na zvolenom riadku a kurzorovými klávesami sa vykonávajú pohyby vrcholu, na ktorom sa nachádza grafický kurzor. Zmenou súradníc vrcholu sa súčasne mení tvar znaku. Každý pohyb grafického kurzoru je kopírovaný zmenou číselných hodnôt súradníc na príslušnom editovanom riadku. Vytvorenú znakovú sadu je možné zapísať na disketu a kedykoľvek ďalej upravovať, alebo dokončovať. Vektorový znakový súbor je označený príponou „.fnt“.

KOMPATIBILITA ZNAKOV

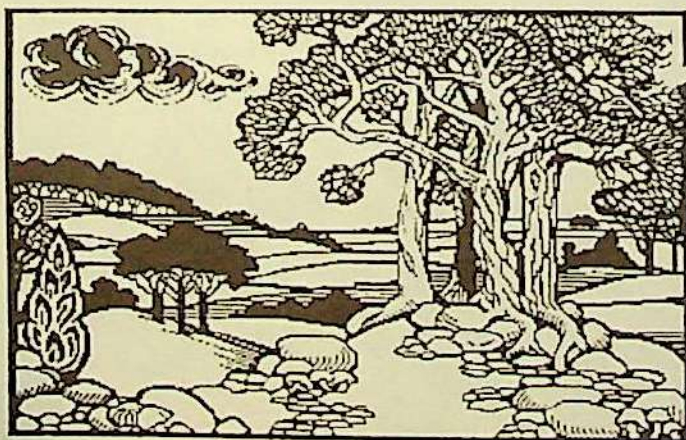
Znakové sady, vytvorené vo FONT EDITORE majú maximálne 139 znakov. Môžu obsahovať všetky slovenské a české znaky, všetky špeciálne CBM znaky. Maximálny pamäťový rozsah na jeden font je 12kB. Relatívne súradnice znaku sa môžu pohybovať v rozsahu 0 až 199 bodov pre obidva smery. Namiesto znakov sa dajú editovať aj rôzne ikony a špeciálne symboly. V programe PLOTTBASIC II sa potom dajú s nimi prevádzať rozsiahle operácie ako so znakmi. Formát kódov vytvorený FONT EDITOROM budú podporovať aj ďalšie programy s česko-slovenskou diakritikou.



Okrem čiarovej, vektorovej grafiky a vektorového textu je niekedy potrebné vytlačiť z počítača aj grafiku typu hardcopy. Pri tomto type grafiky nie je obrázok definovaný vektormi, ale pevnou bitovou maskou. Takýto druh grafiky sa dobre tlačí na tlačiarňach, pretože sa v podstate jedná o farebné body, usporiadané v matici. Pri použití plotteru nie je možné len napodobiť fungovanie tlačiarne, grafiku je treba vykresliť určitým postupom, ktorý bude zohľadňovať druh údajov.

ROZLOŽENIE GRAFIKY NA ČIARY

Obrázok vytvorený v grafickom editore obsahuje okrem vodorovných a zvislých čiar aj nepravidelné útvary a plné plochy. Všetky útvary grafiky je potrebné pre plotter rozložiť na úsečky, v ktorých kreslení má plotter pred tlačiarňou určité prednosti. Ak zohľadníme šírku použitého pera, podarí sa nám týmto spôsobom vykresliť pekné plné plochy, ktoré budú mať väčšiu sýtosť ako pri použití ihličkovej tlačiarne. V niektorých obrázkoch prevládajú čiarly vodorovné, v iných zase zvislé. Program sa musí rozhodnúť, aký typ čiar v obrázku prevláda, vyhľadávať ich a postupne kresliť. Inteligentný algoritmus pre vyhľadávanie čiar a farieb obsahujú všetky ďalej predstavované programy pre tlač hardcopy.



Obr. 2

FORMÁT GRAFIKY

Na C64 môže jemná grafika vystupovať v dvoch základných režimoch: Hires (320 x 200) a Multicolor (160 x 200). V režime Hires platíme za väčšie rozlíšenie nutnosťou použiť v jednom atribúte len dve farby. Medzi týmito dvomi módmi nie je možné grafiku prevádzať bez straty kvality a informácie. Rôzne grafické editory podporujú jeden z dvoch grafických módov, majú rôzne prevedenie a formát zaznamenávaných dát. Niektoré pracujú s kazetou, iné len s disketovou jednotkou. Najznámejšie editory a ich tvar výstupnej grafiky sme podporili pre plottery sadou komfortných driverov. Z ostatných editorov, ktoré pracujú so štandardným rozmerom grafiky je možné obrázky vždy previesť do základných formátov pomocou programov na prevod. Jednoduché programy pre prevod formátov sú k dispozícii, alebo si ich niektorí užívatelia napíšu aj sami. Za základný formát grafiky sme zvolili tvar, ktorým nahráva program ART STUDIO pre Hires a program Koala Painter pre grafiku Multicolor. K dispozícii je ešte driver pre editor Vidcom 64 a špeciálny program pre tlač grafiky z výborného editoru Star Painter.

ART STUDIO – HARDCOPY V5.0

Program Hardcopy V5.0 tvorí spojovací článok medzi ART STUDIOM a plotterom. Okrem rutín potrebných pre tlač grafiky je vybavený aj ovládačom joysticku a TURBO rutínami pre ukladanie grafiky na kazetu. Majitelia datasetov túto možnosť určite ocenia, pretože samotné ART STUDIO poskytuje pre archiváciu len pomalý záznam. Jeden obrázok sa takto natiahne aj na 50 otáčok kazety. Program HARDCOPY V5.0 pred nahratím grafiky v TURBE vykoná aj kompresiu dát, takže obrázok je za niekoľko sekúnd na páske. Pri opätovnom kreslení je okamžite k dispozícii. Ovládanie programu je realizované komfortne, pohyblivou šípkou, ktorá sa riadi joystickom v porte 2. Program kreslí inteligentnou rutínou, ktorá vyhľadáva čiarly aj farby, berie do úvahy aj hrúbku použitého kresliaceho pera. Obrázky je možné ľubovoľne zväčšovať, od veľkosti poštovej známky až po veľkosť, ktorá vyplní plochu formátu A4. Na kvalitu kresby má vplyv kvalita použitého pera a papiera. Tušové kresby vychádzajú kvalitne na kriedovom papieri, pri možnosti výmeny fa-

rebných pier sa dá kresliť farebné hardcopy, napríklad vytlačíť obrázkov z hry.

VIDCOM 64 – HARDCOPY V6.0

Je riešený podobne ako V5.0, má tiež zabudované rutiny pre obsluhu kazety, joysticku. Poskytuje, podobne ako bázový editor VIDCOM, možnosť kreslenia vo všetkých typoch grafiky. Kreslenie je možné aj vo farbách, rôznou hrúbkou pera s fubovoľným zväčšením.

KOALA PAINTER – HARDCOPY V7.0

Veľmi rozšíreným a používaným editorom pre grafiku MULTICOLOR je editor Koala Painter. S programom HARDCOPY V7.0 môžete vytlačíť farebnú MULTICOLOR grafiku v jeho formáte. Tento formát podporujú aj niektoré moduly pre vyberanie grafiky z programov, napríklad HARDCOPY MODUL II. Driver V7.0 má ovládanie riešené šípkou a joystickom, príkazy sú zoradené v oknách podobne ako u predchádzajúcich hardcopy programov. Spolupracuje iba s disketovou jednotkou, podobne ako Koala Painter. V obrázkoch môžete zmeniť farby, otáčať ich, zväčšovať. Program má zabudované automatické vysunutie papiera a manuálne ovládanie písacej hlavy pomocou joysticku.

STAR PAINTER – HARDCOPY V8.0

Veľmi vydareným editorom je Star Painter. Má zabudovanú veľkú kresliacu plochu a mnohé užitočné funkcie, ktoré ho predurčujú na spracovanie schém, grafov, štítkov a rôznych ďalších aplikácií. Program V8.0 umožňuje tlačíť v ňom vytvorenú grafiku na plotteroch. Pred tlačením sa dajú v programe vykonať základné operácie s grafikou, zväčšenie, otáčanie, inverzia. Čo je dôležité, program podporuje až 12 rýchlostí kreslenia plotteru. Pri kreslení rôznymi typmi pera je potrebné pre dosiahnutie maximálnej kvality správne určiť rýchlosť tlače. Napríklad pri kreslení tušom nesmie byť rýchlosť príliš veľká, aby pero stačilo kresliť a aby nedošlo k jeho poškodeniu. Program má ovládanie riešené pomocou ikon a pohyblivej šípky.

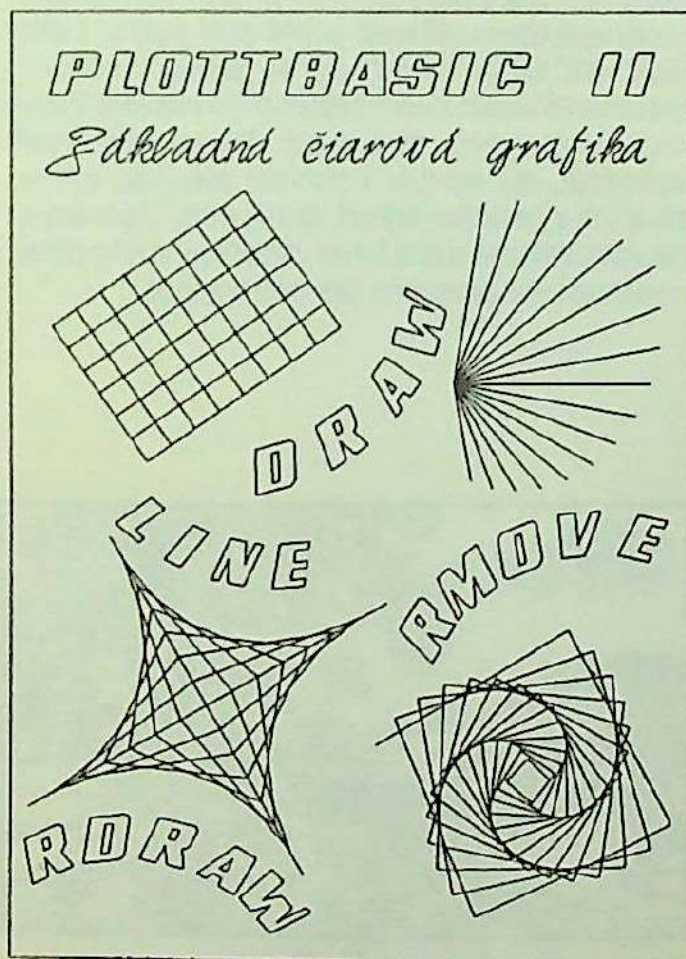
PREVOD GRAFIKY ZO ZX SPECTRUM

Program ART STUDIO existuje aj vo verzii pre počítače ZX SPECTRUM a kompatibilné, to znamená ZX SPECTRUM+ a DIDAKTIK GAMA. Pre tieto počítače existuje aj veľké množstvo dobrej grafiky v hrách a programoch. Zaujímavou problematikou je pre-

nos grafiky ZX SPECTRUM do počítača C64 a jej následné spracovanie, prípadne vytlačenie.

PROGRAM SCREEN PIRAT V1.0

Pomocou tohto programu môžete previesť grafiku ZX SPECTRUM do C64 bez nutnosti použitia ďalších hardwarových doplnkov. Stačí zasunúť kazetu, na ktorej je nahratý obrázok zo sinclairovského ART STUDIA do datasetu a pomocou príkazov programu SCREEN PIRAT môžete túto grafiku priamo prečítať a nahrať si ju na kazetu alebo disketu vo formáte ART STUDIA. Program podporuje aj záznam na kazetu v TURBE spolu s kompresiou dát, takže zosnímanú grafiku môžete priamo tlačíť programom HARDCOPY V5.0. SCREEN PIRAT umožňuje aj vyberanie obrázkov priamo z hier na ZX SPECTRUM. Má zabudované účinné funkcie na prehľadávanie programov pre SPECTRUM na grafiku. Dajú sa vybrať ako titulné obrázky, tak aj obrázky, „schované“ v programe, alebo ich časti, všetko vo farbách. Obsluha je veľmi komfortná a jednoduchá, program má peknú grafickú úpravu.



Obr. 3

MEGA RAM DISK PRO C64

Americká firma CMD (Creative Micro Devices) uvedla na trh v loňském roce vedle floppydiskové jednotky na diskety 3,5 inch FD 4000 (bylo o ní pojednáno v pátém čísle Funu) a harddisku (20 resp. 85 Mbyte) také modul RAM link. Jedná se o RAM disk. Jeho kapacita ho řadí do špičky periferních přístrojů pro expanzní port C64.

Hovořit v tomto případě o modulu snad ani nemá význam, protože se jedná o stabilní plechovou krabici s mírami: 15 cm šíře, 13 cm hloubka a 5 cm výška. Celý přístroj váží ca 1 kg, proto je uložen na gumových nožkách, takže expanzní port nemusí nést plné zatížení. Dodávka je doplněna napájecím zdrojem a obsáhlým manuálem v angličtině. Do přístroje je vestavěn floppy speeder „Jiffy DOS“.

Základní deska nese sloty pro zasunutí paměťových modulů SIM RAM o kapacitě 1 nebo 4 Mbyte. Maximální kapacita paměti může dosáhnout neuvěřitelných 16 Mbyte. Jako příslušenství je možno k mo-

dulu objednat i akumulátor, který jednou uložená data podrží i při vypnutém zdroji několik hodin (např. při transportu přístroje).

Instalace RAM link je jednoduchá. Pokud vlastníte další rozšíření paměti pro C64 (Commodore 1750, 1764 nebo Georam) užijete i ta, neboť RAM port na horní straně modulu umožní jejich zasunutí. Vedle se nachází průchozí expanzní port, není proto nutno zbavit se ani dalších rozšiřujících modulů.

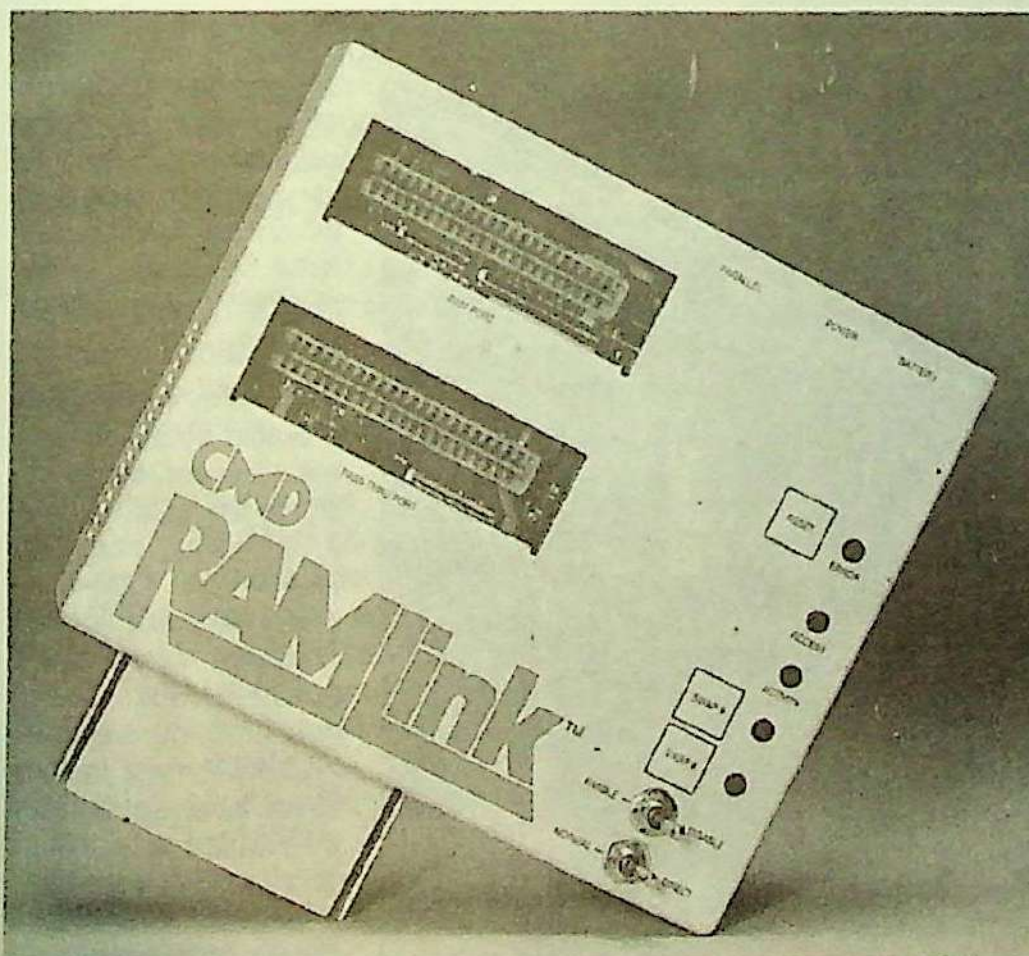
Na zadní stěně jsou přípoje pro napájecí zdroj, aku a mohutný 20 kolíkový konektor pro paralelní připojení harddisku. Mimochodem, disketa obsahuje dva podpůrné programy pro vzájemnou komunikaci (RAM link, HD s C64, resp. C128). Rychlost přenosu dat činí v módu C64 51 kbyte/s a v módu C128 dokonce 100 kbyte/s. Jinými slovy C64 natáhne 200 bloků v necelé sekundě!

Tento paralelní mód se dá zapnout a vypnout dvěma příkazy, takže ani programy se sériovým přenosem dat nečiní žádné problémy.

RAM link toho dokáže mnohem více i bez harddisku. V přístroji se nachází kontaktní lišta pro zasunutí

paměťových modulů. Celková kapacita 16 Mbyte odpovídá ca 100 stranám disket. Díky zmíněným SIM modulům, které jsou běžně známé ve světě PC, představuje rozšíření paměti úkon omezující se na pouhé zasunutí těchto modulů do připravených soklů. Použitelné jsou jak 1 tak 4 Mbytové moduly. Rozšiřování se může provádět v jedno Mbytových krocích až do kapacity 4 MByte a potom dále po 4 Mbytech až do hranice 16 Mbyte. K plné spokojenosti nám při testování vyhověla základní verze s 1 Mbytem.

K čemu vůbec takovou kapacitu paměti využít? RAM link představuje alternativní mohutnou floppyjednotku. Ba, ještě lépe podobně jako harddisk se dá jeho paměť rozdělit do



částí, čímž prakticky vytváří v RAM několik floppy. Modul vedle tzv. „nativního módu“, v němž se využívá plná kapacita jako jedna floppy (!), umožňuje emulovat známé commodorácké jednotky 1541, 1571 a 1581. Dělá to tak šikovně, že fungují různé druhy datových souborů (PRG, SEQ, REL i USR) i příkazy B-R nebo M-R. RAM floppy je vybavena přesně jako originál BAM, directory, stopami a sektory. Stejně tak se dá i používat.

Pouze v jednom bodě se RAM link neshoduje v dosažených hodnotách se svým vzorem floppydiskem. Časy natahování programů se podstatně zkrátily. Doba loadingu harddisku s paralelním kabelem se jeví vedle RAM link jako koňské spřežení vedle formule 1. Při prvním testu loadingu 200 blokového programu jsme se domnívali, že zařízení je poroucháno, neboť v zápětí po zadání příkazu se rozsvítilo hlášení READY. Čas v sekundách označovaný jako velmi rychlý se tím ještě snížil na zlomky sekund. Potěšitelná je i přítomnost speederu „Jiffy-DOS“ vybaveného příkazy pro listing directory bez ztráty programu, kopírovací funkcí a obsazením funkčních kláves.

Průvodní dokumentace je psána lehce srozumitelnou angličtinou.

V souhrnu je RAM link supermodul pro C64 a C128. Zaujme nejen majitele harddisku, jejichž přístroj urychlí podstatným způsobem, ale i ostatní, neboť představuje alternativní řešení potřeby velké kapacity paměti. Celkový dojem z přístroje RAM link je velmi příznivý.

Důležité informace zkratkou:

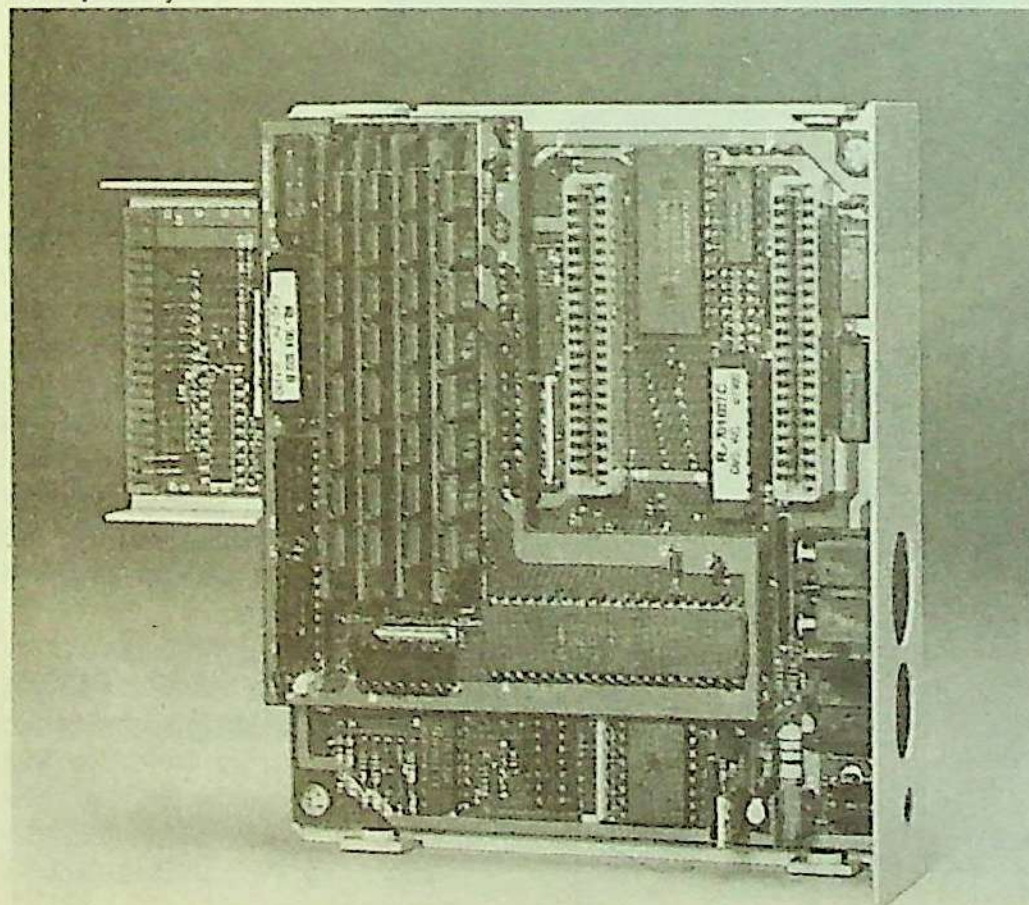
RAM link je modul pro C64 a 128, jímž se připojuje harddisk stejného výrobce paralelním kabelem. To umožňuje podstatně kratší doby loadingu. Dále RAM link tvoří paměťové rozšíření až na gigantických 16 Mbyte. Paměť se dá rozdělit na menší části využívané jako RAM disk s ultrakrátkou dobou přístupu. V modulu je obsažen flopspeeder „Jiffy-DOS, který pro ovládání přináší komfortní příkazy.

- harddisk je s RAM link podstatně rychlejší
- průchozí expanzní port
- rozšiřitelný na 16Mbyte
- použití předchozích rozšiřujících pamětí je zachováno
- provoz možný i bez harddisku
- obsahuje RAM floppy
- emuluje běžné floppydiskové jednotky
- extrémně krátká doba přístupu
- velmi dobrá příručka
- výborné zpracování (vnější vzhled)
- zálohování baterií možné
- vestavěný Jiffy-DOS
- přepínatelná adresa přístroje
- vlastní napáječ

HARD DRIVE HD-40

Hard Disk, bez kterého dnes nemůže žádné PC existovat, je v kategorii 8.bitových počítačů spíše zvláštností. Svědčí o tom i skutečnost, že naši katalogovou nabídku tohoto přístroje využilo opravdu málo zákazníků. Nicméně jde o přístroj, který o třídu zvyšuje hodnotu systému C64 a tak neuškodí, když si o něm řekneme více.

Hard Disk k C64 nabízí prakticky pouze jedna firma, Creative Micro Designs, Inc. (CMD) z USA. Z její dílny pocházejí i RAM-Link, RAM-Drive a Jiffy DOS, dále pak 3.5" disketové mechaniky FD 2000 a FD4000. Všechny tyto produkty tvoří promyšlený komplet, bez problémů spolu spolupracují a navzájem se doplňují. Jejich vzájemné spolupráci můžeme věnovat další článek, bude-li zájem, ale



nyní se věnujeme jen nejdokonalejšímu produktu CMD, Hard Disku HD 40.

V dnešní době miniaturizace nás asi nijak neudiví sebevíce miniaturizovaný přístroj, takže HD 40 se svými rozměry 13 x 8 x 25 cm není žádný drobeček. Robustní plechový kryt spolu s 3.5" mechanikou a elektronikou, zabezpečující kompatibilitu skoro ke všemu, váží 3.5kg, což je v porovnání se samotným C64 pořádná váha. Díky tomu však snese disk při troše štěstí i pád se stolu, aniž byste jej museli odep-
psat.

Indikační a ovládací prvky na panelu HD

Přední ovládací panel přístroje je vybaven membránovými tlačítky s mikrospínači, což je technika dnes u přístrojů zcela běžná. Výhodu má v tom, že přispívá k přehlednosti ovládacího panelu. Kromě toho obsahuje i řadu indikačních diod LED:

- POWER – dioda indikující zapnutí HD
- ACTIVITY – horní dioda indikuje připravenost disku k přístupu
 - dolní dioda svítí v okamžiku přístupu na disk
- ERROR – indikuje systémovou chybu, lze nejn-
snadněji odstranit RESETEm.
- GEOS – indikuje práci v OS GEOS
- SWAP 8 – k nastavení HD jako periferie 8
- SWAP 9 – k nastavení HD jako periferie 9
- WRITE PROTECT – indikuje SW nastavenou ochranu proti zápisu
- RESET – HW reset disku

Konektory na zadním panelu přístroje

Konektory umožňují propojení přístroje se systémem Commodore 64/128, Amiga i PC. K tomu slouží vybavení seriovým rozhraním IEE 448 i SCSI.

- SERIAL – k propojení se sériovým vstupem počítače C64/128 nebo další sériovou periferií
- AUXILIARY – nevyužito, rezerva
- PARALLEL – paralelní linka k připojení RAM Link modulu
- POWER – k připojení síťového napájecího zdroje
- SCSI – k propojení s dalšími SCSI periferiemi (dalším HD)

Obsah HD

Když si HD40–85 koupíte, není zcela prázdný, ale

obsahuje řadu užitečných programů.

Je na něm naformátováno cca 20 adresářů (disket) formátu 1541, 1571 a 1581 a do některých z nich jsou nahrány programové balíky:

1. 1541 HD Utilities pro práci počítače C64/128 s HD,
2. 1541 prázdné
3. 1571 Loadstar 128 – počítačový časopis s řadou utilit i her
4. 1541 Loadstar 64 – počítačový časopis, verze C64
5. 1581 formát CP/M
6. 1541 komunikační program Q-link pro spojení přes modem (RS232)
7. Native prázdné

Práce s HD není tak komplikovaná, jak se na první pohled zdá. Pro přehlednost asi každý zvolí dělení adresáře HD na jednotlivé diskety. Máte-li nějaké celodiskové firemní programy, typu PRINTFOX, EDISSON nebo podobně, lze je snadno pomocí dodaných utilit překopírovat do zvolené virtuální diskety na HD a běžně s nimi pracovat. I pro tento účel je HD vybaven panelovým přepínáním na periferii 8(9).

HD je konstruován tak, že jeho motor běží neustále, což ze začátku působí poněkud rušivě, neboť jsme zvyklí na to, že motor disketové mechaniky se po určité době sám zastaví. Ale jde jen o zvyk, motor je tak tichý, že si na jeho zvuk po chvíli zvykneme a přestaneme jej vnímat.

Pro běžnou práci s HD pak vystačíme s dodanými utilitami a s několika příkazy, které umožňují přístup k datům na HD (v našem případě s nastavením periferie č.12).

Otevření požadovaného čísla diskety (Partition)

```
OPEN15,12,15:PRINT#15,"CPčíslo diskety":CLOSE15
```

Načtení adresáře z nastaveného čísla diskety

```
LOAD"$",12
```

Načtení souboru z HD do počítače

```
LOAD"jméno souboru",12
```

Z uvedeného vyplývá, že HD je i pro C64 užitečnou pomůckou, jejíž komfort oceníme až při jejím používání. Je sice v současnosti pro nás dost drahou záležitostí, ale i u nás již má své příznivce a uživatele. Jistě především pro své dobré vlastnosti.

(JV)

DOPISY ČTENÁŘŮ

HODINY

Program HODINY vyobrazuje digitálne čas v pravom hornom rohu obrazovky nezávisle od ďalšieho programovania v Basicu. Je možné nastaviť aj budík. Hodiny po nahratí príkazom Load „Hodiny“, 8,1 z diskety spustíme príkazom SYS 49152,H,M,S. Za H dosadíme hodiny, M minuty, S sekundy. Budík nastavujeme príkazom SYS 49569,H,M,S. Keď sa ozve, vypneme ho mezeríkom. Celý program sa vypína RUNSTOP+RESTORE.

PROGRAM : HODINY

```

C000 : 20 FD AE 20 9E B7 E0 18 2A
C008 : B0 1A 8E EB C1 20 FD AE 59
C010 : 20 9E B7 E0 3C B0 0D 8E 24
C018 : EC C1 20 FD AE 20 9E B7 82
C020 : E0 3C 90 03 4C 4B B2 8E 92
C028 : ED C1 A9 00 20 D1 C1 78 E9
C030 : A9 3D A0 C0 8D 14 03 8C 57
C038 : 15 03 58 60 EA AD 08 DC E7
C040 : C9 09 F0 08 A9 00 8D EE 7A
C048 : C1 4C CA C0 AD EE C1 C9 E7
C050 : 01 F0 F6 20 59 C0 4C 8F 77
C058 : C0 AD ED C1 C9 3B F0 04 E5
C060 : EE ED C1 60 AD EC C1 C9 9E
C068 : 3B F0 09 EE EC C1 A9 00 BF
C070 : 8D ED C1 60 AD EB C1 C9 45
C078 : 17 F0 0C EE EB C1 A9 00 5C
C080 : 8D EC C1 8D ED C1 60 A9 67
C088 : 00 8D EB C1 4C 80 C0 A9 A1
C090 : 01 8D EE C1 AD EB C1 20 CD
C098 : F9 C0 8D EF C1 8C F0 C1 1A
COA0 : AD EC C1 20 F9 C0 8D F1 F7
COAB : C1 BC F2 C1 AD ED C1 20 36
COB0 : F9 C0 8D F3 C1 8C F4 C1 C3
COB8 : A0 00 B9 EF C1 69 B0 99 22
COC0 : EF C1 C0 05 F0 04 C8 4C 4B
COC8 : BA C0 AD EF C1 AC F0 C1 14
COD0 : 8D 20 04 8C 21 04 A9 BA 4E
COD8 : 8D 22 04 8D 25 04 AD F1 36
COE0 : C1 AC F2 C1 8D 23 04 8C 07
COE8 : 24 04 AD F3 C1 AC F4 C1 D1
COF0 : 8D 26 04 8C 27 04 4C 97 16
COF8 : C1 8D F5 C1 A0 00 E9 09 F9
C100 : C9 3C B0 04 C8 4C FE C0 00
C108 : 8C F6 C1 A9 00 C0 00 F0 9D
C110 : 07 88 EA 69 09 4C 0D C1 EE
C118 : AC F5 C1 8D F5 C1 98 EA B6
C120 : 38 ED F5 C1 AB AD F6 C1 5C
C128 : B0 01 60 18 60 AD F7 C1 4B
C130 : AC F8 C1 AE F9 C1 CD EB 5B
C138 : C1 F0 01 60 CC EC C1 F0 DB
C140 : 01 60 EC ED C1 F0 01 60 D2
C148 : A0 00 98 99 00 D4 C0 18 1B
    
```

```

C150 : F0 04 C8 4C 4B C1 A9 0F 85
C158 : 8D 18 D4 8D 03 D4 8D 02 E9
C160 : D4 A9 00 8D 05 D4 A9 F0 3A
C168 : 8D 06 D4 A9 21 8D 04 D4 9B
C170 : A2 FF 8E 01 D4 A0 7F 8C 3F
C178 : 00 D4 20 DE C1 EA F0 0C 15
C180 : C0 00 F0 04 8B 4C 77 C1 49
C188 : CA 4C 72 C1 A9 00 8D FA 14
C190 : C1 8D 18 D4 60 EA EA AD 1D
C198 : FA C1 C9 01 F0 3A 4C FB 0F
C1A0 : C1 20 FD AE 20 9E B7 E0 5E
C1A8 : 18 B0 1A 8E F7 C1 20 FD 7B
C1B0 : AE 20 9E B7 E0 3C B0 0D DA
C1B8 : 8E F8 C1 20 FD AE 20 9E 4A
C1C0 : B7 E0 3C 90 03 4C 4B B2 22
C1C8 : 8E F9 C1 A9 01 8D FA C1 E4
C1D0 : 60 8D 08 DC 8D FA C1 60 0D
C1D8 : 20 2D C1 4C FB C1 AD 01 0F
C1E0 : DC C9 EF 60 AD 01 DC BD 1A
C1E8 : 01 04 60 01 05 1F 00 B0 CE
C1F0 : B1 B0 B5 B3 B1 1E 03 01 F7
C1F8 : 05 00 00 A0 00 C0 01 F0 FD
C200 : 03 4C 31 EA A0 01 C0 00 EB
C208 : F0 12 A9 60 8D DB CE 20 AB
C210 : B4 CE A9 00 8D 05 C2 A9 F5
C218 : 58 8D DB CE 4C A0 CE 00 0C
    
```

KLÁVESY F1-F8

Program slúži na oživenie kláves F1-F8. Najprv aktivujeme testovaciu rutinu príkazom SYS 52882. Potom definujeme klávesy. Príkazom SYS 53110 sa aktivuje podprogram na zadefinovanie.

Napríklad chceme, aby kláves F3 vypísal príkaz LIST. Potom napíšeme LIST:.0.3 a potvrdíme. Po stlačení F3 sa nám vždy vypíše príkaz LIST:, ale sa nepreviedie. Ak chceme kláves zadefinovať tak, aby sa príkaz aj previedol, zmeníme 0 na 1 (LIST:.1.3). Číslo 3 môžeme zmeniť na 1 až 8 podľa toho, ktorú klávesu chceme definovať. Refazec môže mať maximálne 9 znakov pred prvou bodkou.

PROGRAM : KLAVESY F1-F8

```

CE50 : C9 01 F0 07 C9 00 F0 05 21
CE58 : 4C FC CF A9 0D A0 01 91 48
CE60 : FB CA BD 00 04 C9 2E D0 19
CE68 : EF 88 CA BA 91 FB C8 C8 4D
CE70 : 20 87 CE B9 00 04 C9 20 C6
CE78 : B0 03 EA 69 40 91 FB C0 94
CE80 : 08 F0 0E C8 4C 73 CE BD 54
CE88 : 80 CE E6 FB E6 FB 88 88 2A
CE90 : 60 60 78 A9 A0 A2 CE BD E9
CE98 : 14 03 8E 15 03 4C B4 CE 77
CEA0 : A0 05 C0 0F F0 06 EE A1 13
CEAB : CE 4C 31 EA A9 00 8D A1 5A
CEB0 : CE 4C 00 CF A0 00 B9 EE 6D
CEB8 : CE 85 FB B9 EF CE 85 FC 04
    
```

```

CECO : A0 00 A9 FF 91 FB AD B5 E6
CECB : CE C9 10 F0 09 EE B5 CE 19
CED0 : EE B5 CE 4C B4 CE A9 00 3E
CED8 : BD B5 CE 58 60 EA C9 C0 04
CEE0 : B0 04 38 4C 9E CF 38 E9 47
CEE8 : B0 4C A0 CF 00 00 44 CE 5F
CEF0 : 39 CE 2E CE 23 CE 18 CE 9C
CEF8 : 0D CE 02 CE F7 CD 00 FF B5
CF00 : A5 CB C9 04 F0 0F C9 05 36
CF08 : F0 10 C9 06 F0 11 C9 03 F8
CF10 : F0 12 4C 31 EA A9 00 4C D7
CF18 : 26 CF A9 02 4C 26 CF A9 59
CF20 : 04 4C 26 CF A9 06 AC BD 66
CF28 : 02 C0 01 F0 07 C0 00 F0 41
CF30 : 05 4C 31 EA 69 00 0A AA 19
CF38 : BD EE CE B5 FB BD EF CE DC
CF40 : 85 FC A2 00 A1 FB C9 FF 0D
CF48 : F0 E7 E6 FB A8 A1 FB E6 BA
CF50 : FB C9 00 F0 05 C8 99 77 3A
CF58 : 02 88 B1 FB 99 77 02 C0 69
CF60 : 00 F0 04 88 4C 5A CF C6 4F
CF68 : FB C6 FB A0 00 B1 FB 69 2A
CF70 : 01 85 C6 4C 31 EA A9 A0 C1
CF78 : A0 00 99 00 04 C0 0D F0 DB
CF80 : 04 C8 4C 7A CF A2 00 A9 E0
CF88 : A0 9D 00 04 20 3E F1 C9 C7
CF90 : 00 F0 F9 C9 0D F0 32 C9 75
CF98 : 14 F0 12 4C BA CF E9 40 85
CFA0 : 9D 00 04 E0 0D F0 DE EE 0C
CFAB : 86 CF 4C 85 CF AE 86 CF 06
CFB0 : E0 00 F0 03 CE 86 CF 4C 26
CFB8 : 85 CF AE 86 CF C9 40 90 0F
CFC0 : DF 4C DE CE E9 B0 4C A0 ED
CFC8 : CF AE 86 CF A0 00 8C 86 D3
CFD0 : CF CA BD 00 04 E9 30 C9 58
CFDB : 09 B0 21 C9 00 F0 1D 0A CB
CFE0 : AB B9 EC CE 85 FB B9 ED 75
CFE8 : CE B5 FC CA BD 00 04 C9 91
CFF0 : 2E D0 09 CA BD 00 04 E9 E2
CFF8 : 30 4C 50 CE 4C 76 CF C0 75

```

Obydva programi (Hodiny, Klavesy) môžu pracovať aj súčasne. Keď máme oba prg. v pamäti, spustíme najprv Hodiny a potom zadáme príkaz POKE 49660,1 a definujeme Klavesy príkazom SYS 53110.

(P.Ševčík, Bratislava)

SPRITE EDITOR

Program SPRITE EDITOR by měl usnadnit práci programátorům začátečníkům, kteří se učí programovat grafiku či spíše používat SPRITY tzv. skřítky (vhodné pro animaci atd.).

Ještě nezbytné informace:

Program pozorně opište do počítače a spusťte příkazem RUN. K ovládní se používá joystick v portu 2. To lze změnit na port 1 v řádce 21 (údaj v závorce

změňte na 56321). Kreslí se do plochy 24x21 bodů podobně jako se vytváří SPRITE. Stiskem Fire se zakreslí jednotlivé body. Pokud se budete chtít opravit, použijte klávesu INST/DEL, a to tak, že najedete joystickem na požadovaný bod a tuto klávesu stisknete. Pokud chcete vymazat celý výtvar stisknete F1. Po vytvoření určitého obrazu stisknete RETURN a vpravo se vám vypíše data potřebná pro definici SPRITU.

```

5 REM * SPRITE EDITOR *10
  DIM A (24),L(64)
11 A0$="CLR 23xDOWN"
12 V=53248: POKEV+21,15: FOR I=1
  TO 4: POKE (2039+I),13: NEXT I
13 FOR N=0 TO 62: POKE 832 + N,255:
  NEXT N14 POKE V+23,5: POKE V+29,3
15 GOSUB 65
16 POKE 53281,7: POKE 53280,12:
  PRINTCHR$(147): POKE646,0
17 PRINT "SPACE
  SHIFU24xSHIFT*SHIFTI"
18 FOR I=1 TO 21: PRINT
  TAB(1) "SHIFTB"TAB(26) "SHIFTB:
  NEXTI
19 PRINT "SPA-
  CESHIFTJ24xSHIFT*SHIFTK"
20 X=1: Y=0
21 GETA$: J=PEEK(56320): C=1
22 IF A$=CHR$(133) THEN 16
23 IF A$=CHR$(20) THEN B=32: C=2
24 IF A$=CHR$(13) THEN POKE Z,W2:
  GOSUB 43: Y=0: X=1: B=W2
25 IF J<>0 THEN A$=" "
26 IF (J AND 1)=0 THEN Y=Y-1
27 IF (J AND 2)=0 THEN Y=Y+1
28 IF (J AND 4)=0 THEN X=X-1
29 IF (J AND 8)=0 THEN X=X+1
30 IF (J AND 16)=0 THEN B=160: C=2
31 IF X>24 THEN X=24
32 IF X<1 THEN X=1

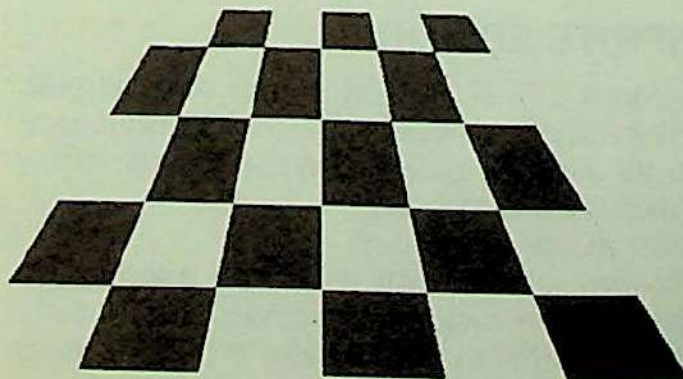
```

```

33 IF Y<0 THEN Y=0
34 IF Y>20 THEN Y=20
35 Z=1105+X+Y*40
36 Q=PEEK(Z)
37 ON C GOTO 39,38
38 POKE Z,B: GOTO 21
39 IF Q=160 THEN W1=32: W2=160:
   GOTO 41
40 IF Q=32 THEN W1=160: W2=32
41 POKE Z,W1: FORI=1 TO 40: NEXT
   I: POKE Z,W2
42 GOTO 21
43 FOR I=1 TO 23: A(I)=0: NEXT I44
   FOR I=1 TO 64: L(I)=0: NEXT I45
   POKE V+21,0
46 X$="„: K=0: PRINT "CLRDOWN„
47 FOR I=0 TO 20: X=0: MAX=256:
   P=0
48 FOR Y=0 TO 23: MAX=MAX/2: IF
   Y=0 OR Y=80 OR Y=16 THEN
   MAX=128
49 IF PEEK(1106 + Y+I*40)=160 THEN
   A(Y)=MAX: GOTO 51
50 A(Y)=0
51 NEXT Y
52 FOR C=0 TO 7: P=P+A(C):NEXTC:
   PRINT X$;TAB(27);P:L(K)=P:P=0:K=K+1
53 FORC=8TO15:P=P+A(C):
   NEXTC:PRINT" SHIFTDOWN„TAB(31)P:L(K)=P:
   P=0:K=K+1
54 FOR C=16 TO23: P=P+A(C): NEXT C
55 IF P>=100 THEN X$=SHIFTDOWN:
   GOTO 57
56 X$="„
57 PRINT"SHIFTDOWN„TAB(35)P:
   L(K)=P:P=0:K=K+1
58 NEXT I59 PRINT A0$"7xRIGHT
   STISKNI RETURN„
60 GET K$: IF K$="„ THEN 60
61 IF K$=CHR$(13) THEN PRINT"CLR„:
   GOSUB 63: GOSUB 68: RETURN
62 GOTO 60
63 FOR I=0 TO 21: PRINT "7xRIGHT1
   2xSPACE„: NEXTI:
   PRINT:"7xRIGHT14x*„
64 POKE V+21,15: RETURN
65 POKE V+0,255: POKE V+2,255:
   POKE V+4,255:POKE V+6,255
66 POKE V+1,196: POKE V+3,157:
   POKE V+7,62
67 RETURN
68 Z=0: FOR I=0 TO 62: Z=Z+L(I):
   NEXTI:IF Z=0 THEN 13
69 FOR N=0 TO 62: POKE 832+N,L(N):
   NEXT N: RETURN

```

(M. Lošák, Křenůvky)



AMERICKÉ PRODUKTY PRO C64

Comotronic nabízí uživatelům C64 z produkce firmy CMD následující výrobky:

DISKETOVÁ JEDNOTKA FD 2000

Disketová jednotka na diskety 3,5 inch, umožňující záznam DD i HD. Maximální kapacita záznamu na **jedné disketě** činí 1,6 Mbyte. Jednotka emuluje mechaniky 1581,1571,1541. Plnou kapacitu diskety je možno rozdělit na části o velikosti 170 nebo 340 kbyte. Přístroj umožňuje i práci pod CP/M a je plně kompatibilní s GEOS. Vedle vlastní mechaniky je součástí dodávky napájecí zdroj, propojovací kabel a software na disketě (Tools, kopírovací programy, Geos...). Podrobný manuál je v anglické verzi.
Cena 8900,- Kč

DISKETOVÁ JEDNOTKA FD4000

Vnější vzhled jednotky je stejný jako u FD2000. Přístroj **umožňuje zápis i na diskety ED s maximální kapacitou 3,2 Mbyte (!)**. Jednotka opět emuluje mechaniky 1581,1571 a 1541. Způsob práce je shodný s jednotkou FD2000.

V dodávce je zahrnuta jednotka, napájecí zdroj, propojovací kabel (sériový), disketa se software, návod v angličtině a 1 kus ED diskety. Další diskety je nutno objednat.

Popis disketové jednotky byl uveřejněn v 5. čísle FUN.

Cena 10 500,- Kč

HARDDISK HD85

Inteligentní harddisk pro sběrnici SCSI doplněný software, která umožňuje jeho práci s C64. Přístroj pracuje ve více módech. Mimo vlastního módu je možno kapacitu HD parciálně rozdělit na virtuální diskety se záznamem typu 1541, 1571 nebo 1581. HD je kompatibilní s GEOS, programy od firmy Scantronik a také CP/M. Přístup k datům je o 30 % rychlejší než u VC1541. Samozřejmostí jsou adresáře se stromovou strukturou, obvyklou u PC. Vestavěné hodiny reálného času zaručí správný časový údaj u důležitých dat.

HD se dodává spolu s napájecím zdrojem, kabelem

pro připojení k C64, softwarovým balíkem a návodem v němčině. Návod k obsluze je přeložen do češtiny.

Cena 20 500,- Kč

RAM LINK 16MBYTE

Přídavná paměť pro C64 standardně osazená 1Mbyte s možností rozšíření až na 16 Mbyte. Přístroj je vysoce kompatibilní. Emuluje flopy 1541, 1571 i 1581. Má zabudovaný speeder Jiffy DOS. Dodávaný software zaručuje i práci s GEOS. Data je možno zálohovat baterií. Návod je v angličtině.

Cena 7900,- Kč

JIFFY DOS

Speeder pro všechny typy C64 a disketových mechanik. Zrychlovací faktor se blíží 14. User i expanzní port zůstává volný, není potřeba žádný paralelní kabel. Vlastní speeder sestává ze dvou pamětí eprom s redukcí patice a přepínačem. Tím se dá vždy přepnout na původní operační systém C64, takže nevzniknou s kompatibilitou žádné problémy.

Program obsazuje funkční klávesy příkazy pro listování v adresáři disket a pro natahování a spouštění programů. Při objednávání nutno uvést typ počítače a disketové jednotky.

Cena 3000,- Kč

DALŠÍ NABÍZENÉ VÝROBKY:

Programová disketa 5

–je zaměřena na programy využívající schopnosti C64, resp. obvodu SID generovat zvuky. Disketa obsahuje programy:

Digisound Ripper

Soundwriter V2.0

Funky-Drummer

Musicmaster

Prvý program, Digisound-Ripper slouží k vyhledávání digitalních hudebních dat v ostatních programech pro C64. Soundwriter je pak špičkový hudební editor pro komponování rozsáhlejších skladeb. Poslední dva programy simulují hudební nástroje, Funky-Drummer bicí a Musicmaster klávesové.

Cena 120,- Kč

Cartridge Maker CM256

CM256 je modul zasouváný do expanzního portu C64. V pamětech eprom, zasunutých do objímek, se dají uchovávat programy (uživatelské i hry). Modul se vyznačuje vysokou rychlostí přenosu dat, kompatibilitou a snadností obsluhy.

CM obsahuje výkonný modulgenerátor jehož úlohou je přestrukturování basicovských dat či strojových programů. Programy se nejprve přetáhnou do paměti C64. Modulgenerátor je automaticky doplní několika kontrolními daty (název, startovací adresa, délka), autostartem a nakonec vyrobí datové soubory, jejichž obsah je možno přímo vypálit do eprom. Pomocí CM se dají zpracovat téměř všechny jednodílné programy psané v basicu či strojovém kódu. Do eprom jsou programy ukládány úsporně. Do čipu s větší kapacitou se může uložit více programů. Možné jsou i alternativy, kdy delší program přesahuje do dalšího čipu. V tomto případě se mo-

dulgenerátor postará o správné uložení programů do jednotlivých čipů.

Software pro řízení CM je umístěno ve zvláštní eprom (je součástí dodávky).

Význačné vlastnosti CM 256 kB:

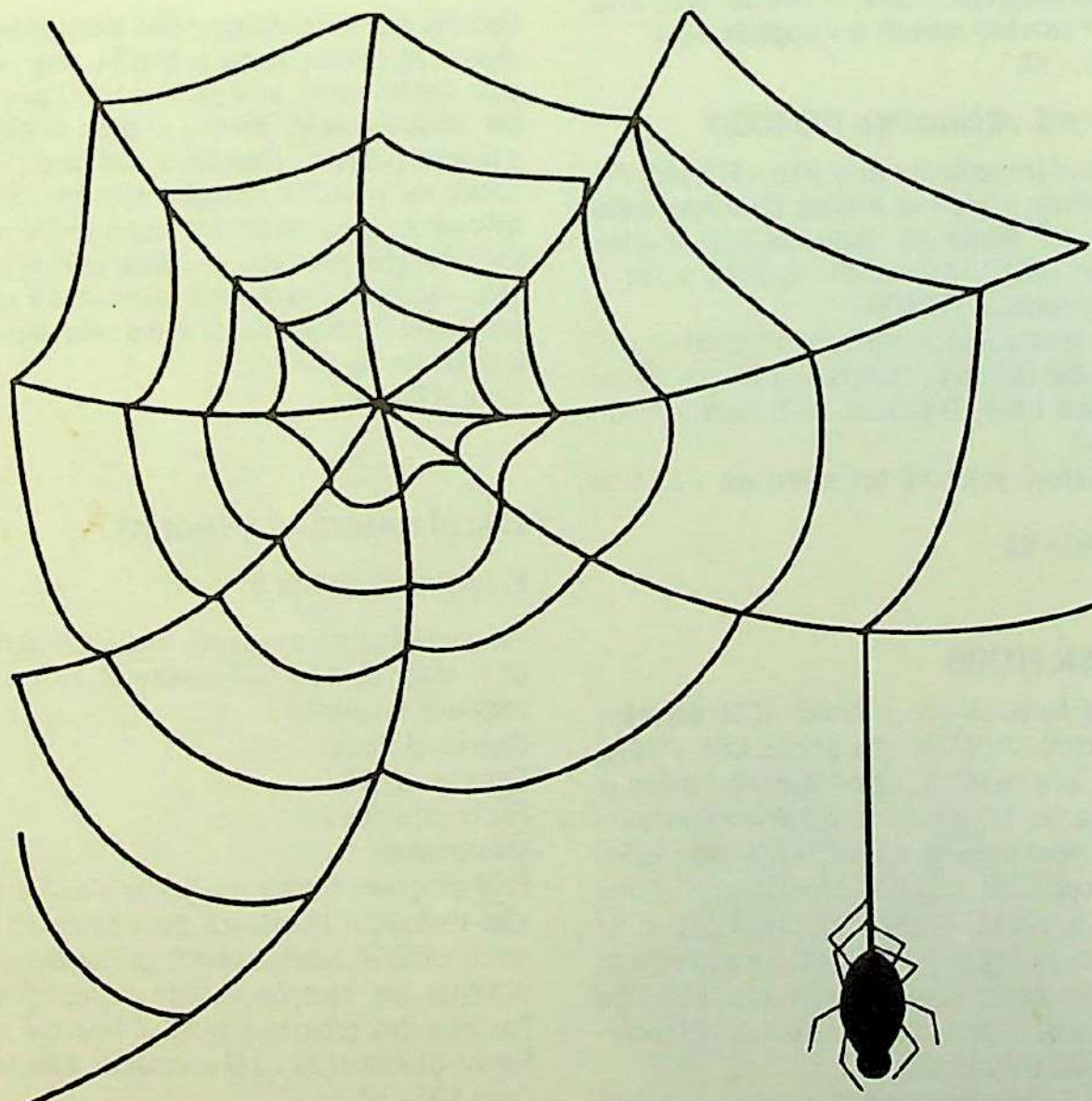
- 4 patice pro eprom typu 2764 až 27512
- celková kapacita programů uložených v eprom činí 256 kB
- automatické rozeznání typu eprom
- vlastní directory programů v eprom abecedně seřazeno
- softwarově odpojitelný

CM je určen pro ty, kteří si chtějí do cartridge zabudovat programy podle vlastního výběru.

Cena 1200,- Kč

Pro majitele CM dále Comotronic nabízí epromer, resp. po předložení diskety s datovými soubory provede vypálení pamětí.

(JV)



Genlock? Genlock!

Genlock míchá různé zdroje je videosignálu z nich synchronizovaný výstupní signál, např. FBAS. Co se na papíře snadno napíše, je ve skutečnosti tvrdý hardwarový oříšek. K řešení se nabízí hned několik problémů. Jedním z nich je rozdílný počet řádků. Televizní obraz sestává z 575 řádků rozdělených do polobrazů po 312,5 řádků. Naproti tomu C64 má na videovýstupu signál, který má jen 624 řádků. K tomu nepřechází paprsek řádky v televizním obrazu postupně v normálním pořadí 1,2,3,4 ..., nýbrž přes řádek, nejprve první půlobraz (řádky 1,3,5,7...) a potom druhý (2, 4, 6, 8...). Tímto trikem se docílí dvojnásobné opakovací frekvence obrazu. Genlock od Scanntroniku synchronizuje oba rozdílné zdroje geniálním trikem. Digitalizuje jednoduše kompletní obraz C64 a ukládá jej do své paměti. Nyní může každému přečtenému bodu ze zdroje videosignálu přiřadit analogový bod obrazu C64 a přenést jej na obrazovku.

Genlockem je možno míchat pět barev ze 184. Transparentní černé přitom přísluší zvláštní úloha. Tam, kde se v obraze z C64 objeví transparentní černá, dojde automaticky k vyjasnění obrazu. Tím je možno realizovat titulky, které se vznášejí nad vlastním obrazem videa. Možné je i vsazení běžícího obrazu do nakreslené předlohy.

Hodnocení genlocku časopisem 64'er

Digitální genlock od Scanntronicku smíchává dva rozdílné zdroje obrazového signálu. Obraz vytvořený počítačem se potom může přeložit přes obraz videa. Vestavěnými přípoji SCART, SVHS a Cinch je umožněno spojení s téměř všemi videorekordéry a kamerami.

Klady:

- řada konektorů pro připojení přístrojů
- všechny výstupy sepnuty současně
- genlock je možno pomocí videoprofi dálkově ovládat
- k zařízení je přiloženo upgrade pro Videofox II
- součástí dodávky je i speciální kabel
- genlock je ovladatelný i bez Videofoxu nebo Videoprofi programem, který je zahrnut v dodávce

Zápory:

- relativně drahý
- opticky nepodařený, přístroj této třídy a schopností by měl mít odpovídající design

Důležitá data

Výrobek: digitální genlock

Cena: 19 500,- Kč

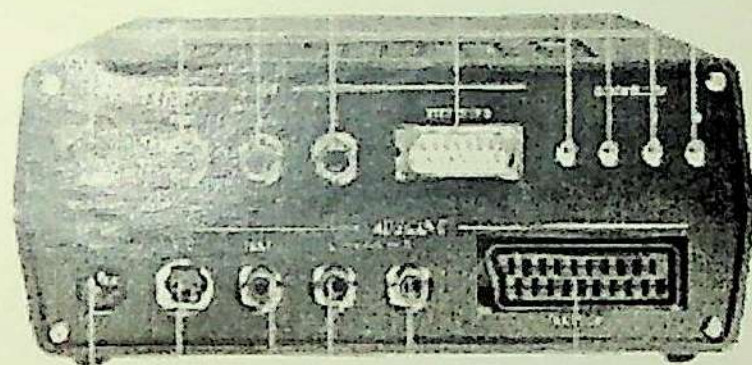
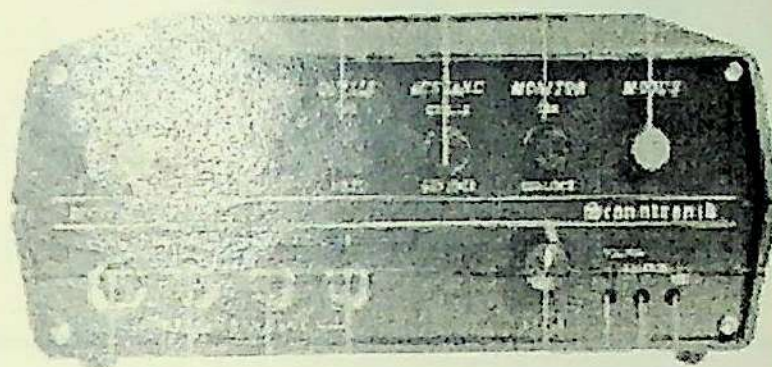
Konfigurace při testování:

C64, VC1541, Dolphin DOS 3.0, videorekordér Toshiba, kamera SONY

Výrobce: Scanntronik Mugrauer GmbH, Parkstrasse 38, 8011 Zorneding-Pöring, tel. 0049 8106 22570

Prodej v ČR:

Comotronic spol. s r.o., Dolnomlýnská 2, 787 01 Šumperk, tel. 0649 42 21/286





Vydává Comotronic klub Šumperk