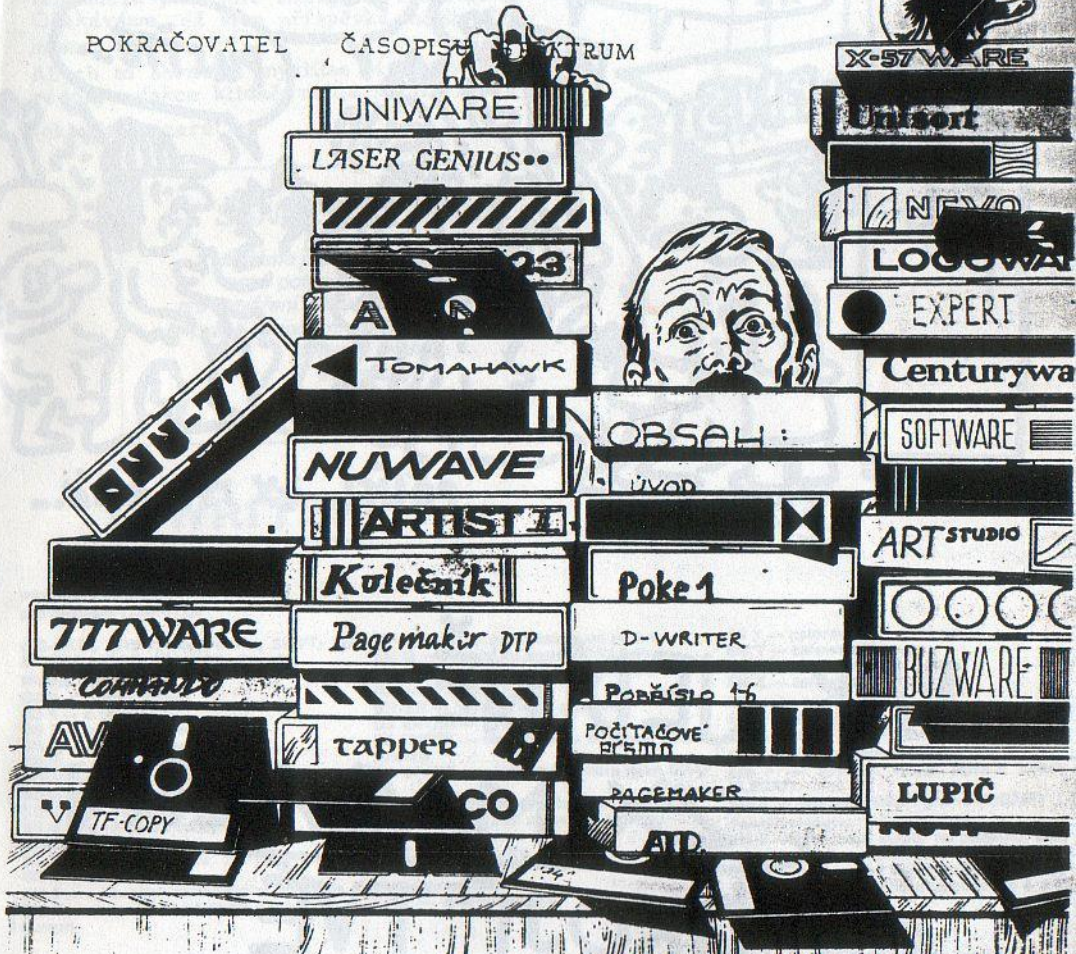


ZX magazin

POKRAČOVATEL ČASOPISU SPECTRUM



STODAK 10K

JSI SPEKTRISTA ?

ANO ?



PAK JEDINĚ
ZK MAGAZÍN
USPOKOJÍ TVÉ
POTŘEBY !!

Vážení čtenáři,

poprvé píšeme na malý formát,
poprvé s důrazem na úpravu.

Proč ?

Dlouhou dobu se někteří členové
redakce hádali, jakým stylem dělat
Spektrum. Buď zhuštěně - více in-
formací, nebo úpravně - větší poži-
tek ze čtení.

Možnost zmenšeného formátu rozhodla.
Při stejném množství údajů je 2x více
prostoru.

Jinak prosíme, zvědavost a ješitost nás
k tomu nutí, napište, jaký dojem ve vás
změna zanechala, zda jde o změnu k
lepšímu, a jaké jiné změny navrhuje-
te. Očekáváme též více příspěvků - máme více
místa.

Abych to shrnul : myslíme si, že by se naše
rádobyredakce klidně mohla přihlásit třeba na

takovýto inzerát :



ŠANCE PRO MLADÉ

Hledáme pracovníky s grafickým
nebo polygrafickým vzděláním
pro vytváření výtvarné podoby časopisů,
práce s moderní výpočetní technikou,
centrum Prahy,
informace
Zn.: tel. 26 38 65."

PP31318410

D-WRITER

Mini
Manual

PŘEHLED PÁIKAZŮ pro program D-WRITER

(CS-CAPS SHIFT; SS-SYMBOL SHIFT; EM-
EXTEND MODE; GRAF-GRAPHICS)

Přesun kursoru:

CS 5 — o znak vlevo
CS 6 — o řádku dolů
CS 7 — o řádku nahoru
CS 8 — o znak vpravo
TRUE — o slovo vlevo
INV.V. — o slovo vpravo
SS Q — na začátek řádku
SS E — na konec řádku
SS U — o 18 řádek vpřed
SS I — o 18 řádek vzad
SS F — na začátek textu
SS G — na konec textu
EM 8 — na začátek následujícího odstavce
EM Y — na začátek odstavce (ve kterém je
kursor)

EM P — na začátek následující stránky
EM G — na značku pozice (GRAF R)
EM K — na značku bloku (GRAF Q)

Rušení a vkládání textu:

ENTER — nová řádka + odstavce
SS ENTER — nová řádka
SS ENTER — nová řádka + odstavce + nová
stránka
SS SPACE — významná mezera (také GRAF
S)
DELETE — zrušit znak vlevo od kursoru
SS D — zrušit znak vpravo od kursoru
SS A — zrušit slovo
SS S — zrušit řádku
BREAK — tabulátor
Formátování textu:
SS W — vystředit řádku

SS Y — naformátovat odstavce
EM T — zarovnat řádek
EM R — zarovnat řádek vlevo
EM E — zarovnat řádek vpravo

Blokové operace:

EM D — smazat blok
EM C — kopírovat blok na pozici kursoru

Vyhledání a nahrazení textu:

EM F — vyhledat výskyt řetězce (menu
MLEDAT)
EM A — nahradit řetězec (menu MLEDAT)

Pořezání textu:

EM I — přesunout kursor na písmeno, kam
chceme dát diakritické znaménko
EM Q — čárka nebo háček (normální znaky)
EM W — čárka nebo háček (pouze pro š a ú)

Ostatní:

EM S — zvětšit řádkování
EM X — zmenšit řádkování
EM Z — zkopírovat řádku do pravítka
EM V — přepnutí režimu VLOŽ/PŘES
EM B — návrat do BASIC-u

PROGBASE

Zavedli jsme novinku zvanou PROGBASE ZXM. Jedná se o evidenci, databanku, v níž budeme shromažďovat údaje o programech. Zatím v ní redakce asi na 100 dat. V příštích číslech je budeme uveřejňovat podle abecedy. Zapojte se i vy !!! Podle níže uvedeného vzoru pošlete data o méně známých programech. V řádku hodnocení napíšete pro čtvři kritéria značkov.

Kritéria jsou - Zábavnost, Grafika, Hudba, Manuál. Značky jsou :
nejlepší - O
průměr - =
slabé - X

PROGBASE ZX MAZAZÍNU

NÁZEV PROGRAMU :
AUTOR PROGRAMU :
ROK VÝROBY :
DÉLKA PROGRAMU :
HODNOCENÍ ZGHM :
KÓD PROGRAMU :
OVLÁDÁNÍ JK :
MANUÁL TP :

OVLÁDÁNÍ - J - JOYSTICKEM
- K - KLÁVESNICÍ
MANUÁL - T - TAIWARD
- P - V PROGRAMU

KÓDY :

- (H) HRY - X1- STŘÍLEČÍ
X2- SIMULAČNÍ
X3- LOGICKÉ (KARETNÍ)
X4- KONVERZAČNÍ
X5- LECHTIVÉ
X6-6 OSTATNÍ
- (U) UŽIVATEL X1- TEXTOVÉ a DATABANKOVÉ
X2- PŘEKLADAČE a KOMPILÁTORY
X3- KOPÍROVACÍ
X4- TABULKOVÉ
X5- GRAFICKÉ
X6- MATEMATIKA a EKONOMIKA
X7- OSTATNÍ
- (V) VÝUKA X1- DEMO
X2- CIZÍ JAZYKY
X3- ZEMĚPIS
X4- P. OSTATNÍ
- (M) MUSIKA

vzrok :
podraz jakožto konverzační
bu napíšeme do
kódu - H24
HCA →
KONVERZAČNÍ
DVOU-
KOLNEKÉ

X- znamená, že pro dvourozměrné programy napíšete na místo X-2
a pro třírozměrné programy -3

POKE 1

ZX magazin

Ano, opět přinášíme seznamek příkazů POKE. Možná namítnete, že se některé opakují už asi po třetí, že patrně nemáme nic jiného na zaplácnutí časopisu, leč tentokrát nás k tomu donutila možnost A5 formátu a oboustrannosti. Doufáme, že vám tedy POKE 1 časopisu ZX magazin bude pomocníkem...

ANTIBREAK

23613,82 - ODPOJÍ RUTINU BREAK

23643,0 - SMAŽE PROGRAM PŘI BREAK

23659,0 - -II-

23610,0-255 -II-

23659,0 - ZAVEDENÍ NUL-EDIT. ZONŮ

23617,236 - PŘED INPUTEM, ŽEMĚNÍ BLIKAVÍCÍ KURTOR

154 ...
154 ...
204 ... DATA
252 ...
atd.

POKE 23570,10 odpojil klávesu EDIT
POKE 23570,16 nastavil listing při stisku klávesy EDIT
POKE 23736,181 autoSAVE (nahrávání bez systémového hlášení)

POKE 23743,80 na obrazovku nelze nic psát (mimo

ed. řádek) - např. pro zapřínění poškození obrázku (zákaz nutno v programu odvolat)

POKE 23743,83 zrušení zákazu psaní na obrazovku
POKE 23624,120 způsobí BRIGHT řádků 22 a 23
POKE 23691,120 způsobí BRIGHT řádků 0-22

HRY

TUTANOHAMUN 27783,0
TRAVEL W. TRASHMART 39650,163
TIRANOG (L) - po zabíli
, I - vracení do hry)

UNDERWULDE 39575-4,0 56380,0
39041,0 diamond-orientation

45018,201 odstraňuje příhr
59981,0 zbrání jsou na stejném místě

W.A.R. 38394,0
WHO DARES WINS 2 50833,163
WIZARD'S LAIR 25522,x počet životů

WAY OF THE EXPLODING více programů:
10 LOAD="SCREENS:LOAD"CODE:
ENR:LOAD"CODE:
POKE 44793,255:
RAND USR 39982

YABBA DABBA DOGO 43612,0

ZUB více programů:
10 REM ... I, I, I ...
20 BORDER NOT PL: INK NOT PL: PAPER NOT PL: OVER
SGH M

30 CLEAR VAL "24000"
40 LOAD "SCREENS:LOAD"CODE:
LOAD"CODE:
50 POKE 37473, 201
60 RAND USR VAL "24700"

ZAXXON 48825,x počet životů
ZP ZAP 53382,x nebo 54065,0
ZOOM 24743,0
ZAXXAN 56244-7,0
ZORRO 53729,0
3 D SPACE WARS 26244,0 20849,0

1942 více programů I. bloku:
10 CLEAR 64000
30 LOAD"CODE"
40 POKE 65332, 240: POKE 65333,255
50 RESTORE
60 FOR I=65520 TO 65527
70 READ A: POKE IA
80 NEXT I
90 RAND USR 65280
100 DATA 62,33,50,247
110 DATA 204,195,47,204

TRICKY THINGS FOR YOUR SPECTRUM

COLLECTED BY MARCUS PAULI

Daž unsere Leserschaft international ist, wurden wir, folgender Bericht aus dem Herzen Afrikas erreicht uns von unserem Leser Marcus Pauli, Kenya, den wir hier im Originaltext abdrucken. Wir würden uns freuen, wenn es unter unserer Leserschaft zu einem regen Austausch zwischen Europa und Afrika kommt.

POKE:	23756.9	Makes first line into line 0
	23756.1	Makes line 0 into line 1
	23609.0 - 255	Keyboard beeps
	23609.0	Keyboard silent
	23656.6	Cass Lock on. (...58.0 =>) off
	23736.1B1	Save by pass (only without INTERFACE 1)
	23692.255	Stops scrolling
	23613.0	If BREAK THEN RANDOMIZE USR 0
	23608.64	Buzz
	23562.1	Cursor on higher speed
	23562.5	Cursor on normal speed
	23636.150	Protects program, no BREAK possible
	23636.92	Unprotected program
	23726 - 23729	Are not used

FOR THE FOLLOWING POINTS REMEMBER TO WRITE PRINT & AFTER THE COMMAND!!

Let t = USR	3582	Scrolls screen one line
	4757	Prints "Sinclair" ornament but no clearing
	3212	Stops and asks Scroll?
	65536-USR7562	Prints Memo in Bytes
	3593	Scrolls bottom of screen one line
	3652	Clears top half of screen
	3350	Scrolls whole page to top line, by CLS
	1331	special border in magenta and blue
	1290	special border in red and blue
	1269	special border with special sounds
	1251	special border in green and magenta
	1248	special border in white and black
	1333	special border in cyan and blue
	1327	special border in white and blue
	1267	special border in blue and yellow
	455:	VERIFY

FOR THE NEXT POINTS WRITE PRINT PEEK BEFORE EACH NUMBER.

23627	Start of variables
23795	Start of Basic programs
23612	Shows line number which computer works on
23641	Address of statement which was out in last
23662	Line number which CONTINUE jumps to
23677	33 numbers of the printer position
23664	Address of PRINT position in the display
23730	Shows PANTOP
23560	Shows last flushed letter
23668	Shows length of emergency beep
23612	Shows flags which work together with TV

OTHER PRINT PEEK STATEMENTS, JUST A BIT MORE COMPLICATED

23627+256*PEEK 23628-PEEK 23635-256*PEEK 23636
Bytes used for Basic program, excluding variables

(23641+256*PEEK 23642) - (PEEK 23635+256*PEEK 23636)
Bytes used for Basic program, including variables

23730+256*PEEK 23731-PEEK 23653+256*PEEK 23654
Free memory (very correct display)

THE NEXT NUMBERS ARE RANDOMIZE USER COMMANDS, SO DO WRITE RAND USR BEFORE EACH NUMBER

6640 =>	Enter
7632 =>	Break
8000 =>	Pause 0
6200 =>	Prints R tape loading error
0000 =>	Resets the computer

SMALL PROGRAMS WHICH CAN BE VERY HELPFUL!

```
10 FOR X = 0 TO 99 STEP 10
20 PRINT PEEK (23627+X) + 256 * PEEK (23628 + X)
30 NEXT X
Shows the variables in the program
Let a = INT (a+100 + .5) / 100
Leaves 2 figures after a point
Let a$ = SCREEN$ (0,0) : PRINT a$,
Prints screen$
1 LET A=PEEK 23637+256*PEEK 23638:POKE A,0:POKE A+1,0:STOP
2 REM (The text which will be written in here can't be
removed)
LET time = (65536+PEEK 23674+256*PEEK 23673+PEEK 23672)/50
Gives time in seconds since the computer was turned on, and
resets every four days
```

HAVE A GOOD TIME WITH YOUR SPECTRUM, YOURS:

Marcus Pauli

PS: tato strana byla nechána v originále s motem: At se ti spektristé trochu procvičí v angličtině..

Jak si "schovat" obrazovku do paměti.

=====

Návod pro naprosto začátečníky ve strojovém programování.

Alé nejen uschovat - samozřejmě - také jak ji vyvolat zpět - a to tak rychle, jak nám to SPECTRUM dovolí.

Ono to vlastně začalo tím, že se na mne obrátil můj přítel Ondřej s žádostí, abych mu poslal jeden progránek, který u mne kdysi viděl. (Program se jmenoval GOBLET a byl z jakéhosi Sinclairovského časopisu). Trik tohoto prográčku spočíval v tom, že se na obrazovce velmi jednoduchým způsobem nakreslí nějaké těleso a program je překreslí vždy o něco pootočeně. Tenhle obrázek si program uloží do paměti a když má takhle definováno asi šest obrázků, tak je potom jeden za druhým - hezky svižně postupně překresluje na obrazovku a tam znikne velice reálný dojem, že se to těleso otáčí.

Ten můj známý mi to psal proto, protože ho ta "uschovávací" rutina zajímala a zajímalo jej také, zda by se dala takhle rutina z programu použít. A mně nad jeho dopisem trochu klesly ruce do klína, protože jsem si uvědomil, že ta rutina je v podstatě velice a velice jednoduchá, A L E pro toho, kdo se nevyzná ve strojovém kodu je to skoro neřešitelné.

A tak vzniklo tohle povídání. Myslím si totiž, že když jiz to budu psát pro přitele, že by to mohlo zajímat i další. A protože vím, že se Ondřej nevyzná ve strojovém programování, je tohle povídání určeno právě těm začínajícím programátorům, kteří se v něm nevyznají - a potřebovali by pro své programy tu obrazovku (nebo obrazovky) uschovat a zase vyvolávat.

A ještě něco - tento článek nechce konkurovat kurzům programování ve strojovém kodu, takže si nebudeme vůbec vysvětlovat proč a proč něco zrovna takhle, ale řekneme si prostě jak a k tomu udělat, aby to fungovalo.

Obrazovka, její paměť, délka a umístění.

=====

Cokoliv namalujete nebo napíšete na obrazovku, to vše si pamatuje obrazovková paměť jako souvislou řadu čísel. Jak si to převádí a jak to dělá je v tuhle chvíli vedlejší. Důležité je vědět, že paměť obrazovky začíná na adrese 16384 a je celkem dlouhá 6912 bajtů, čili že je to tedy souvislá řada čísel dlouhá skoro 7 kilo. To co je zrovna zapísáno v paměti na téhle adrese, to je současně zobrazováno na obrazovce. Když na těchto adresách budeme mít samé 0, bude obrazovka prázdná. Když tam budeme mít samé 255 tak bude obrazovka černá.

P A M A T O U A

Obrazovková paměť začíná na adrese 16384 je dlouhá 6912 bajtů

Princip uschování a vyvolání obrazu.

=====

Paměť SPECTRA je souvislá řada paměťových buněk od 0 do 65535. BASIC a ostatní pomocné části obsadí adresy od 0 do 23754 a od adresy 23755 začíná náš program. Jsme-li začátečníci tak pravděpodobně neobsadíme celou zbývající volnou paměť. Ostatně ani ji celou volnou nemáme, stačí když budeme vědět, že je volná zhruba do adresy 65000. Jestliže chceme uschovat obraz nakreslený na obrazovce znamená to vlastně, že potřebujeme paměť obrazovky (tedy kus paměti od adresy 16384 o délce 6912 BYTE) okopírovat někam dozadu, za BASIC program. Kdykoliv později si tuhle část paměti můžeme zase přečíst - to znamená překopírovat ji nazpět od adresy 16384 - čili vlastně - zobrazit ji na obrazovce.

Výpočet je tedy jednoduchý. Potřebujeme uschovat o něco málo méně než 7 kilo. Tedy $65000 - 7000 = 58000$

Paměť druhé obrazovky budeme začínat na adrese 58000. Je samozřejmé, že obraz nemůžeme uschovat kamkoliv, kde je volné místo, ale "načpene" ho takhle dozadu proto, aby nám zbylo co nejvíce místa na BASIC.

Abychom tuto rezervovanou oblast skutečně uchránili, tak si před ní předsuneme RAMTOP.

Co je to RAMTOP? Je to takový "horní doraz pro BASIC", takové znaménko, které nedovolí BASICU aby byl delší než chceme. Posadíme tedy RAMTOP těsně před adresu 58000, tedy na adresu 57999. To uděláme jednoduše příkazem: CLEAR 57999. (On toho ten příkaz udělá víc, především smaže všechny proměnné, ale nám stačí, že budeme mít dostatečně velký kus paměti rezervovaný pro uschování obrazovky.)

P A M A T O U A T

Obraz uschováme přepsáním 6912 bajtů z obrazovkové paměti na předem zvolený úsek paměti.

Obraz vyvoláme přepsáním tohoto úseku paměti na adresu 16384. Omezení úseku paměti provedeme předstunutím RAMTOPU na adresu nižší příkazem CLEAR (adresa počátku rezervace - 1)

Jak přesuneme paměť.

=====

Musíme si trochu ozřejmit situaci - máme tedy nějaký obraz, který chceme uschovat do paměti. Ten obraz ani nemusel vzniknout samotným programem, ve kterém byl používán, mohli jsme ho vytvořit třeba nějakým kresličským programem, dejme tomu, že jsme použili Hutchbyho ART STUDIO. Nakreslený obraz máme tedy na obrazovce. Přesunout dozadu (na adresu 58000) jej umíme i BASICEM, tuto maličkost o stará jediný řádek:

```
100 FOR n=0 TO 6911 : POKE 58000+n, PEEK (16384+n) : NEXT n
```

Mimochodem - přesun z t. by vypadal naprosto stejně, jen adresy by byly mezi sebou prohozeny.

J E N Z E - než se tenhle příkaz provede, tak to trvá bratru půl minuty. To ještě tak můžeme použít při tvorbě nějakého programu, když jej píšeme a nezáleží nám na té chvilce času, ale v konečném programu to musíme udělat podstatně rychleji, takže ať chceme, nebo ne, sáhneme po strojáku.

Co je to strojový program a jak se zapisuje.

=====

Poněkud zjednodušeně se dá říci, že je to jazyk stejně jako třeba BASIC. Pochopitelně, že je tady řada rozdílů. Basic zapisujeme pomocí slovních příkazů, které jsou každému (Angličanovi) naprosto jasné. Stroják pomocí řady čísel. BASIC hlídá a kontroluje co může. Nechá nás udělat chybu. Jenom těch různých chybových hlášení co má. Od onoho hrůzostrašného "Nonsense in BASIC" (Nesmysl v BASICU) přes desítky různých hlášení, které mnohdy ani na obrazovce nevidíme - mimochodem, komu se už napsalo na obrazovce "Parameter error" (špatný parametr) - chyba vznikající při nedorozumění v zadávání funkce - až po to nejběžnější "OK" (vše v pořádku).

Basic hlídá i syntaxi - když zapomenete třeba v příkazu PRINT udělat uvozovky, tak vás BASIC vůbec nepustí z editační zóny a ještě v místě chyby rozbliká otazník.

Tohle všechno stroják neumí. Příkazová slova nezná, používá čísla - místo RETURN musíte napsat 201 - atd. Chybu vám klidně dovolí napsat a pak na ni spokojeně havaruje. Když mu poručíte (samozřejmě omylem) aby běhal dokolečka, jako pes za svým ocasem, tak se uběhá. Nemyslete, tato chyba se dost lehce stane a je to jedna z jednoduchých chyb - vypadá třeba tak, že strojový program narazí na příkaz skoč o dvacet kroků dopředu. A o dvacet kroků vpředu je příkaz skoč o dvacet kroků zpátky.

Stroják to udělá a bude skákat svou ohromnou rychlostí tam a zpět a zastavit ho může jedině RESET tlačítko, nebo úplné vypnutí.

Zlatý BASIC, řeknete si. Nu ano, jenže ten BASIC pro samé hlídání, kontrolování ale hlavně překládání má nakonec na samotné výkonné povely hrozně málo času. Takže když něco potřebujeme opravdu rychle, tak to uděláme v tom strojáku.

Zapisování strojáku je kapitola sama pro sebe. Pro vlastní zápis strojáku jsou vymyšlena různá pomocná programová vybavení - to všechno se naučíte, budete-li chtít v kurzech strojového programování. My si - protože se budeme pohybovat v oblasti opravdu jednoduchého strojáku - převedeme povely přímo na čísla a tato čísla vložíme do paměti.

P A M A T O U A I

Stroják je základní jazyk, který se zapisuje do paměti jako souvislá řada čísel. Pracuje 50 až 100 x rychleji než BASIC.

Jak si stroják pamatuje čísla a jak se zapisují.

=====

Když si potřebujete v BACICU nějaké číslo (třeba 14) zapamatovat, zvolíte si jméno proměnné (třeba x) a napíšete příkaz:

100 LET x=14

a je to hotovo. Ve strojáku je to hodně podobné, ale má to přece jenom několik odchylek. Předně - v BASICU si jméno proměnné volíte. Může to být x, může to být TTT, může to být T32XA7 a když na to přijde může se proměnná jmenovat třeba "V úterý nebude pršet". Proměnných můžete mít kolik chcete - dokud máte volné místo v paměti.

Ne tak ve strojáku. Proměnných je tam jenom několik (asi tak 20 - za určitých okolností) a jmenují se A, B, C, D, E, F, H, L a vy si jenom můžete do kterého chcete něco psát.

Az potud to je asi jasné. Komplikace však hned přijdou - tyhleto proměnné se ve strojáku jmenují REGISTRY, a největší číslo, které do nich můžete zapsat je 255 (a to ještě ke všemu vůbec neznačí záporná čísla, jenže ty my nastěti potřebovat nebudeme).

Když ale potřebujeme zapsat větší číslo než 255, tak to děláme tak, že si na to vezmeme dva registry. Představte si třeba takový příklad: potřebujeme spočítat určité množství kuliček - a máme na to dva pohárky z umělé hmoty. Do každého pohárku se vejde přesně 256 kuliček. Takže - dáme postupně kuličky do jednoho pohárku a když je plný, tak jej vysypeme vedle a do druhého pohárku si dáme kuličku na znamení, že jsme jednou napočítali do 256. A zase budeme vkládat kuličky a když naplníme pohárek podruhé, dáme druhou kuličku do toho pamatovacího a tak dál. A budeme to dělat tak dlouho, až nám zbyde několik kuliček, které již pohárek nenaplní. Kolik ze těch kuliček vlastně bylo? No 256 krát počet kuliček v tom pamatovacím + ty kuličky co nám zbyly.

S registroumi páry je to stejné. Máme třeba jeden registr, který se jmenuje H a jeden, který se jmenuje L. Dohromady vytvoří REGISTROU PAR HL. A potřebujeme do něj zapsat číslo 520.

520 je 256 + 256 + 8. Registr H je "pamatovací pohárek", takže do něj zapíšeme 2 (2x jsme naplnili registr L dvěstěpadesáti šesti) a do registru L dáme potom to co nám zbylo - čili 8.

Matematicky je to jednoduché:

$$\text{REGISTR H} = \text{INT}(520 / 256) = \text{INT } 2,03125 = 2$$

$$\text{REGISTR L} = 520 - (256 \times \text{REGISTR H}) = 520 - 256 \times 2 = 520 - 512 = 8$$

A úplně nejjednodušeji to jde na kalkulačce asi takle:

- Napíšete 520 : 256 = a na zobrazovací budete mít 2,03125. Poznamenáte si na papír celky, které vám vyšly - tedy 2, což je obsah registru H.

- Od hodnoty, kterou máte na zobrazovací odečtete toho celky.

čili máte tam 2,93125 a napíšete - 2 = a na zobrazovací budete mít 0,93125.

- Toto číslo vynásobíte 256. Tedy napíšete $\times 256 =$ a na zobrazovací budete mít 8 a to je obsah registru L.

I. P A M A T O V A T

Číslo, které chceme zapsat do registrového páru HL rozložíme tak, že ho dělíme 256. Celky jsou obsah registru H, odečteme celky a zbytek násobíme 256, což je obsah registru L

Je samozřejmé, že s ostatními registrovými páry (BC, DE) je to stejné.

Zapisování takovýchto čísel do paměti probíhá naprosto stejně, píšeme takto získané hodnoty za sebou, ale pozor na skutečnost, že nejdříve se vždy píše tak zvaný méně významný bajt, tedy ten ve kterém máme zbytek (v naší ukázce by to byl ten, který obsahuje číslo 8 a z a n ě m teprve píšeme ten "pamatovací" - říkáme mu významný bajt - ten, který obsahuje číslo 2. Takže, kdybychom chtěli na příklad na adresu 6+200 uložit číslo S20 zapíšeme: POKE 6+200,8:POKE 6+201,2.

P A M A T O V A T

Všechna čísla ukládaná do dvou bajtů se zapisují v pořadí :
méně významný bajt (druhý registr z registr. páru) L C E
více významný bajt (první registr z registr. páru) H B Ď

CO BLBNE ?!?



MA' RADOST, ŽE
DOSTAL NOVÉ ČÍSLO
Z MAGAZÍNU !!

Přesun pametového bloku strojovým programem.

Pro přesun paměti má strojový kód svoji vlastní instrukci, která se jmenuje:

L D I R

Je to "Blokový přenos s inkrementací a opakováním". Když si to řekneme trochu srozumitelněji, bude to znít asi takto:

Povel zkopíruje kus paměti z místa, (jehož adresa je zapsána v registrovém páru HL) na místo, (jehož adresa je zapsána v registrovém páru DE), přičemž délka přeneseného bloku je zapsána v registrovém páru BC.

Tento povel má dvě kódová čísla. To znamená, když jej chceme provést, tak musíme napsat obě za sebou. Je to 237 a 176.

P A M A T O V Ě A I

do HL zadáme ODKUD

do DE zadáme KAM

do BC zadáme KOLIK

a dáním povelu LDIR přeneseme celý blok

Všechny adresy jsme si již připravili, takže to bude vypadat takto:

uschování obrazu

vyvolání obrazu

ODKUD	- do HL zadáme	16384	58009
KAM	- do DE zadáme	58009	16384
KOLIK	- do BC zadáme	6912	6912

Jak zapsat číslo do registrového páru.

Když chceme napsat něco do proměnné v BASICU použijeme prostě povelu LET. Stroják má podobný povel (u mnemotechnické řeči se jmenuje LOAD a píše se LD) jenže povel LET platí pro všechny proměnné, zatímco LD je pro každý pár jiný.

Takže povelu LD HL řekneme strojáku, že má zaplnit registrový pár, ale ještě mu musíme říci čím ho má zaplnit. To je však dáno přesným významem povelu LD HL. Víme, že do paměti nebudeme psát žádné LD HL (to bychom psali, kdybychom použili nějaký překladáč - třeba GENS), ale budeme tam psát přímo kódové číslo tohoto povelu - to je 33.

Když tedy program "jede" po jednotlivých povelích a narazí na povel 33 znamená to pro něj: "Vlož do registrového páru HL číslo, které si přečteš za touto instrukcí a to tak, že do registru L vložíš to, co je napsáno ihned za instrukcí a do registru H vložíš to, co je zapsáno za tím. Potom tato dvě čísla přeskočíš a budeš pokračovat plněním instrukce, která je zapsána

INZERCE :

- Josef Musil, Špičák 125, 38101 ČESKÝ KRUMLOV: shání cartridge do ZX Microdrive, libovolné množství. Dále shání levně mechaniku disketové jednotky pro ZX S.

INFORMACE :

DNES POTĚŠITELNĚ

SPORTKOMPLEX



■ ELEKTRONICKÁ BURZA,
setkání majitelů osobních počítačů.
Každou první sobotu v měsíci.
Součástky, polotovary, finální výrobky.
Výměna programů, zkušenosti.
Pořádá Amiga klub a TJ Motorlet ve Sportkomple-
xu přímo na stanici metra Radlická od 8 do 16 hod.
Rezervace stolů a informace na tel. 53 82 55.
Zn.: „P931414720.“

Uvedené hodnoty představují i přibližný poměr zastoupení jednotlivých firem na britském trhu. Očekávaný nástup 16-bitových domácích počítačů je však podstatně pomalejší, než se všeobecně předpokládalo.

Spectrum	43.7%
C64	24.3%
Amstrad	17.7%
Atari ST	4.6%
IBM Amiga	3.0%



*A PAK
ŽE
ZHYNUL !!*

Prodej softwaru ve Velké Británii
Podle týdeníku Computer Trade Weekly je podíl prodaného programového vybavení v září 1988 následující:

SINCLAIR PC200

má standardní vybavení IBM PC, klávesnici typu AT (102 tlačítek), vestavěnou disketovou jednotku 3.5" a FM výstup. Lze k němu přikoupit i monitor-pak je součástí sestavy i joystick, 4 hry a program PC Organizer zdarma. Cena přibližně 250 GBP.

Sir Clive Sinclair

zahajuje prodej systému pro satelitní příjem za 150 GBP. Systém je určen pro příjem 14 kanálů družice Astra a je zajímavý tím, že plochá přijímací anténa má tvar čtverce o hraně 60 cm.

*AT ŽIJE
STREJDA
KLAJV !!
(BT: CLIVE)*

ZÁBAVA :

Dnes pro vás máme přímo lahůdkovou SCI-FI povídku amerického autora J. Browna. Vychutnejte :

- Poslední člověk na Zemi seděl sám v místnosti.
V tom někdo zakleval na dveře...-

je tak trochu nenápadným, ale charakteristickým rysem toho kterého počítače či systému. My spektristé známe především původní písmo Romky ve znakovém čtverci 8x8 bodů. Také C-64 má 8x8 bodů, však písmo je již jiného typu, tučnější a hlavně při hustotě 320 bodů na řádek vychází úzký obdélník: 40 znaků je až dost. A přesto i do Commodora pronikl strýček Clive! Existuje emulátor Spektre Basicu i s typickým písmem a 32 znaky na řádek.

Proč o tom mluvíme? Mezi programy je řada takových, jenž nám umožní vytvořit vlastní základní abecedu a po příslušném "pouku" ji plnohodnotně využívat. Asi každý z nás se o to pokusil a ta jeho abeceda je ozdobou nebo "ozdobou" programu.

Pisatel těchto řádek má výtvarně - typografické zaměření a k počítači přišel náhodou. Ano, původně jsem kupoval počítač jako řachově trénovacího souneře. Ale brzy jsem pochlehl kouzlu basicu a jeden z prvních pokusů byla proporcionální abeceda s mezerami titulkové sazby, vysoká 32 bodů a dost pomalu "chodící". Ted pokouším assembler (nebo on mne). Ale stále, třeba jen sám pro sebe, dělám abecedy. Základní 8x8, poloviční 4x6, ale i 6x8 a jiné. A také hodnotím, sám pro sebe, nebo pro známé.

Nejdůležitější je čitelnost. Určitě si vzpomenete na program i importovaný, kde je nutno písmena doslova luštit. Omezený počet bodů využitý na "kudrlinky" nedovolí zdůraznit rozlišovací typické tvary. Příkladně u hranatého písma širokého někdy těžko rozeznáme D-O, A-R, U-V, při chybném tvaru i C-G, I-J, K-H, písmeno či I bez patek (vodorovné čárky nahoře a dole) bude vypadat jako chudinka. Úzka písma (4x8) musí nutně zajistit maximální čitelnost: w, m, n je obtížné v šířce 3 bodů vůbec udělat.

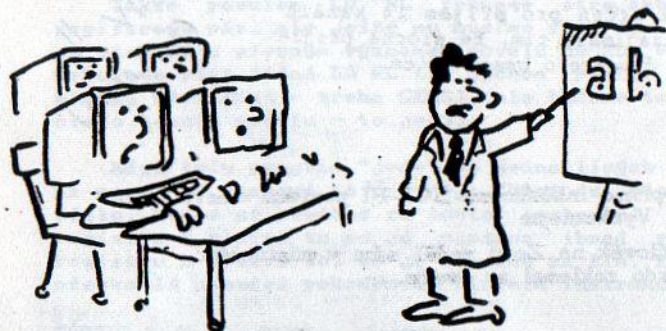
A na závěr zajímavosti: Jiřímu Lamačovi se příliš nelíbila původní abeceda ROM a tak v jeho LEC-Romce je nová, osobitá, myslím, že mnohem lepší a čtivější. Jen některé znaky jsou diskutabilní, např. minuska "r" a verzálka "l". Téžoo autora úvodní obrazovka CP/M 2.2 je také velmi zdařilá, nadprůměrná. Za kosmetickou vadu považují jen úzko osazený diagonální tah číslovky 2 (v 3D titulním nápisu) a "překukující" atributy. Kartriďz program "Q-cart" též od J.L. je opět vzorem dobrého grafického zpracování, na rozdíl od podobně zaměřěného O-cony (M. Moráček). Obě úvodní obrazovky jsou hluboko pod možnostími grafiky spektra i kvalitou kopíráku (průběžné zobrazení obsazování paměti).

Výborné písmo 6x8 má assembler LASER GENIUS - osobité a dobře čitelné, zatímco EDIT-ASSEM ve stejném znakovém poli má abecedu mnohem horší.

Různé velká, ale vždy perfektní písma (i celé obrazovky) mají fy OCEAN, US GOLD, IMAGINE a celá řada dalších. V domácí produkci se můžeme poučit u firmy FUXOFT, AMISIN, doporučují i písmo a celou obrazovku programu EXPERT - autor OKF, nebo úpravné Menu Datalogu P. Adámka; dobře ovladatelné. Úvodní obrazovka je naopak odstrašujícím příkladem. I v úvodním režimu jimž je kreslena, stačilo jen trochu jinak uspořádat čtverečky, z nichž se skládá.

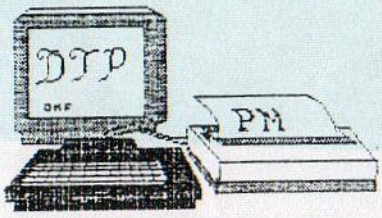
Grafické řešení se rozhodující měrou podílí na "profesionálním" dojmu z programu a proto chybu činí ten kdo mávne rukou: "Hlavně, že to vůbec chodí...".

V. Kolenský



THE PAGE MAKER™

Zde vidíte praktickou ukázkou
 vyzkoušíte program PageMaker. Jenž
 umožňuje vytvářet DTP (desktop
 publishing) na mikropočítači
 IBM, Spectra a tiskárnou Brother
 1100.



K systému DTP umožňuje automatic-
 kým lamami stránku z obrázku a
 textu. Je to jednoduchá a spří-
 ká profesionálních programů
 počítačů třídy IBM PC pro-
 jekty, které umožňují tiskárna-
 m různých typů připojit se k tisk-
 ňí. Program PageMaker umožňuje
 vytvářet různé typy dokumentů.

Tento program umožňuje tisk
 pomocí textového editoru typu
 IBM, nebo pomocí tiskárny
 IBM, nebo pomocí tiskárny
 IBM, nebo pomocí tiskárny



Ukázka, jak se pracuje s
 textem a grafikou. Ukázka, jak
 se pracuje s textem a grafikou.
 Ukázka, jak se pracuje s textem
 a grafikou. Ukázka, jak se
 pracuje s textem a grafikou.

ve kterém se vymezi prostor pro
 obrázek, který se vytváří pomocí
 nástrojů vytvořené obrazovky se
 nazývá "okénko". V okénku se
 provádějí všechny úpravy. To se
 provádějí v okénku, ve kterém lze
 zobrazit všechny stránky scroll-
 ovaně. Práce se systémem
 probíhá tak, že uživatel stávkou
 vlastní tiskárny. Ukázka, jak
 se pracuje s textem a grafikou.
 Ukázka, jak se pracuje s textem
 a grafikou. Ukázka, jak se
 pracuje s textem a grafikou.
 Ukázka, jak se pracuje s textem
 a grafikou. Ukázka, jak se
 pracuje s textem a grafikou.

Ukázka, jak se pracuje s
 textem a grafikou. Ukázka, jak
 se pracuje s textem a grafikou.
 Ukázka, jak se pracuje s textem
 a grafikou. Ukázka, jak se
 pracuje s textem a grafikou.

Ukázka, jak se pracuje s
 textem a grafikou. Ukázka, jak
 se pracuje s textem a grafikou.
 Ukázka, jak se pracuje s textem
 a grafikou. Ukázka, jak se
 pracuje s textem a grafikou.



Kresba **MP**

Ukázka, jak se pracuje s
 textem a grafikou. Ukázka, jak
 se pracuje s textem a grafikou.
 Ukázka, jak se pracuje s textem
 a grafikou. Ukázka, jak se
 pracuje s textem a grafikou.



PAGEmaker
 PAGEmaker
 PAGEmaker
 PAGEmaker

PAGEmaker
 PAGEmaker

Konec

Tuto ukázkou zpracoval textové i grafické D.Kafka OKF dne
 21. června 1989.

SPECTRUM

SINCLAIR

1. TO JE ON!
A'S HLAVNA' (NICOLI
HRDINA (NICOLI
PROGRAM OD
MHWLY) --



A TO JE JEHO MAMI
POČTAC SINCLAIR

2. COHKA KONTEJNER
A CO DAL? MO
NA'S HRDINA
MA' POČTAC ALE
TO
NESTACI... ZDALEKA...
CO S NI?!!
MA SE HRDINA...
JE TO JAKO ČEJE NARUŠIT
POČTAC... NE TAK TO
JAKO ČEJE NARUŠIT
POČTAC... NE TAK TO



3. UŽ HRDINA BEŽKAPNOSTI
PRO PADA'...



(TĚA SE NA PEČI AŽ JE
JÍŠ BIVA NEJAK
OD SPECTRA (KAPNOSTI))

NEVĚŽ HLAVU !!!
TOT RADA KOUĐRA'
ODBÍREJ



ČASOPIS ZX MAGAZÍN !!

(DR. LOFT) 439 23
SPEKTRUM MAGAZÍN - DAVID HERTL LENEŠICE 8. NĚMCOVÉ 127

-ENDE

ZX MAGAZÍN
 Redakce : Dr. Software
 Študák Soft
 OKF

 Adresa : D. Hertl
 B. Němcové 127
 Lenešice
 439 23