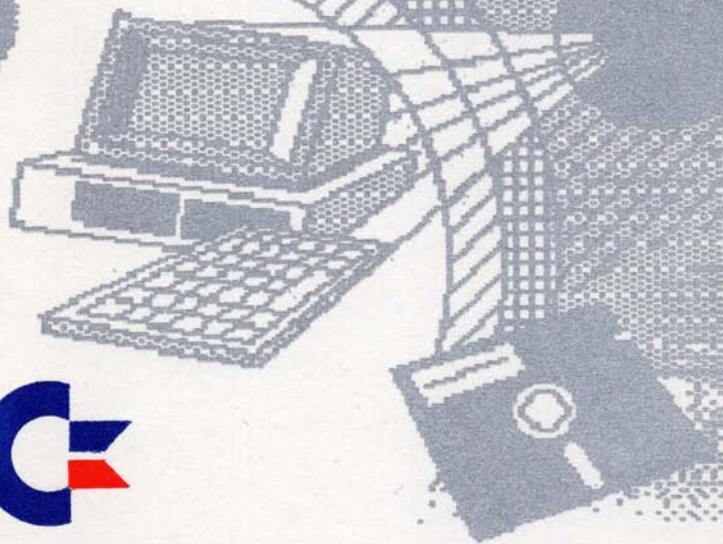


Commodore Klub



KRAJSKÁ SEKČIA PRI KDPM KOŠICE

SPRAVODAJ

7

Založenie AMIGA klubu v Košiciach

(Milan BOBULA)

20. januára 1990 sa v priestoroch KDPM uskutoční prvé stretnutie užívateľov počítačov Amiga. Zároveň oficiálne zahájí svoju činnosť Amiga klub. V tejto súvislosti upozorňujeme záujemcov o aktívne členstvo, aby zaslali žiadosť o zavedenie do členskej základne na adresu Milan Bobula, Olbrachtova 10, 040 01 Košice.

Hlavným zámerom klubu bude združenie užívateľov všetkých typov počítačov Amiga, ktorí sa chcú svojimi vedomosťami a skúsenosťami podieľať na rozvoji klubovej činnosti. Ide nám teda skôr o angažovaný prístup členov, než o neprimerané rozširovanie členskej základne.

Na základe Vášho záujmu vám zašleme prihlášku a ďalšie spresňujúce informácie, týkajúce sa prevádzkových priestorov, termínov stretnutí, náplne a pod. Novovznikajúci klub je otvorený vašim návrhom a podnetom, ktoré následne pomôžu skvalitňovať jeho činnosť. Mal by byť organizovaný na platforme voľného záujmového združenia s takou náplňou stretnutí, ktorá by pružne reagovala na požiadavky a angažovaný prístup každého člena.

Klubové oznamy

Commodore klub zahajuje svoju činnosť 13. januára 1990 v priestoroch KDPM, v miestnosti č. 403. Stretávať sa budeme v párne týždne. Dochádza k rozdeleniu klubu na dve samostatné sekcie:

1. sekcia C-16, 116, +4 a VIC 20 má zabezpečené prevádzkové priestory v čase od 10.00 do 13.30 hod.
2. sekcia C-64, 128 v čase od 15.30 do 18.30 hod.
 - v čase od 14.00 do 15.30 prebiehajú kurzy programovania v Basicu a Assembleri. Tieto úpravy si vyžiadal nedostatok vhodných priestorov pre klubovú činnosť
 - v roku 1990 plánujeme vydávanie ďalších štyroch čísel Spravodaja, v ktorých sa budeme snažiť uverejniť čo najviac zaujímavých článkov pre užívateľov počítačov Commodore.
 - vo februári pripravujeme výročnú členskú schôdzu. Po spresnení termínu vám čas a miesto konania včas oznámime

FINAL CARTRIDGE 2

V čísle 6/89 sme začali uverejňovať manuál k veľmi obľúbenejmu doplnku počítačov C-64/128. V tomto čísle pokračujeme druhou časťou.

5. MONITOR STROJOVÉHO KÓDU

5.1 Prehľad funkcií

„Final Cartridge“ je vybavená veľmi výkonným monitorom strojového kódu. Nasleduje v abecednom poradí krátky prehľad príkazov monitoru:

- A assemblovanie s vstavaným priamym assemblerom
- C porovnanie (compare) obsahu pamäte
- D disassemblovanie programu v strojovom kóde
- F zaplnenie určitého rozsahu pamäte žiadanou hodnotou (\$00 ... \$FF = 0 ... 255)
- G odštartovanie (go) programu v strojovom kóde
- H hľadanie určeného obsahu pamäte
- L „loudovanie“ programu s voľbou miesta v pamäti
- M zobrazenie ľubovoľnej oblasti pamäte RAM alebo ROM s možnosťou zmeniť obsah RAMky
- O presmerovanie medzi oblasťami pamätí RAM a ROM počítača (bank-switching) a RAM a ROM pamätí disku
- P prevedenie obrazkového výstupu na tlačiareň
- R ukázanie obsahu registra
- S „sejvovanie“ programu z ľubovoľnej oblasti pamäte
- T prenos obsahu ľubovoľnej oblasti pamäte do inej oblasti
- X opustenie monitoru a návrat do Basicu
- @ dopyt na status, prípadne chybové hlásenie diskety; s prídavnými údajmi odoslanie diskového príkazu (ako DOS)
- # konverzia decimálneho čísla na hexadecimálne
- \$ konverzia hexadecimálneho čísla na decimálne
- *R prečítanie sektoru z diskety a uloženie jeho obsahu do ľubovoľnej oblasti pamäte
- *W zápis sektoru na disketu z ľubovoľnej oblasti pamäte počítača

Príkazy a ich syntax budú na nasledujúcich stránkach detailne prebrané.

Monitor nezaberá v pamäti počítača žiadne miesto, takže máte neobmedzenú voľbu miesta pre vaše assemblerovské programy.

Aby ste si mohli prehliadnuť rôzne oblasti pamäte a pre disassemblovanie, môžete použiť dopredné a spätné rolovanie na obrazovke, takže môžete na jedno vyvolanie vidieť ľubovoľne veľké súvislé oblasti pamäte.

5.2 Volanie monitoru

Monitor môžete zavolať niekoľkými spôsobmi. Druh volania závisí aj od

toho, či monitor chcete volať z chráneného (normálne neprerušiteľného) programu alebo z programu, ktorý sa dá jednoducho zastaviť pomocou RUN/STOP, alebo ho jednoducho voláte z RESET menu, alebo zo stavu READY.

Do monitoru sa dostanete:

- a) zo stavu READY, prípadne po prerušení programů pomocou RUN/STOP. Stlačiť funkčnú klávesu F2 (zadáť príkaz MON).
- b) z bežiacieho chráneného programu:
 1. Zavolať FREEZE menu (tlačidlo FREEZE na FC),
 2. z FREEZE menu stlačiť klávesu F1 (menu 2),
 3. z menu 2 stlačiť F7 (Resest),
 4. z RESET menu aktivovať monitor klávesou F1.

Monitor sa okamžite ohlásí tým, že predvedie na obrazovku stav všetkých registrov. Uvidíte približne nasledujúci obrázok (obsahy registrov sa budú meniť):

```
C*
   PC          IRQ  SR   AC   XR   YR   SP
, , ADD4      EA31 B1   BD   FF   05   F3
```

Skratky registrov:

Program Counter, Interrupt ReQuest, Status Reg.,
ACcumulator, X Register, Y Register, Stack Pointer

Za bodkou v poslednom riadku bliká kurzor. Teraz môžete zadávať monitoru príkazy.

5.3 Obsadenie funkčných kláves

V monitore dostávajú funkčné klávesy nový význam:

F3 kurzor do HOME pozície (ľavý horný roh), ďalšie stlačenie klávesy spôsobí spätné rolovanie zvoleného zobrazenia, a to tak dlho, pokiaľ držíte klávesu.

F5 kurzor do ľavého spodného rohu, ďalšie stlačenie spôsobí rolovanie zvoleného zobrazenia vpred tak dlho, pokiaľ klávesu neuvoľníte.

F7 ukáže obsah diskety (directory) bez straty programu. Zodpovedá príkazu monitora @

Funkčné klávesy F1, F2, F4, F6 a F8 nemajú žiadnu funkciu.

5.4 Bankswitching (prepínanie medzi ROM a RAM)

Niektoré oblasti pamäte sú, pokiaľ sa týka adres, viackrát obsadené. Tak je napr. v určitej oblasti pamäte Basic interpreter a operačný systém, aby boli po zapnutí okamžite k dispozícii. Oblasti RAMky, adresovo ležiace „pod“ týmito oblasťami ROMky nemôžete bez ďalšieho neobmedzene používať. Môžete síce do týchto oblastí zapísať, ale čítať sa bude z oblasti ROMky.

Bankswitching vám umožňuje čítať podľa potreby z ROMky alebo tiež z RAMky, ležiacej pod ROMkou. Tým máte k dispozícii ďalšiu cennú pamäť.

Bankswitching je ovládaný nastavovaním príznakového bitu 1 Bytu nultej stránky. Nastavovanie sa týka bitu 0, 1 a 2.

Máte k dispozícii 8 príkazov O0–O7 (prvý znak je písmeno o, nie číslica!). Ďalší príkaz OD slúži k prepínaniu RAM a ROM disku (viď kapitola 6).

V ďalšej časti sú tabuľkovo uvedené oblasti pamäte, ktoré monitor po jednotlivých príkazoch Bankswitchingu sprístupní.

Oblasti pamäte (decimálne)

príkazy	0	32768	40960	49152	53248	57334
	32767	40959	49151	53247	57343	65535

O0	RAM	RAM	RAM	RAM	RAM	RAM
O1	RAM	RAM	RAM	RAM	RAM	RAM
O2	RAM	RAM	FC-ROM	RAM	CHAR-ROM	KERNAL
O3	RAM	FC-ROM	FC-ROM	RAM	CHAR-ROM	KERNAL
O4	RAM	RAM	RAM	RAM	RAM	RAM
O5	RAM	RAM	RAM	RAM	I/O-REG	RAM
O6	RAM	RAM	FC-ROM	RAM	I/O-REG	KERNAL
O7	RAM	FC-ROM	FC-ROM	RAM	I/O-REG	FC-ROM

Vysvetlivky:

RAM – normálna RAM pamäť počítača

CHAR-ROM – znaková ROM-ka

I/O REG – vstupné/výstupné registre a časovač (CIA#1 a CIA#2)

KERNAL – ROM operačného systému

FC-ROM – ROM Final cartridge – nedá sa prekopírovať do RAM-ky! V oblasti \$A000–\$BFFF je uložený FC-BASIC interpreter

Štandardné nastavenie pre adresáciu RAM a ROM je O7.

5.5 Ukázanie a zmena obsahu pamätevej bunky

Jedna z najjednoduchších funkcií monitoru je ukázať obsah vami zvolenej adresy a umožniť jej zmenu, ak sa jedná o pamäť RAM. Ak ste sa naučili pomocou BANKSWITCHINGU určovať oblasť pamäte, môžete MEMORY príkaz používať.

Zobrazenie obsahu pamäte bude v hexadecimálnom tvare a kým je to zobraziteľné, tak aj v znakovkej forme (písmená, číslice, zvláštne znaky, bloková grafika). Každý zobrazený riadok obsahuje 8 Bytov (vľavo hexadecimálne, vpravo znakový ekvivalent). Riadok vždy začína adresou prvého z ôsmich zobrazených Bytov. Môže to vyzeráť napríklad takto:

```
. : 8000 46 49 4E 41 4C 20 43 41 FINAL CA
```

Na to, aby ste takýto výpis dostali, stačí zadať **M8000**. Pri zadávaní príkazov monitora dávajte pozor na to, aby ste písali do stĺpca, kam umiestnil monitor kurzor. Inak by mohlo dôjsť k problémom pri interpretácii príkazu, čo monitor signalizuje otáznikom (?). Takmer pre všetky vstupy nastavuje monitor na začiatok vstupného riadku bodku a za ňu kurzor.

Syntax: **M xxxx,yyyy** alebo **M xxxx yyyy** alebo **M xxxx**

xxxx – začiatočná adresa

yyyy – koncová adresa, medzeru po M možno vynechať

Príklad: M 8000 8100 ukáže obsah pamäte od adresy \$8000 do \$8100

Ak chcete vidieť obsah pamäte, ktorá priamo susedí zhora a zdola so zadanou oblasťou, môžete pohodlne rolovať obrazovku. Použijete na to klávesy F3 a F5 alebo CRSR. Akonáhle sa kurzorom dostanete na spodný okraj obrazovky, začne sa obrazovka rolovať hore a objavia sa vyššie adresy.

Iste ste si všimli, že každý riadok začína . : (bodka a dvojbodka). Ak v niektorom riadku nahradíte dvojbodku čiarkou (,), zistíte, že od nasledujúceho riadku je zapnutý disassembler (bližšie v časti 5.7). Ak potom nahradíte opäť úvodné znaky ., na . : bude prepnutý znova výpis obsahu pamäte.

Ako zmeniť obsah adresy? Veľmi jednoducho. Pohnete kurzorom na Byte, ktorý chcete zmeniť (ale len na hexadecimálnu časť) a prepíšete hodnotu na takú, akú potrebujete. Potom potvrdíte pomocou klávesy RETURN.

Všimnite si, že zmena obsahu adresy sa robí vždy po riadkoch, takže sa dá zmeniť naraz hodnota 8 Byte na jednom riadku.

Pozor, ak máte zobrazenú nejakú oblasť ROM-ky! Ako je známe, obsah ROM sa nedá meniť. Zmeny, ktoré urobíte budú prevedené v RAM-ke pod ROM-kou, a tam môžete urobiť veľké škody (napr. prepísať dôležité dáta).

5.6 Ukázanie obsahu registrov

Obsah registrov si môžete nechať ukázať kedykoľvek. Použite k tomu **REGISTROVÝ** príkaz:

Syntax: **R** (bez ďalších údajov)

Dostanete aktuálny stav registrov v tvare, ktorý sme dostali pri prvom volaní monitora (5.2). Budú ukázané obsahy týchto registrov: Program counter, Interrupt request, Status register, Accumulator, X, Y a Stack pointer. Nie je možné zmeniť stav registrov.

5.7 Assemblovanie a disassemblovanie

Monitor obsahuje vstavaný priamy assembler, takže je možné zadávať, testovať a ukladať krátke programy priamo v asembleri bez toho, aby ste museli naťahovať nejaký editor alebo assembler.

Na assemblovanie použijete **ASSEMBLOVACÍ** príkaz.

Syntax: **Axxxx mnemo (operand)**

xxxx – hexadecimálna štartovacia adresa

mnemo – symbol strojovej inštrukcie, napr. JSR

operand – priamo alebo adresný operand, ak je potrebný, zapísaný obvyklým spôsobom. Názvy, ako symboly operandu sa nedajú použiť.

Dávajte pozor, aby štartovacia adresa bola vždy 4-miestna, prípadne doplnená začiatočnými nulami a aby bola umiestnená priamo za A (bez medzery).

Akonáhle napíšete assemblovací príkaz, bude okamžite preložený a zapísaný v riadku tak, že pred mnemotechnickým symbolom príkazu bude v správnom poradí zapísaný operačný kód a operand (hexadecimálne) napr. u adries nižší/vyšší Byte.

Po zápise riadku s prvým príkazom bude do nasledujúceho riadku automaticky vypísaný príkaz A a adresa ďalšieho assemblerovského príkazu, takže ďalej stačí už len zadávať mnemotechnické názvy a operandy. Ak chcete assembler opustiť, stlačte jednoducho po zapísaní posledného mnemo kódu ešte jeden RETURN. Dostanete sa späť do monitoru.

Príklad: zapíšete tento krátky program

```
. A1000 LDA #$01  
. A1002 STA $0400  
. A1005 LDA #$07  
. A1007 STA $D800  
. A100A BRK
```

Výpis na obrazovke bude vyzerat takto:

```
. A1000 A9 01 LDA # $01
. A1002 8D 00 04 STA $0400
. A1005 A9 07 LDA # $07
. A1007 8D 00 D8 STA $D800
. A100A 00 BRK
. A100B
```

Odštartujte tento krátky program príkazom **G1000**. V ľavom hornom rohu obrazovky sa objaví žlté A. Pod zapísaným G-o príkazom sa objavia obsahy registrov. (Go-príkaz Gxxx bude popísaný v časti 5.8)

Pre disassemblovanie použite príkaz pre **DISASSEMBLOVANIE**.

Syntax: **D xxxx yyyy** alebo **D xxxx,yyyy** alebo **D xxxx**

xxxx – začiatková adresa

yyyy – koncová adresa (medzera za D možno vynechať)

Príklad: **D1000 100A** – disassemblovanie uvedenej oblasti pamäte

D1000 – vydá jeden príkaz z uvedenej adresy v assembleri

Príkaz pre disassembler má význam jedine vtedy, ak od štartovacej adresy skutočne začína program v strojovom kóde. V opačnom prípade sa síce disassembler pokúša previesť obsah pamäte na postupnosť assemblerovských príkazov, lenže výsledný program celkom iste nebude mať žiadny zmysel.

Zobrazenie disassemblovanej časti pamäte si ukážeme na nasledujúcom príklade (výňatok z malého programu vo vyrovnávacej pamäti datasetu). Štart disassembleru je príkazom **D 0382 0389**:

```
. , 0382 A0 00 LDY # $00
. , 0384 B1 14 LDA ($14),Y
. , 0386 AA TAX
. , 0387 29 7F AND # $7F
. , 0389 20 D2 FF JSR $FFD2
```

Rolovanie dopredu a dozadu je možné i v tomto móde. Pomocou kláves **F3**, **F5** alebo **CRSR** získate zobrazenie príľahlých susedných oblastí.

Pri disassembleri každý riadok začína znakmi **. ,** (bodka a čiarka). Ak prepíšete čiarku dvojbodkou prepnete monitor na výpis obsahu pamäte (časť 5.5). Prírodzene sa môžete vrátiť na disassemblovanie, ak znova prepíšete dvojbodku čiarkou.

Ak chcete ukončiť rolovanie obrazovky, prípadne celú zvolenú funkciu monitора bez toho, aby ste z monitора vystúpili, stlačte RETURN. Ak sa vám objaví výpis ďalšieho riadku, stlačte znova RETURN. Kurzor zostane stáť v nasledujúcom riadku za bodkou, ktorá uvádza príkazy monitора. Teraz môžete zadať príkaz monitора.

5.8 Spúšťanie a testovanie programu v strojovom kóde

Programy v strojovom kóde môžete odštartovať príkazom **Go**, o ktorom sme sa už krátko zmienili v predchádzajúcej časti. Akonáhle sa narazí v programe na inštrukciu BRK, program sa zastaví a monitor vám ukáže aktuálny stav registrov. Výpis registrov je rovnaký, ako pri štarte monitора, alebo po príkaze R (časť 5.2 a 5.6), ale začína s B*, miesto C*. Týmto nám monitor ukazuje, že behom programu narazil na bod prerušenia (breakpoint). Potom môžete zadať ďalšie príkazy monitора.

Syntax: **G xxxx**

xxxx – hexadecimálna štartovacia adresa, medzeru po G možno vynechať.

Tento príkaz je vhodný aj pre testovanie programov v strojovom kóde, ak ste vsunuli inštrukciu BRK. Programový čítač (program counter) ukazuje po dosiahnutí bodu prerušenia na adresu nasledujúcej inštrukcie, takže môžete priebeh programu sledovať alebo riadiť. Po otestovaní programu, môžete inštrukciu BRK odstrániť, alebo ak vzniknutý posuv je neprijateľný, nahradiť príkazom NOP.

5.9 Porovnanie oblasti pamäte

Pre porovnanie obsahu rôznych oblastí pamäte použijte príkaz **COMPARE**. Budú vypísané všetky adresy dopredu zadanej oblasti pamäte, v ktorých sa obsah líši od zadanej porovnáwanej oblasti. Ak sú oblasti identické, neobjaví sa žiadny výpis.

Syntax: **C xxxx yyyy zzzz** alebo **C xxxx,yyyy,zzzz**

xxxx – začiatková adresa

yyyy – koncová adresa oblasti, ktorá sa má porovnávať s rovnako veľkou oblasťou, ktorá začína na adrese zzzz. Medzeru za C možno vynechať.

Príklad: **C2000,3000,4000** porovnáva obsahy adries \$2000 až \$3000 včítane s obsahmi adries \$4000 až \$5000 včítane.

5.10 Vyplnenie oblasti pamäte jednotným obsahom

Pomocou príkazu FILL môžete zaplniť určenú oblasť pamäte vami definovaným jednotným obsahom.

Syntax: **F xxxx yyyy zz** alebo **F xxxx,yyyy,zz**

xxxx – začiatočná adresa

yyyy – koncová adresa oblasti pamäte, ktorú chcete zaplniť obsahom zz

zz – obsah (\$00–\$FF). Medzeru po F možno vynechať.

5.11 Prenos celého bloku pamäte

Príkaz TRANSPORT slúži na prenesenie obsahu určenej oblasti pamäte do inej, rovnako veľkej oblasti. Prekrývanie adresy vysielacej a prijímacej oblasti je dovolené, vedie však podľa okolností k nežiadúcim výsledkom.

Syntax: **T xxxx yyyy zzzz** alebo **T xxxx,yyyy,zzzz**

xxxx – začiatočná adresa

yyyy – koncová adresa vysielacej oblasti

zzzz – začiatočná adresa prijímajúcej oblasti. Medzeru po T možno vynechať.

Príklad: T0000,03E8,0400 – Vyskúšajte si tento príklad. Po stlačení RETURN-u sa objaví na obrazovke „smetie“. Zaplnili sme totiž pamäť obrazovky od adresy \$0400 obsahom Bytov \$0000 až \$03E8. Ilustračná akcia, ktorá nemá zmysel.

Prekrývanie medzi vysielacou a prijímacou oblasťou môže mať zmysel napr. vtedy, ak potrebujete získať oblasť pamäte, v ktorej sa pravidelne za sebou opakujú rôzne obsahy Bytov. Zaplníte napr. Byty \$3000 až \$3007 hodnotami \$F0, \$F1, \$F2 atď. až \$F7. Teraz zadajte príkaz \$T3000,3FFF,3008. Akonáhle sa adresy \$3008 až \$300F zaplnia obsahom adresy \$3000 až \$3007, stanú sa pre pokračovanie príkazov adresami vysielacími. To sa potom opakuje tak dlho, kým nie sú všetky adresy v rozsahu \$3008 až \$3FFF zaplnené opakujúcou sa postupnosťou \$F0 . . . \$F7.

5.12 Hľadanie určitého obsahu v pamäti

Pomocou príkazu HUNT môžete hľadať v zadanej oblasti pamäte postupnosť 1 až 8 Bytov dlhú s určitým obsahom. Na obrazovke sa vypíše vždy začiatočná adresa, od ktorej sa hľadaná postupnosť nachádza. Ak sa hľadaná postupnosť v zadanej oblasti nenachádza, nevypíše sa na obrazovke nič.

Syntax: **H xxxx yyyy aa** alebo **H xxxx,yyyy,aa** alebo
H xxxx yyyy aa cc ... ii alebo **H xxxx,yyyy,aa,bb,cc,...,ii** alebo
H xxxx yyyy "znakový reťazec"

xxxx – začiatočná adresa

yyyy – koncová adresa prehľadávanej oblasti

aa, bb, cc, ..., ii – hexadecimálne hodnoty alebo reťazec znakov zapísaný v úvodzovkách. Medzeru po H možno vynechať.

Príklad: **H 8000 9FFF 41** – každý Byte oblasti \$8000 až \$9FFF bude preverený, či nemá hodnotu \$41 (písmeno A)

H C000 CFFF "FINAL" – prehľadaná bude oblasť \$C000 až \$CFFF.

V prípade, že bude hľadaný reťazec "FINAL" nájdený, vypíše sa na obrazovke začiatočná adresa, kde je reťazec uložený.

5.13 LOAD a SAVE programu

Monitor vám dáva možnosť natiehnúť program z disku alebo kazety do ľubovoľnej voľnej oblasti RAM-ky. Tak isto môžete program z ľubovoľnej oblasti pamäte uložiť na disk alebo kazetu. Takto je možné pohodlne uchovávať programy v strojovom kóde z oblasti mimo Basic-RAM, u ktorých sa normálne nedá použiť SAVE (DSAVE).

LOAD programu:

Syntax: **L "Názov",ga** alebo **L "Názov",ga,xxxx**

Názov – meno programu

ga – dvojmiestna hodnota určujúca zariadenie (01–kazeta, 08..0F disk)

xxxx – začiatočná adresa pamäte, kam má byť program uložený.

Medzeru po L možno vynechať.

Príklad: **L "SORT",01** – program SORT bude natiehnutý z kazety do miesta, odkiaľ bol sejtovaný.

L "QUICK",08,C000 – program QUICK bude natiehnutý z disku od adresy \$C000. Ak bol pôvodne sejtovaný z iného miesta, musí byť programovaný relokatívne, inak nebude fungovať.

SAVE programu:

Syntax: **S "Názov",ga,xxxx,yyyy**

Názov – meno programu

ga – dvojmiestna hodnota určujúca zariadenie

xxxx – začiatočná adresa

yyyy – koncová adresa oblasti pamäte, v ktorej je program, ktorý sa má uložiť.

POZOR! Koncová adresa (yyyy) musí byť o 1 vyššia, než je posledný Byte obsadený programom. Musí to byť prvý voľný Byte.

Príklad: S"DUMP",08,033C,03EA – program DUMP bude uložený na disk z oblasti \$033C až \$03EA (kazetová vyrovnávací pamäť).

5.14 Prevod hexadecimálnych čísel na decimálne a opačne

Prepočítavanie čísel z jedného číselného systému do druhého je veľmi jednoduché:

Z hexadecimálneho na decimálne:

Syntax: \$xxxx

Príklad: \$C000 – objaví sa \$C000 = #49152

Z decimálneho na hexadecimálne:

Syntax: #d

Príklad: #45054 – objaví sa #45054 = \$???? ...

xxxx – hexadecimálne číslo

d – decimálne číslo 1 až 5 miestne

Vyskúšajte si to sami. Prirodzene, že sa neobjaví ????, ale platné hexadecimálne číslo.

5.15 Presmerovanie výstupu z obrazovky na tlačiareň

Pomocou príkazu PRINT môžete určiť, že výpisy nepôjdu na obrazovku, ale na tlačiareň s adresou zariadenia 4. Na obrazovke sa takto neobjaví nič. Toto opatrenie má samozrejme zmysel jedine vtedy, ak monitorské príkazy, ktoré budú následne zadané majú normálne výstup na obrazovku (B, D, M, R atď.)

Syntax: P (bez ďalších údajov)

Následujúce príkazové riadky už nebudú uvádzané bodkou. Zadávať ďalšie príkazy monitora do 1 stĺpca obrazovky.

Ak chcete mať znova výstup na obrazovku, zadajte znova príkaz P. Nasledujúci príkazový riadok bude opäť uvádzaný bodkou a výstup na tlačiareň bude odpojený.

5.16 Opustenie monitora

Monitor opustíte pomocou príkazu EXIT. Dostanete sa späť do Basic-u.

Syntax: X (bez ďalších údajov)

V prípade, že ste mali monitor prepnutý na výstup na tlačiareň, bude všetko prerušené a výstup bude na obrazovku.

Príkaz X môže byť použitý v ktoromkoľvek riadku v stĺpci 1 alebo 2 a to i v tom prípade, že zvyšok riadku nebude prázdny.

6. DISKOVÝ MONITOR

Diskový monitor je len časť monitoru strojového kódu. Jeho funkcie sa volajú priamo zo strojového monitoru. Preto môžete nájsť jeho funkcie už v prehľade funkcií strojového monitora (5.1)

Vo vzťahu k adrese zariadenia sa chovajú príkazy diskového monitora tak isto, ako diskové príkazy v móde FC (4.3.2). Ak je spojených viac mechaník (viac diskových jednotiek), adresa 186 na stránke 0 určuje, ktorá mechanika bude zvolená.

6.1 Prístup monitoru k RAM a ROM disku

Prepínanie oblasti pamäte disku sa robí podobne ako Bankswitching príkazom **OD**. Čo sa týka adresy, platí to, čo bolo popísané vyššie.

Aj keď procesor 6502 v mechanike 1541 vie adresovať 64kB, môžete mať prirodzene prístup len k reálne existujúcim oblastiam pamäte:

- 2kB RAM od \$0000 – \$07FF (0 – 2047)
- 16kB VIA 1 (522 Bus control) od \$1800 – \$180F (6144 – 6159)
- 16kB VIA 2 (6522 Bus control) od \$1C00 – \$1C0F (7168 – 7183)
- 16kB ROM (operačný systém) od \$C000 – \$FFFF (49152 – 65535)

Akonáhle sa pokúsite o prístup k iným oblastiam pamäte, dostanete na obrazovke nepravdivý údaj.

Syntax: **OD** (bez ďalších údajov)

Väčšinu príkazov strojového monitora môžete použiť podobne ako pri operáciách s pamäťou u C-64 alebo C-128 s rovnakým účinkom. Napriek tomu platia nasledujúce obmedzenia:

- ukázanie obsahu registra príkazom R platí len pre C-64/C-128
- prevod výstupu na tlačiareň príkazom P nie je možný
- príkaz G pre štart strojového programu je nepoužiteľný
- v príkazoch L a S (Load a Save) nájdú uplatnenie len adresy pamäte C-64/C-128
- pre čítacie a zapisovacie príkazy pre disk, ktoré budú popísané ďalej platí to isté. Pokiaľ sa príkazy budú týkať vyrovnávacej pamäte v 1541, príkazy sa vyplnia, výsledok je však nepoužiteľný.

Môžete prirodzene kedykoľvek pomocou príkazu O0–O7 pre Bankswitching prepnúť späť pamäťové rozsahy počítača a takto tieto obmedzenia zrušiť.

6.2 Čítanie a zápis sektoru

Pomocou týchto príkazov môžete pracovať priamo so sektormi na diske. Dávajte ale pozor na to, že pri zápise sektoru sa nerobí test na obsadenosť sektoru a nerobí sa zmena v directory. O to sa musíte postarať sami, čo vyžaduje dokonalú znalosť organizácie diskety.

Syntax:

Čítanie: *R tt ss mm alebo *R,tt,ss,mm

Zápis: *W tt ss mm alebo *W,tt,ss,mm

tt – číslo stopy (track)

ss – číslo sektoru obe v hexadecimálnej forme

mm – vyšší Byte (číslo stránky) v ktorej má byť čítané (písané)

Príklad: *R 12 04 CE – týmto príkazom bude prečítaný obsah 4. sektoru 18. stopy (tieto čísla sú decimálne) do oblasti pamäte \$CE00 – \$CEFF

6.3 Ukázanie stavového, prípadne chybového hlásenia disku

Syntax: @ (bez ďalších údajov)

Po zadaní príkazu a stlačení RETURN sa na obrazovke objaví hlásenie stavu disku, v prípade chyby (červená LED bliká) potom jej príčina. S touto činnosťou spojené čítanie chybového kanálu ukončí blikanie LED-ky. Príkaz pôsobí rovnako ako v móde FC príkaz DOS" bez ďalších údajov.

6.4 Vyslanie diskového príkazu

Syntax: @ s bezprostredne nasledujúcim diskovým príkazom

Tento príkaz pracuje rovnako ako v móde FC vyvolávaný príkazom DOS" s doplňujúcimi údajmi (4.3.2)

Príklad: N:TESTDISK,T1 – naformátuje disketu, ktorá bude mať meno TESTDISK a ID "T1"

6.5 Opustenie diskového monitora

Pretože diskový monitor je súčasťou strojového monitora, opustíte ho tiež príkazom X a dostanete sa do Basic.

Ak ste pracovali s RAM/ROM disku a chcete teraz opäť pracovať s pamäťou počítača, zadajte jeden z príkazov 00–07 pre Bankswitching. Teraz máte k dispozícii zase celý rozsah inštrukcií strojového monitoru.

7. RIADENIE TLAČIARNE A PARALELNÝ INTERFACE CENTRONICS

Skôr, ako urobíte funkčný test s pripojenou tlačiarnou, presvedčte sa ešte raz, či máte všetky potrebné zariadenia prepojené. Pri práci s FC nesmie byť načítaný žiadny program pre prácu s paralelným interface. Ak ale potrebujete takýto program použiť, musíte FC odpojiť povelenom KILL. Podobne treba postupovať aj v prípade hardwerových interface.

Niektorí výrobcovia soft a hardweru predpisujú používať čísla zariadení a sekundárne adresy, ktoré sa od štandardu Commodore a od dohôd pre používanie FC líšia. Preskúšajte si z tohoto hľadiska vaše programy, inak môžu pri používaní vzniknúť problémy.

7.1 Sekundárne adresy

Majitelia tlačiarní Commodore (napr. MPS801/803) majú k dispozícii sekundárne adresy 0 a 7, pomocou ktorých môžu zvoliť sadu znakov (bloková grafika, veľké/malé písmená).

Ak je tlačiareň s interface CENTRONICS pripojená na USER PORT, používajte iné sekundárne adresy, aby ste mohli používať tlačiareň v móde ASCII a COMMODORE. FC podporuje tieto sekundárne adresy:

0 – vyhradené len pre tlačiarnie Commodore (MPS801/803 atď.), ktoré sú pripojené na sériovú zbernicu. U tlačiarní s paralelným interface na User Porte funguje ako sekundárna adresa 3.

1 – ako adresa 0.

2 – prekladá všetky špecifické Commodore kódy pre písmenové znaky (nie blokovú grafiku a riadiace kódy) do kódov ASCII, takže sa dajú tieto znaky tlačiť na tlačiarniach s ASCII sadou znakov. Riadiace kódy sú odfiltrované. Bloková grafika, ktorú môžete získať len cez klávesu CBM bude vo väčšine prípadov prekódovaná na šikmé znaky (kurzivu).

3 – posiela všetky znaky bez filtrácie na tlačiareň. V odbornej literatúre sa tomu hovorí „transparentný mód“ alebo „priamy kanál“. Pomocou sekundárnej adresy 3 môžete pohodlne vysielat všetky riadiace kódy, s ktorými tlačiareň spolupracuje. Kódy sa dočítate v príručkách k tlačiarniam. Pri vysielaní riadiacich znakov na tlačiareň používajte typ CHR\$ napr.: PRINT#x,CHR\$(...);CHR\$(...);..., a nie PRINT#x,".....".

7 – vyhradené len pre tlačiarnie Commodore, pripojené na sériovú zbernicu. Pri tlačiarní s paralelným interface má rovnaký účinok ako adresa 2.

9 – umožňuje vám využívať a zobrazovať na tlačiarniach typu EPSON a kompatibilných (s ESC-P-štandardom pre riadiace kódy) všetky Commodorovské grafické a riadiace kódy. Táto sekundárna adresa je veľmi vhodná pre listing a výpisy podobné, kde sa vyskytujú grafické znaky Commodore.

Pretože je pri tom potrebné prepnúť na grafiku podľa bitových vzorkov, môžu vzniknúť aj iné šírky znakov, ktoré dovoľujú tlač 60 alebo 120 znakov na riadok. Tlač veľké/malé písmená je závislá na tom, ako máte nastavený výstup na obrazovku. Pri tlačiarňach, ktoré vedia tlačiť grafiku sa dá pomocou POKE 56332 (\$DC0C) ovplyvniť šírku znakov. Táto adresa pracuje so špeciálnym registrom v CIA#1, ktorým sa riadi sériovo-paralelný prevod. Potrebné kódy pre riadenie hustoty tlače pre vašu tlačiareň sa dozviete z príručky k tlačiarňi (kapitola s bitovou grafikou). Niekoľko typov, ktoré platia pre bežné typy tlačiarňi:

úzke písmo (120 znakov na riadok) POKE 56332,76

stredne široké písmo (72 znakov na riadok) POKE 56332,53

široké písmo (60 znakov na riadok) POKE 56332,0

Niektoré POKE odpojujú v tlačiarňi grafický mód, takže spôsobia stav, akoby bola zvolená sekundárna adresa 2 alebo 3.

10 – to isté ako 9, lenže všetky znaky sú negatívne.

Použitie iných sekundárnych adries vedie k výsledkom, ktoré je ťažko vopred odhadnúť.

7.2 Vstavaný interface CENTRONICS

Akonáhle FC zistí, že na User Porte je pripojená tlačiareň s interface CENTRONICS, bude túto tlačiareň oslovovať ako zariadenie číslo 4. Ak zároveň používate ešte ďalšiu tlačiareň na sériovej zbernici, nastavte ju ako zariadenie 5.

Funkciu tlačiarne, pripojenej na User Port preskúšajte nasledovne:

OPEN1,4,4:CMD1 a stlačte RETURN

Tlačiareň by mala vypísať "READY". Ak miesto toho dostanete hlásenie "DEVICE NOT PRESENT", vyskúšajte tlačiareň inicializovať cez register, o ktorom bola reč pri sekundárnej adrese 9:

tlačiareň EPSON POKE 56332,128

tlačiareň Smith-Corona POKE 56332,53

Aj teraz budete musieť podľa okolností skúšať, kým nájdete hodnotu, vhodnú pre vašu tlačiareň. U väčšiny tlačiarňi ale funguje štandardné nastavenie registra, ktoré je normálne 0.

Ak sa tlačiareň ohlásí "READY", potom ste našli vhodnú hodnotu pre POKE 56332.

Teraz zadajte: PRINT#1,"TOTO JE TEST" a stlačte RETURN

Teraz by mala tlačiareň vytlačiť text v úvodzovkách včítane prípadných commodorovských grafických značiek. Takto môžete na tlačiarňi získať listing programov, za predpokladu, že program je v BASIC. Uzatvorenie kanálu pre tlačiareň: PRINT#1:CLOSE1

POKE 56332 ostane zachovaný, kým jeho hodnotu neprepíšete inou hodnotou, nevymažete pomocou hard reset-u alebo nevypnete počítač. Doporučuje sa dôkladne preštudovať príručku k tlačiarni. Môžu byť v nej zvláštnosti pre ovládanie vašej tlačiarne.

7.3 Odtlačok (hardcopy) obrazovky

Pri tlači grafickej alebo textovej obrazovky ako hardcopy treba plnú šírku papiera (80 normálne širokých znakov – približne 8" – asi 20 cm). Pre jeden obrázok je treba asi polovicu stránky A4. Viacfarebné obrázky budú prevedené do rôznych odtieňov šedej farby.

Funkciu pre odtlačok obrazovky vyvoláte cez FREEZER menu, ktoré vyvoláte stlačením tlačidla FREEZE na „Final Cartridge“. Ďalšie kroky nájdete v kapitole 3, v ktorej je postup detailne popísaný.

– Tlačiarne s Centronics interface: Centronics interface „Final Cartridge“ je založený na EPSON štandarde pre riadiace kódy. Pre grafiku s vysokým rozlíšením treba Bitovú mapu s najmenej 960 bodmi (grafika bitového vzorku). Všetky moderné tlačiarne ju majú. Dokonca aj tlačiarne konfigurované pre IBM grafiku pracujú normálne, pretože „FC“ vysielala iba riadiace kódy.

Odstup riadkov je nastavený na N/216, kde N=17.

Prepnite DIP prepínač „7bit/8bit“ do polohy „8bit“.

– Tlačiarne Commodore 801 a 803 vedú tiež vytlačiť grafiku s vysokým rozlíšením na rôzne stupne šede. To však neplatí pre tlačiarne typu 802 a 1526, pretože tie nemajú k dispozícii bitovú mapu.

Pokiaľ sa pri hardcopy objavia problémy, mohlo by to byť tým, že tlačiareň je v nevhodnom výstupnom móde. Použite funkciu Printer reset z Print Options menu (klávesa F5), čím privediete tlačiareň do základného nastavenia. V núdzi tlačiareň na krátko vypnete. Takto dostanete tlačiareň do základného nastavenia celkom určite.

7.4 Tlačiareň ako elektronický písací stroj

Pomocou príkazu TYPE zmeníte váš počítač, spojený s tlačiarnou na elektronický písací stroj. To je vhodné obzvlášť v prípadoch, keď potrebujete napísať len krátku správu, ktorú nechcete uchovať, a pre ktorú sa nevypláti štartovať textový editor.

Obsluha je veľmi jednoduchá. Zadajte:

OPEN1,4,2 TYPE (pre tlačiarne Centronics), alebo

OPEN1,4,7 TYPE (pre tlačiarne Commodore)

Teraz môžete písať text. Akonáhle stlačíte RETURN, vytlačí sa celý riadok. Pritom môže byť samozrejme tlačенý riadok širší, než obrazovka.

Predtým, než tlačový riadok odošlete, môžete ho samozrejme editovať pomocou kláves CRSR a INST/DEL.

Odoslať môžete ľubovoľný počet textových riadkov. Každý vstup bude pojatý ako text. Preto nie je možné zadávať žiadne príkazy, pokiaľ ste v móde TYPE.

Ak chcete TYPE modus opustiť, stlačte tlačidlo RESET na „FC“. Dostanete potom normálny obrázok ako po zapnutí a ste v Basicu. „Final Cartridge“ je opäť aktívny.

TYPE modus môžete ukončiť i pomocou resetu, vyvolaného z FREEZE menu, do ktorého sa dostanete stlačením tlačidla FREEZE. Tak zabránite tomu, aby mal systémový reset vplyv na periférne zariadenia.

8. VYUŽITIE VŠETKÝCH OBLASTÍ RAM POD ROM

Final Cartridge dovoľuje aj prístup k oblastiam RAM-ky, ktoré sú ukryté pod oblasťou ROM. Tabuľka pamäte v časti 5.4 (Bankswitching) vám ukazuje, o ktoré oblasti pamäte ide.

Tieto oblasti môžete využívať ako prídavnú pamäť pre odkladanie dát, ako sú premenné, text, grafický obsah obrazovky, tabuľky atď. a pohodlne ich obsluhovať z Basic.

Tieto prídavné 24kB RAM na adresách \$A000 až \$FFFF, ktoré sú normálne len nepohodlne dostupné pomocou POKE a PEEK vám sprístupnia dva ďalšie príkazy Basic:

MR (memory read) a MW (memory write)

Oba príkazy nie sú prístupné okamžite po zapnutí FC, ale až po ich inicializácii. To sa robí pomocou inštrukcie SYS\$DFC0 (SYS57280 decimálne). Možnosť, aby oba príkazy boli okamžite k dispozícii po zapnutí FC nebola využitá z dôvodu kompatibility s doterajším softverom. Po inicializácii sú príkazy k dispozícii po celú dobu, kým nezadáte RESET alebo nevypnete počítač.

Príkazy pracujú takto:

MR číta 192 Bytov a pre ich medziuloženie používa vyrovnávaciu pamäť datasetu od adresy \$033C (828 decimálne)

MW prečíta 192 Bytov z vyrovnávacej pamäte datasetu a uloží ich na iné miesto.

Nasledujúci krátky program by to mal vyjasniť:

```
10 SYS $DFC0
20 FOR N=1024 TO 2024 STEP 192
30 MR N
40 MW 60000+N
50 NEXT N
```

Vysvetlenie: inicializácia inštrukcií MR a MW, slučka v skokoch po 192 byte, čítanie pamäte obrazovky v blokoch po 192 bytoch, zápis blokov do pamäte nad adresu 60000.

Po dobehnutí tohto programu je celá časť pamäte obrazovky uložená v inej časti pamäte, začínajúc adresou 61024. Aby sme si ukázali, že akcia bola úspešná, uložíme pôvodný obsah obrazovky na správne miesto. Vymažte obrazovku pomocou SHIFT/CLR a zapíšte tento program:

```
10 PRINT CHR$(147)
20 FOR N=1024 TO 2024 STEP 192
30 MR 60000+N
40 MW N
50 NEXT N
60 POKE $D021,0
```

Vysvetlenie: vymazanie obrazovky, čítanie blokov 192 byte dlhých z RAM pod ROM, zápis do obrazovkovej pamäte, nastavenie farby na čiernu.

Nasledujúci program má ukázať, ako sa dá využiť inštrukcia MR a MW k uloženiu nejakej oblasti pamäte do premennej:

```
10 SYS $DFC0
20 BUFFER$=""
30 X=PEEK($2E)*PEEK($2D)
40 POKE X+2,192
50 POKE X+3,60
60 POKE X+4,3
70 MR $0400
80 A$=BUFFER$
90 PRINT CHR$(147);
100 FOR N=1 TO 192
110 POKE 1023+N,ASC(MID$(A$,N,1))
120 POKE 55295+N,1
130 NEXT N
```

Vysvetlenie: Inicializácia MR a MW, vymazanie reťazcovej premennej, v X je začiatok pamäte premenných v Basic, nastavenie dĺžky reťazca na $3*256+60=828$ (vyrovnávacia pamäť datasetu), 192 bytov od adresy \$0400 kopírovať do reťazcovej adresy, obsah vyrovnávacej pamäte do A\$, vymazať obrazovku, A\$ čítať po znakoch a obsah ukázať na obrazovke, 192 bytov pamäte farieb na bielu.

9. ČO FREEZER NEOVLÁDA

V niekoľkých prípadoch môže FREEZER zlyhať, napríklad pre tieto príčiny:

1. Pokúšate sa aktivovať FREEZER z monitoru.
2. Program má ochranu pred kopírovaním, kde sa využíva zvukový čip (6581 SID). Pretože tento čip má k dispozícii len zapisovacie registre a niekoľko čítacích registrov, nie je možné získať uložené hodnoty určitého programového stavu a rekonštruovať ich.
3. Program načíta pri svojej činnosti ďalšie programové časti. V tomto prípade musí byť k dispozícii originálna disketa, ak „zmrazený“ a opäť aktivovaný program požaduje čítanie ďalších častí.
4. Program spôsobuje intenzívnu činnosť oboch prerušovacích časovačov v čípe 6526 (CIA 1 a CIA 2). Tak napr. pri hlasovom výstupe nie je možné pokračovanie programu, lebo pre veľkú rýchlosť časovača sa nedá program bezchybne uložiť.
5. Ak je nejaký obrázok postavený z viacerých prerušovacích rastrov, môže sa stať, že na prvé použitie FREEZER-u nedostanete obrázok, aký si prajete. Skúste „zmrazenie“ na inom mieste, kým obrázok nezodpovedá vašim požiadavkám.
6. Mnohé, komerčne rozširované programy, chránené pred kopírovaním majú svoje vlastné zavádzacie rutiny, ktoré sa nedajú obísť. FC sa pokúša vždy nasadiť svoje zavádzacie rutiny, ale tam, kde to nie je možné, predáva riadenie pomalšiemu, originálnemu zavádzaču. Kópie, vytvorené pomocou FREEZER-u budú ale zavádzané turbo rýchlosťou.
7. Z originálnej kazety môžu byť zakúpené programy zavádzané jedine normálne. Ak si urobíte RESAVE pomocou FREEZER-u, budete pracovať aj tu rýchlejšie.

Dúfame, že tento popis postačí užívateľom na oboznámenie sa s funkciami Final Cartridge 2. Keďže je v praxi veľmi rozšírená, dúfame, že skúsenosti z používania odovzdáte aj iným. Ak niekto má vedomosti o ďalšom využití o ktorom sme sa v popise nezmienili napíšte na adresu Milan Bobula, Olbrachtova 10, 040 01, Košice.

V roku 1990 Vám všetko najlepšie praje

Commodore klub Košice

Obsah čísel 1–7 Spravodaja

Obvody LSI pre Commodore 64	3
Turbo tape 61k	4
Používanie súborov	6
Úprava magnetofónu pre odposluch	10
Tipy a triky	12
Univerzálny paralelný interface	13
Disková jednotka VC-1541	18
Softverový odposluch Datasetu	21
Tipy a triky	22
Jednoduchý hardverový odposluch	24
Princíp činnosti kompilátorov	24
Využívanie kompilátorov pri tvorbe softveru	26
Sériový interface	30
Conrad Electronic	36
Číselné systavy	41
Tipy a triky	44
Profi-Ass 64	45
Profi-Mon 64	47
Debugger 64	48
Colossus 2 (1)	49
Grafický editor EGA V3.2	53
Colossus 2 (2)	62
Myš pre C-64/128	63
Tlačiareň MPS 1250	66
RE-NEW rutina pre C-16, +4	68
Multigraphic pre C-16, +4	69
Nastavenie hlavy v datasette	70
Dragobasic	71
Tipy a triky na C-16, +4	74
Profi Assembler 64	75
Programovacie jazyky	77
Celoštátne stretnutie C-16, 116, +4	82
Complex Interface Adapter CIA 6526	84
Ochrana programov pre C+4	97
Obsluha tlačiarne D-100 pre C+4	97
Čo dokážu C-64, Atari ST, Amiga a PC	101
Celoštátne stretnutie užívateľov C-64	107
Final Cartridge (1)	108
Final Cartridge (2)	122

Spravodaj COMMODORE č. 7/1989 pre mikroelektroniku a výpočtovú techniku.

Vydáva: Krajský dom pionierov a mládeže – Commodore klub, Malinovského 2, Košice.

Vedúci redaktor: Juraj Rusnák. Odborní redaktori: Milan Bobula, Ing. Daniel Gábor, Ing. Zoltán Rábek, Pavol Žalobín.

Vydávanie povolil VS KNV – odbor kultúry, ev. číslo 25/1988.

Tlač: Východoslovenské tlačiarne z. p., Košice.