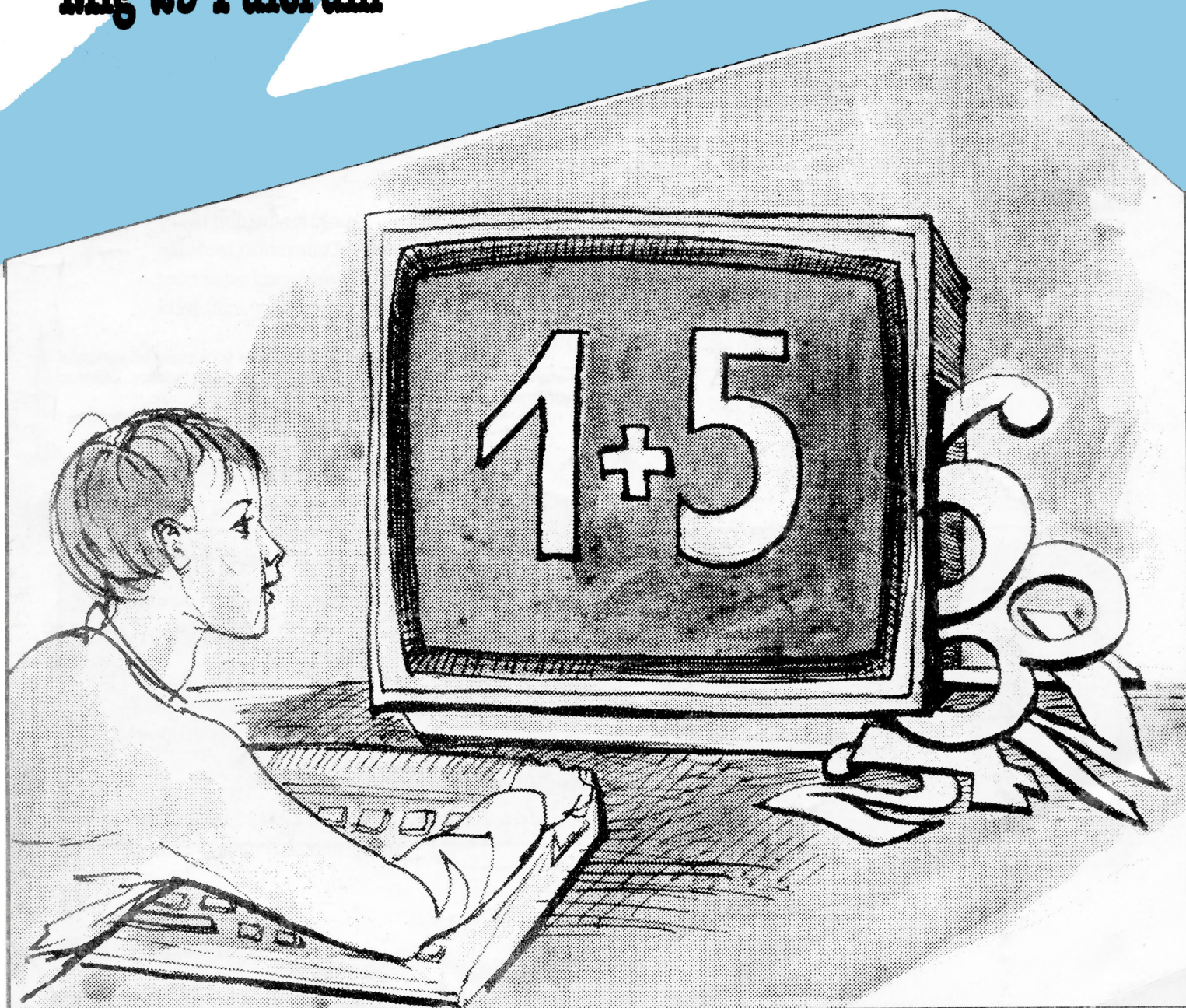


# amiga

měsíčník pro uživatele počítačů amiga

20 Kčs - 4/92

**Dabaze a Maxiplan Plus**  
**Safe Hex International**  
**Genlock A8802**  
**Mig-29 Fulcrum**





# Vážení čtenáři,

### podnikatelé, výrobci, obchodníci, autoři.

Těší nás Váš zájem o náš časopis, který je poznat ze zvyšujícího se počtu předplatitelů a též z Vašich dopisů.

Jak jste si již jistě všimli, začíná se na našem trhu objevovat i tuzemský soft i hardware a naše redakce se těmto věcem podrobně věnuje. Máte-li zájem, aby se o Vašich výrobcích dozvěděli i ostatní čtenáři časopisu, informujte nás o Vaší produkci. Poskytnete-li nám Vaše dílo k testování, podle jeho povahy vyhledáme odborníky a s jejich recenzemi seznámíme čtenáře.

Nezáleží na tom, zda jste přímo autoři, nebo obchodníci s

dovezeným zařízením. Jediné nebezpečí, které se za tím skrývá, je to, že recenzi uveřejníme, i když nebude příliš optimistická. To je rozdíl od placené reklamy. Zájem zákazníka je prvořadý. Výrobek k testování (program či hardwareový doplněk k Amize), by měl být v takovém stavu a balení, jak se bude i prodávat, abychom mohli posoudit i to, zda nepříliš technicky zdatní uživatelé jeho obsluhu a instalaci zvládnou bez problémů. Každý produkt získá i bodové ocenění, na které jste zvyklí u her a používá se i v ostatních odborných časopisech.

*Tomáš Stibor*

Hledáme externí spolupracovníky ke spolupráci při vydávání našeho časopisu. Písemné nabídky zašlete na adresu redakce. Zájemce pozveme k osobnímu podrobnějšímu jednání.

Redakce

### IMPRESSUM

(c) 1992 "amiga - měsíčník pro uživatele počítačů amiga." Nevyžádané příspěvky se nevracejí.

Ročník III. Cena 20 Kčs. Reg. č. MK ČR 5281. MČ 46067.

Vydavatel: Jiří Próza

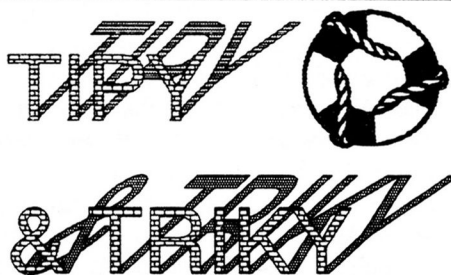
Šéfredaktor: Tomáš Stibor Jazyková úprava: Dr. Daniela Háková

Obálka: Iva Fialová Layout: DANSY Tisk: PRINTSERVIS

Podávání novinových zásilek povoleno Ředitelstvím pošt Praha č.j.10183 ze dne 25.9.1991

Redakce: Balská 29, 120 00 Praha 2 Inzerce, informace a předplatné: Balská 29, 120 00 Praha 2





## Orchestr místo sólistů

Když využíváte možností MIDI programu SONIX (V1.3), bude vám pravděpodobně vadit, že při 4 MIDI-hlasech E, F, G, H nemohou být použity různé nástroje. Problém spočívá v tom, že SONIX hlasy E, F, G a H vysílá všechny na jeden kanál. Jak snad víte, mohou být předurčené instrumenty SONIXU měněny. Aby nyní mohly být použity různé instrumenty, musíme 4 hlasy E, F, G a H uložit na rozdílný MIDI-kanál. K tomu natáhneme instrument "MIDIpatch" a přejdeme do instrumentového editoru. V okně, které se nyní objeví, klikneme na pole "MIDI-Channel". Na to se ukáže číslicí "2", což má následující význam:

Obsadí-li se hlas F tímto nástrojem, bude vyslán na kanál 2. Tento instrument nyní uložíme např. pod jménem MIDI2. Znovu klikneme na pole MIDI-Channel a uložíme instrument ještě jednou, tentokrát pod názvem MIDI3. Tento postup opakujeme znovu, dokud pro každý kanál není jeden vlastní instrument. Pro všechny případy musíme dbát, aby bylo na disketě dost místa. Tyto operace provádějte jen na bezpečnostní kopii.

Na závěr můžeme např. hlasu E přiřadit instrument MIDI1, hlasu F nástroj MIDI2, k G MIDI3 a k H MIDI4. Na klávesnici, případně syntetizátoru, se pak odpovídající kanály (v našem případě 1 a 4) přiřadí různým instrumentům.

## Jde to také bez EXECUTE

Dovedeme si živě představit, jak vám leze na nervy, když při provedení každého dávkového souboru (Batch file) musíte pokaždé napsat:

**execute <jméno\_souboru> (parametry)**

Zadávání tohoto příkazu před jménem dávkového souboru však můžeme pomocí Workbench (od verze 1.3) vynechat. Vytvořte si textový soubor s následujícím obsahem:

echo "Ahoj, já jsem batch!"

echo "ale teď už končím"

Uložte tento sled příkazů pod jménem TEST a na závěr zadejte:

**protect TEST s add**

Nyní můžete jednoduše napsat jméno souboru a ten bude proveden. Když je nastaven bit "s", může být tedy příkaz execute ušetřen, neboť ještě před tím, než je soubor spuštěn, Amiga přezkouší, zda je tento bit nastaven. Pokud

ano, systém rozezná, že se jedná o příkazový soubor a provede jej. Tím je také jasné, že bit "s" smí být nastaven jen u souborů dávkových. V opačném případě je vydáno chybové hlášení.

## Zprava zarovnané psaní

Následující sled příkazů v AMIGA-BASIC slouží k vytištění 10 po sobě následujících řetězců:

```
FOR i = 1 TO 10
  PRINT a$(i)
NEXT i
```

Výsledek bude ale vydán ve formě doleva zarovnaného tisku, což v mnoha případech nevyhlíží právě nejlépe, např. když se jedná o řetězce doprovázené číslicemi. To lze vyřešit zarovnáváním doprava, přičemž zde nehraje žádnou roli, jak jsou řetězce dlouhé.

```
FOR i = 1 TO 10
  LOCATE 1,40-LEN(a$(i)) REM pravý okraj na 40
  PRINT a$(i)
NEXT i
```

číslice 40 ve druhém řádku určuje pravý okraj. Text se také nechá vycentrovat. Zde uvedeme následující příklad:

```
FOR i = 1 TO 10
  LOCATE 1,50-LEN(a$(i))/2
  PRINT a$(i)
NEXT i
```

S těmito instrukcemi budou všechny řetězce vytištěny centrovane, přičemž 50 určuje střed výstupu. V GFA-Basicu instrukce LOCATE přirozeně odpadne, namísto toho bude použito PRINT AT(x,y)a\$(i).

## SEKA tiskne

Kdo chce se SEKA-assemblerem vytisknout zdrojový kód, může pomocí příkazů:

```
>>> <RETURN>>
filename >prt:
```

převést obrazovkový výstup na tiskárnu. Následně bude dalším příkazem:

```
p n
```

vytisknuto n řádků zdrojového kódu a to od řádky aktuální. Poté se musí tiskový výstup znovu obnovit zadáním:

```
>>> <RETURN>>
```

(bez jména souboru), aby došlo k vyprázdnění tiskového bufferu a tisk mohl být ukončen. P.Š



# MAXIPLAN PLUS

## a práce s databázemi

*Jeří Anděl*

Ve svém článku o programu Maxiplan v tomto časopise jsem se záměrně vyhnul popisu práce s databází a makrojazykem. Článek totiž vysvětloval práci jednotlivých položek menu a v této formě by funkce těchto dvou nadstavb vlastního programu nebyly pochopitelné. Protože použití databáze a makrojazyka Maxiplanu silně umocňuje možnosti tohoto vynikajícího spreadsheetového programu, vracím se k těmto funkcím ve dvou samostatných článcích. Nejprve tedy o práci s databázemi. Pokud budu v tomto článku mluvit o položce menu, bude se to vždy týkat menu Data!

### 1. Založení oblasti databáze

Databázi můžete umístit na kterémkoliv místě worksheetu. Skládá se minimálně ze dvou sloupců. Jednotlivé sloupce mohou obsahovat čísla, texty nebo vzorce (zde je použita právě aktuální hodnota buňky). Jedinou a hlavní podmínkou je, aby první řádek každého sloupce začínal textovým řetězcem (návěští-label). Tento text v záhlaví každého sloupce by měl zcela logicky charakterizovat jeho obsah a stává se tak vlastně titulkem sloupce. Pokud potřebujeme jako návěští číslo, musíme je zadat jako textový řetězec; to znamená, že při zadávání musíme začít apostrofem! V jedné databázi nesmí být nikdy dva sloupce se stejným návěští.

Příklad 1.

|   | A            | B         | C         | D    |
|---|--------------|-----------|-----------|------|
| 1 | Předmět      | Inv.číslo | Zakoupeno | Cena |
| 2 | židle dřev.  | 35765     | 1982      | 320  |
| 3 | stůl psací   | 35976     | 1977      | 1750 |
| 4 | židle kov.   | 35761     | 1980      | 260  |
| 5 | lampa stolní | 32176     | 1982      | 365  |
| 6 | židle dřev.  | 26452     | 1979      | 198  |

Databázi založíme tak, že klikneme levým tlačítkem myši na levý horní roh vybrané oblasti (návěští prvního sloupce) a se stisknutým tlačítkem táhneme směrem doprava a dolů. Tlačítko uvolníme, až změněná barva pokryje celou požadovanou oblast. V příkladu 1. klikneme na buňku A1 a vybereme oblast po buňku D6. Nyní zvolíme v menu položku Define Database a objeví se requester, do kterého zadáme jméno, pod kterým chceme databázi uložit. V našem případě třeba DATA.

### 2. Založení oblasti kritérií

Oblast kritérií může být založena na kterémkoliv místě worksheetu. Skládá se z jednoho nebo více sloupců.

Do těchto sloupců budeme zadávat kritéria, podle kterých bude databáze vyhodnocována. Podmínkou je, že první řádek sloupce obsahuje textový řetězec (návěští) toho sloupce, podle kterého bude databáze vyhodnocována. Pokud tedy budeme používat k hledání v databázi všechny sloupce, bude oblast kritérií obsahovat první řádek celé databázové oblasti. Proto je výhodné umístit kritéria pod vlastní databázi okopírováním celého prvního řádku databáze. Je to přehlednější a zůstávají nám zachovány šířky sloupců.

Příklad 2.

|   | A       | B         | C         | D    |
|---|---------|-----------|-----------|------|
| 8 | Předmět | Inv.číslo | Zakoupeno | Cena |
| 9 |         |           |           |      |

Nyní si budeme oblast kritérií definovat. Uděláme to stejně jako při definování oblasti databáze. Klikneme na horní levý roh kritérií a táhneme doprava přes celý řádek návěští a potom sjedeme o jeden řádek dolů. Do tohoto prázdného řádku budeme později zadávat podmínky pro jednotlivé sloupce, podle kterých budeme v databázi vyhledávat. V příkladu 2. tedy zvolíme oblast A8-D9. Nyní vybranou oblast zapíšeme. Zvolíme z menu funkci Define Criteria a zadáme jméno vybrané oblasti. V našem případě třeba KRIT1. Je nutno si uvědomit, že nemusíme pro kritéria zadávat všechna návěští databáze. Pokud bude potřeba hledat jen podle sloupce s cenou, stačí jako oblast kritérií definovat A8-A9, kdy v A8 bude text "Cena" a do buňky A9 budeme zadávat podmínku.

### 3. Podmínky kritérií a vyhledávací funkce Find

Nyní máme definovanou databázi i oblast kritérií. Buňky pod návěštími oblasti kritérií jsou zatím prázdné. Z menu si vybereme položku Select Database a klikneme na jméno uložené oblasti, v našem příkladu tedy DATA. Dále si z menu vybereme položku Select Criteria a vybereme název kritérií KRIT1. Klikneme v menu na vyhledávací funkci Find a na displeji se nám odlišnou barvou zobrazí celý první řádek databáze. Horní řádek worksheetu nás upozorňuje, že jsme v režimu Find. Protože zatím nemáme zadány žádné podmínky, můžeme se kurzorovými klávesami pohybovat dolů a nahoru po všech řádcích databáze. Režim Find opustíme stiskem klávesy Esc.

Nejdříve si popíšeme, jak se zadávají kritéria pro sloupce, obsahující textové řetězce, v naší databázi jde tedy o sloupec A. Do buňky A8 napíšeme "židle dřev.". Spustíme Find a změněnou barvou je vybrán z databáze první řádek, který obsahuje ve sloupci s návěští "Předmět" položku "židle dřev.". Kurzorovými klávesami se můžeme v databázi přesunovat po všech řádcích, které odpovídají zadané podmínce. V našem příkladu podmínce odpovídá řádek 2 a 6. Jestliže chcete vyhledávat jen podle určité části řetězce, můžete použít tzv. jockerový znak \*.

Jestliže tedy zadáte do kritérií ž\*, budou vyhledány řádky 2, 4 a 6, tedy všechny ty, které ve sloupci A začínají

písmenem ž nebo Ž. Jockerový znak může být použit i na začátku nebo uprostřed řetězce a nahrazuje všechna chybějící písmena!

Pro číselná data zapisujeme kritéria pro vyhledávání podle známých logických podmínek: <, >, <=, >=, =, za kterými následuje hodnota, podle které budeme databázi prohledávat. Ale pozor! V případě zadávání logických podmínek v kritériích musí být prvním znakem vždy znaménko =! Teprve za ním následuje vlastní podmínka. Znamená to tedy, že v případě požadavku =, bude zadáno =, jestliže potřebujeme > musíme zadat >=. Vyzkoušíme si to na našem příkladu. Smažeme obsah buňky A9 a do buňky C9 vložíme podmínku >=1980. Zvolíme Find a kurzorovými klávesami se můžeme přesvědčit, že budou vybrány řádky 2, 4 a 5. Všechny odpovídají podmínce, že hodnota ve sloupci C je rovna nebo větší než 1980. Nyní zadáme podmínku pro dva sloupce. Do buňky A9 zapíšeme ž\* a spustíme Find. Nyní je v databázi vybrán jen řádek 2 a 4. Podmínky v řádku se vyhodnocují jako logický součin AND (jestliže platí první podmínka a zároveň druhá podmínka). V našem příkladu jsou tedy vybrány jen ty řádky, které ve sloupci A začínají písmenem ž a zároveň je ve sloupci C hodnota větší nebo rovna 1980. Tak můžeme zadávat kritéria i pro další sloupce a kombinovat textové i číselné podmínky. Tyto jsou neustále spojovány a vyhodnocovány. Jak jste si jistě všimli, místa v řádku kritérií, kam nebyly zadány žádné podmínky nejsou vyhodnocovány. V našem příkladu by mohly být sloupce B a D zrušeny a oblast kritérií by se skládala pouze ze dvou řádků a dvou sloupců. Necháme ji však zatím tak a podíváme se na další možnosti zadávání podmínek. Klikneme na buňku A8 a táhnutím myši zvolíme oblast A8-D10. Zvolíme funkci Define Criteria a zde zadáme jméno nové oblasti kritérií KRIT2. Do této oblasti zapíšeme tyto podmínky:

**Příklad 3.**

|    | A       | B         | C         | D      |
|----|---------|-----------|-----------|--------|
| 8  | Předmět | Inv.číslo | Zakoupeno | Cena   |
| 9  | ž*      |           | >= 1980   | >= 300 |
| 10 | l*      |           | >= 1980   | <= 450 |

Zvolte Find a vidíte, že byly vybrány řádky 2 a 5. Jednotlivé podmínky pod sebou jsou spojovány jako logický součet OR (platí první podmínka nebo druhá podmínka). Tyto podmínky se současně kombinují s podmínkami AND jednotlivých řádků. V našem příkladu je zadáno, že mají být vybrány všechny židle a lampy zakoupené po roce 1980, jejichž cena je větší než 300 a menší než 450. Těmto podmínkám odpovídají právě řádky 2 a 5. Řádků můžeme v oblasti kritérií použít samozřejmě tolik, kolik potřebujeme podmínek. Kritéria typu OR a AND můžeme mezi sebou libovolně kombinovat, musíme však dodržet jednu podmínku. Všimněte si, že ve sloupci C je podmínka z prvního řádku zopakovaná i ve druhém. Pokud by totiž zůstal druhý řádek prázdný, databáze by jej vyhodnotila jako "první podmínka nebo všechno" a podmínka z prvního řádku by přestala platit. Proto musíme u kritérií s více řádky

doplnit všechna prázdná místa ve sloupci poslední podmínkou, kterou jsme pro daný sloupec zadali.

#### 4. Extract (vytvoření nové tabulky podle podmínek)

Pokud chceme s podmínkami vybraných řádků databáze vytvořit novou tabulku, najedeme kurzorem na kterékoliv volné místo worksheetu a zvolíme z menu funkci Extract/All. Všechny řádky databáze, které odpovídají podmínkám z oblasti kritérií, budou přepřesovány do tohoto místa směrem doprava a dolů. Dávejte si pozor, aby nově vytvořená tabulka nepřepsala již dříve zapsaná data! Jestliže se ve vaší databázi vyskytuje několik stejných řádků, potom zvolte u funkce Extract položku Unique. Stejně jako při použití All bude vytvořena nová tabulka podle podmínek, ale všechny řádky, které se opakují, budou odfiltrovány a v tabulce bude obsažen pouze jeden.

#### 5. Delete (smazání řádku podle zadaných podmínek)

Tato další funkce, vybraná z menu Data, nám umožňuje vymazat z oblasti databáze řádky podle zvolených podmínek. Má dvě podmenu Single Record a Matching Records. Single Record nám zruší první řádek, který odpovídá zadaným podmínkám. Před smazáním jste requestem dotázáni, zda řádek má být opravdu zrušen. Po volbě Delete/Matching Records budete postupně dotazováni na jednotlivé řádky. Pokud zvolíte "Yes", bude řádek vymazán, jestliže kliknete "No", bude zachován a jste dotazováni na zrušení dalšího vybraného řádku. Zrušený řádek nezůstává prázdný, ale celé pole databáze se vždy posune o jeden řádek výše. Pozor však na jednu zvláštnost. Pokud zvolíte Delete bezprostředně po funkci Extract, nebudou mazány řádky z oblasti databáze, ale z nové tabulky, vytvořené funkcí Extract. Toho lze v některých případech také využít. Pro mazání řádků v oblasti databáze po funkci Extract musíte nejdříve provést Find.

#### 6. Sort (seřazení)

Maxiplan Plus umožňuje seřadit jednotlivé sloupce databáze vzestupně nebo sestupně podle hodnot kódů ASCII. Pro tuto funkci si musíme zřídit novou oblast kritérií.

**Příklad 4.**

|    | A       | B         | C         | D    |
|----|---------|-----------|-----------|------|
| 12 | Předmět | Inv.číslo | Zakoupeno | Cena |
| 13 | 1       | 2         | 3         | -1   |

Jak vidíte, oblast pro sortování vypadá stejně jako v předchozích příkladech. Opět musí být v prvním řádku návštěví dané databáze. Kritéria pro seřazení však budou mít vždy jen jeden zadávací řádek. Nyní si vybereme pole A12-A13 a po zvolení funkce Define Criteria zadáme jméno SORT. V menu vybereme funkci Sort a všechny řádky databáze budou seřazeny abecedně podle prvního sloupce.

*Pokračování příště.*





# ASSEMBLER

*Pavel Přech*

Část 5.

Tak vás opět vítám u dalšího "sezení" nad naším milovaným MC68000.

Na rozdíl od minulých dílů se dnes nejprve vrhneme na dokončení základních instrukcí, které je nutné bezpodmínečně znát NAZPAMĚŤ

## LOGICKÉ OPERACE

Na logických operacích je postavena celá současná výpočetní technika, u níž je zaveden tzv. binární standard. Tato norma předpokládá existenci dvou stavů: 1 a 0. Stav "1" (označovaný též jako "High") se chápe jako kladný smysl (např. světlo svítí, pivo se pije atd ...) a stav "0 - Low" jako záporný (světlo nesvítí, restaurace zavřená atd ...)

Jakékoliv další stavy se nepřipouštějí a pokud vůbec nastanou, jedná se o chyby, které je třeba tvrdě zlikvidovat. Ona poslední věta pro nás jako programátory neplatí, neboť hardware počítačů je vyřešen tak, aby tyto hazardní stavy nemohly nastat.

Další informací, o které by programátor měl vědět, je existence několika početních soustav. Z našeho hlediska je důležitá hexadecimální, kterou jsem použil v minulých dílech (se znakem "\$") a již ona zmíněná binární. Je dobré si uvědomit, že výpočty lze provádět v JAKÉKOLIV soustavě, přičemž některé soustavy jsou vhodnější než ty druhé.

Na MC68000 jste, bohužel, omezeni na binární a jakž takž se dá počítat i v desítkové, ale binární suverénně vede.

Základem binární soustavy je Booleova algebra, což je předpis pro použití dvojkové soustavy.

### Booleova algebra

|      |                                                                                          |   |              |   |
|------|------------------------------------------------------------------------------------------|---|--------------|---|
| NOT  | negace - záměna 0 za 1 a opačně<br>"NOT(pravda) = lež"                                   | A | Y (výsledek) |   |
|      |                                                                                          | 0 | 1            |   |
|      |                                                                                          | 1 | 0            |   |
| AND  | Tabulka funkce AND<br>jev A nastává SOUČASNĚ s B<br>matematicky: $Y=A * B$               | A | B            | Y |
|      |                                                                                          | 0 | 0            | 0 |
|      |                                                                                          | 0 | 1            | 0 |
|      |                                                                                          | 1 | 0            | 0 |
|      |                                                                                          | 1 | 1            | 1 |
| OR   | Tabulka funkce OR<br>nastane jev A NEBO B<br>matematicky: $Y = A + B$                    | A | B            | Y |
|      |                                                                                          | 0 | 0            | 0 |
|      |                                                                                          | 0 | 1            | 1 |
|      |                                                                                          | 1 | 0            | 1 |
|      |                                                                                          | 1 | 1            | 1 |
| XOR, | taktéž EOR<br>"dva stejné bity dají nulu"<br>$Y = A * \text{not}(B) + \text{not}(A) * B$ | A | B            | Y |
|      |                                                                                          | 0 | 0            | 0 |
|      |                                                                                          | 0 | 1            | 1 |
|      |                                                                                          | 1 | 0            | 1 |
|      |                                                                                          | 1 | 1            | 0 |

Instrukční soubor MC68000 obsahuje samozřejmě i logické instrukce.

NOT.x      čeho      x = .L, .W, .B

Provede negaci. Bity, které byly v "1", se vynulují a naopak.

Povoleny jsou všechny adresové módy a datové registry.

AND.x      co, k čemu  
ANDI.x    #číslo, k čemu

Provede AND dle výše uvedené tabulky. Povoleny jsou všechny adres. módy a datové registry. ANDI se používá pro ANDování konstanty s cílovým objektem.

OR.x      co, k čemu  
ORI.x    #číslo, k čemu

Provede OR. Jinak viz AND.

EOR.x      co, k čemu  
EORI.x    #číslo, k čemu

Provede funkci XOR. Jinak viz AND.

## ROTACE

Další operací v binární soustavě je rotace. Princip vyplývá již z názvu a jde tedy o posun binárního čísla vlevo či vpravo.

Tady doporučuji vzít si tužku a čtverečkový papír a několikrát si nakreslit libovolné bin. číslo a několikrát rotovat. Pokud si odrotované číslo převedete zpět do desítkové soustavy, zjistíte zajímavou skutečnost:

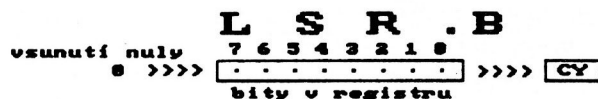
Obecně existuje dvojí rotace a to a) klasická  
b) kruhová

ad a) Klasická je prosté posunování bitu za bitem v určeném směru.

Bity, které se dostanou na krajní pozice, se při následujícím posunu ztratí. Procesory naštěstí pamatují na jakási "zadní vrátka", a tak se v tuto chvíli ztracený bit uschovává do dalšího posunu.

Při rotaci tedy procesor jakoby "rozšiřuje" délku registru o jeden bit.

Bit, který je vyrotován z registru, je uschován v příznaku Carry. Pak lze elegantně skákat pomocí BCS, resp. BCC, podle toho, byl-li odrotovaný bit jednička či nula.





LSR.x # o\_kolik\_bitů, co ; RIGHT ( vpravo )  
 LSL.x ; opačně - LEFT  
 LSR.x nebo LSL.x paměťové\_místo

angl. Logical Shift Right / Left

LSR / LSL je posun vpravo / vlevo. Číslo "o\_kolik\_bitů" udává, kolikrát se vykoná posun o jeden bit. Povoluje se číslo 0-7.

**UPOZORNĚNÍ:** Provádíte-li LSR, LSL s obsahem paměti, lze rotovat POUZE o 1 bit ! Chcete-li o více, je nutné použít smyčku (nebrat doslova!), nebo napsat několik LS. za sebou.

!!! LSR, LSL funguje tak, že do "prázdného místa", kde vzniká po posunu jakési "vakuum", AUTOMATICKY VKLÁDÁ NULU. !!!

U "Logical Shift" dochází k zajímavé matematické operaci, a to DĚLENÍ a NÁSOBENÍ mocninami čísla 2. (To je ten příklad na čtverečkováném papíře).

Zjistíte, že při shiftu DOLEVA, je po prvním posunu původní číslo vynásobeno dvěma, po dalším čtyřmi, osmi, šestnácti atd ...

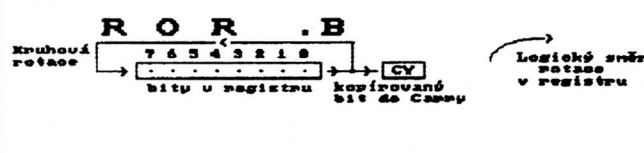
Podobně u shiftu VPRAVO je číslo ( samozřejmě celočíselné ) vyděleno dvěma, pak čtyřmi, osmi atd ...

**Pozn:** Mnozí počítačové teoretici dodnes polemizují nad již zavedeným faktem, a to, na kterou stranu roste mocnina základu soustavy. ( v desítkové doleva, např. 1000 100 - jednička řádu se připsala vlevo )

Tito individualisté jsou však díky svým utopistickým myšlenkám "mocnina roste vpravo" odsouzeni do nezáviděníhodné role statistiků ve velkém sci-fi westernu. Takže - vyšší řád vlevo, nejnižší vpravo.

ad b) Princip kruhové rotace vyplývá z názvu - rotuje se dokola, tedy data se "neztrácejí" a obíhají stále periodicky, přičemž mohou nastat dvě varianty :

- 1) Klasická kruhová - bity obíhají dokola určeným směrem.  
 Princip : Provede se Logical Shift určeným směrem, vyrotovaný bit SE ZKOPÍRUJE do Carry a ZÁROVEŇ do "PROTĚJŠÍHO ROHU". (uzavření kruhu)

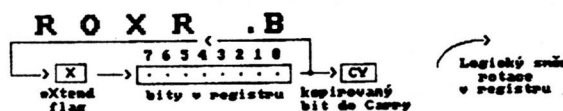


ROL.x resp ROR.x

Parametry stejné jako LSR, LSL.

- 2) Kruhová s rozšířením - k vlastní délce registru se přidává flag eXtend

Princip : Provede se Logical Shift, avšak i s eXtend flag eXtend se zkopíruje do Carry



ROXL.x resp ROXR.x

Parametry jako LSR, LSL.

**POZOR: LOGICKÉ INSTRUKCE OVLIVŇUJÍ PŘÍZNAKY !!!**

Zero se nastaví, je-li výsledek nula.  
 Ostatní podle charakteru instrukce a výsledku.  
 U AND,OR,XOR,NOT je Carry vždy nula.

**Tipy a triky :** Podíváte-li se pozorně na obrázky ROXL a ROXR, zjistíte, že se vždy po provedení těchto příkazů rovnají eXtend a Carry.  
 Toho pak lze elegantně využít.

## TESTOVÁNÍ

Čtete-li dobře tento časopis, tak jste si jistě nenechali ujít předešlý díl našeho MACHINE-SOFTWARE a vzpomenete si na popis skoků.  
 Nicméně doporučuji vzít si do ruky minulé číslo.

Jak vidíte, skoky se realizují na základě výsledku předešlé operace.

Tuto operaci může realizovat jakákoliv aritmetická nebo logická instrukce, čehož se hojně využívá především u matematických rutin. Co však v případě, kdy není třeba počítat, ale zjišťovat např. správnost provedení nějaké funkce?

**Pozn:** Funkce vracejí tzv. RESULT či



**ERROR\_CODE**, což je číslo, které udává druh chyby. Číslo se umísťuje takřka vždy do D0.

Na Amize se používá číslo 0 (LONG tvar) pro CHYBU (POZOR!! NEPLATÍ VŽDY!!) a to u funkcí vracejících odpověď typu CHYBA / SPRÁVNĚ a totéž jako "ÚSPĚŠNÉ PROVEDENÍ" - u funkcí vracejících číslo chyby při provádění. (Uvědomte si prosím, že signál "SPRÁVNĚ" je také druh chyby.)

Odpověď je zdánlivě jednoduchá - pomocí logických instrukcí (AND, XOR...) můžeme sestavit takový test, kdy se po provedení poslední instrukce tohoto testu nastaví příznakové bity na námi požadovanou úroveň.

Protože však platí heslo - Čas jsou peníze - tak si výborný programátor dává velký pozor na to, aby jeho rutina běžela co nejrychleji.

V případě, že bude nucen používat často testy složené z mnoha instrukcí, jeho program bude pomalý a takový programátor pak pravděpodobně pojde hladem.

Nevěšte však hlavu - kvalitní procesory jako MC68000 znají instrukce, které testují specifické stavy, u kterých bylo statisticky zjištěno, že nastávají nejčastěji.

## 1. TEST NULY

Používá se při zjištění, zda funkce či podprogram (např. z Amiga ROM) proběhl správně, nebo při testu proměnných BOOLEAN - true / false, kde true je obecně jakékoliv nenulové číslo.

**TST.x**      **co\_se\_má\_testovat**      **x je .L, .W, .B**

Instrukce povoluje všechny adresové módy.

Tato instrukce zjišťuje, zda obsah či hodnota testovaného objektu je NULOVÁ.

Je-li nulová, nastavuje se příznak Z do jedničky (je NULA - EQUAL), a tak chcete-li při nule skočit, napište to v assembleru takto:

**TST.L**      **D0**      **je v D0 nula ?**  
**BEQ.s**      **je\_nula**      **jestli ano, skoč**

Při nenule obraťte podmínku (BNE)

Pozn: Nebude-li se Vám TST hodit - pak si pomozte třeba takto:

**AND.L D0,D0**  
**BEQ.s**      **je\_nula**

Tento způsob je odvozen z vlastní logické funkce AND - zANDujete-li dvě čísla, je výsledek nula, je-li alespoň jedno číslo nula. Zde ANDujete číslo (v D0) sebou samým.

Podobně to platí u funkce OR. Fantazii se meze nekladou.

**OR.B**      **D0,D0** ; číslo bude např. BYTE - proto **.B**  
**BEQ.s**      **je\_nula**

Zajímavou, nicméně téměř nepoužívanou instrukcí je TAS. ("TEST and SET")

Instrukce funguje podobně jako TST, avšak vždy po otestování nastaví nejvyšší bit objektu do 1. Samotnému mi není příliš jasné, proč taková instrukce byla vůbec zařazena do instrukční sady. Že by náhoda při konstrukci hardware? Ví to někdo z Vás?

Jako ne moc uspokojivé vysvětlení mě napadá: otestuj a změň znaménko.

(u integer čísel slouží nejvyšší bit jako znaménko - viz Relativní adres.)

## 2. TESTOVÁNÍ A OPERACE S BITY

Testování určitého bitu lze v praxi provést několika způsoby.

Nejprve si ukážeme poněkud těžkopádný, avšak dle mého názoru výborný způsob, který oceníte až později.

Testování bitu je možné provést SEPARACÍ. Nepotřebné bity jednoduše pomocí logických instrukcí "ořezáváme" tak dlouho, dokud nedostaneme jednoznačný výsledek typu - je nula nebo ne. Následuje podmíněný odskok instrukcí BEQ, resp. BNE. (Není nutné provádět TST, neboť logické operace ovlivňují flagy - viz výše.)

Příklad: Chceme otestovat bit # 3 v registru D0

Tato varianta používá AND. Logicky se vyžívá Bool. funkce AND, kde "cokoliv \* 0 je nula"

;  
; **v D0.b je již nějaké číslo**      **x ... znamená 0 nebo 1**  
; **and.b #%00001000,D0**      **xxxx1xxx**  
; **and.b**      **AND 00001000 (AND MASKA)**  
; **beq.s je\_nulovy**      **výsledek 00001000**  
; **jak je vidět, bude-li bit 3 v nule, je výsledkem nula**

Tady zase princip spočívá v doplnění "okolí" bitu jedničkami.

Jak jsem výše uvedl, je dobré testovat pouze nulu (lze to i krkolomně) a tak proto NOT.

;  
; **v D0.b je již nějaké číslo**      **xxxx0xxx**  
; **ori.b #%11110111,D0**      **11110111 (OR MASKA)**  
; **ori.b**      **mezivýsledek 11110111**  
; **not.b D0**      **NOT dá 00001000**  
; **bne.s je\_nulovy**

Taktéž lze i pomocí LSR. (jak elegantní...)

**Carry**  
;  
; **v D0.b je již nějaké číslo**      **xxxx0xxx**      **x**  
; **lsr.b #4,D0**      **>>>> 0000xxxx**      **0**  
; **bcc.s je\_nula**

Všechno to, co bylo uvedeno, použijte, prosím, jen jako podklady pro vaši práci, abyste snáze pochopili, jak myslet při tvorbě svých CODE. Ale to nejlepší, co pro vás mohli vývojáři udělat, bylo vytvoření instrukcí přímo pro práci s bity. Při použití těchto instrukcí se automaticky

nuluje Carry flag, lze skákat podle Zero - BEQ, BNE.

**BTST** Bit TeST - otestuj bit a výsledek uloží do flagu Zero  
**BTST.x** # číslo\_bitu, co\_testovat

BTST povoluje všechny druhy adresování, avšak v praxi se nejvíce používá testování v registru. Číslo\_bitu může být 0 - 31, není-li omezeno koncovkou .x-.L, .W, .B. Testovat např. 23 bit WORD slova je nesmysl!

Důležité upozornění: Testujete-li bit na SUDÉ adrese, mějte na paměti, že ZÁKLADNÍM objektem u MC68000 je WORD !

Tudíž :

**BTST** # 3, sudá\_adresa

otestuje ve skutečnosti bit 3 na následující adrese (+1), která je lichá,

avšak **BTST.B** # 3, sudá\_adresa již funguje ideálně.  
 V případě **BTST** # 3, lichá\_adresa to ale pracuje spolehlivě, neboť na lichou adresu se dá umístit pouze BYTE, který je tvořen bity 0-7.

Upozornění pro vás nebude platit v případě, že budete důsledně vypisovat přípony .L,.W,.B. Pokud je nevypíšete, ASSEMBLER IMPLICITNĚ PŘEDPOKLÁDÁ WORD. Dejte si zvlášť velký pozor na adresování s registrem, např. **BTST** # 0, (A0) kde není vždy jisté, je-li v A0 sudá či lichá adresa, tudíž znovu : VYPISUJTE KONCOVKY !

Data (WORD) jsou v paměti uloženy :

|             |            |      |                                   |
|-------------|------------|------|-----------------------------------|
| sudá_adresa | Vyšší_byte | bity | 16-8                              |
| adresa + 1  | Nižší_byte |      | 7-0 <- zde je to jasně vidět (#3) |

Podobně je uložen i LONG, který se rozpadá na dva WORDY

|             |               |      |              |
|-------------|---------------|------|--------------|
| sudá_adresa | Nejvyšší_byte | bity | 31-24        |
| adresa + 1  | Nižší_byte    |      | 23-16        |
|             |               |      | - Vyšší WORD |

|            |               |  |              |
|------------|---------------|--|--------------|
| adresa + 2 | Vyšší_byte    |  | 15-8         |
| adresa + 3 | Nejnižší_byte |  | 7-0          |
|            |               |  | - Nižší WORD |

Dalšími instrukcemi jsou instrukce pro přímou práci s bity.

O těchto instrukcích platí totéž, co o BTST.

**BSET.x** # číslo\_bitu, co " BIT SET "

Nastaví požadovaný bit do jedničky. Zero flag se vynuluje.

**BCLR.x** # číslo\_bitu, co " BIT CLEAR "

Vynuluje požadovaný bit. Zero flag se nastaví do jedničky - ZERO, EQUAL.

**BCHG.x** # číslo\_bitu, cíl " BIT TEST AND CHANGE "

Nejprve otestuje požadovaný bit a změní jeho log. hodnotu (NOT).

Zero se nastaví podle původní hodnoty bitu.

## Dárky pro Vás

\*\*\*\*\*  
 DRUHA ODMOCNINA Z D0.L, postupnou aproximací P.S. Nezapomeňte, že se počítá celočíselně.

\*\*\*\*\*

|            |                |               |                     |
|------------|----------------|---------------|---------------------|
| <b>SQR</b> | <b>move.l</b>  | <b>d0,d2</b>  | <b>MAX - aprox.</b> |
|            | <b>moveq.l</b> | <b># 0,d1</b> | <b>MIN - aprox.</b> |

**SQR\_t**

|               |              |                          |
|---------------|--------------|--------------------------|
| <b>move.l</b> | <b>d2,d3</b> | <b>D0 = D2 = A</b>       |
| <b>sub.l</b>  | <b>d1,d3</b> | <b>D1 D1-D3</b>          |
| <b>lsl.l</b>  | <b>#1,d3</b> | <b>B = (MAX - MIN)/2</b> |
| <b>beq.s</b>  | <b>SQR_e</b> | <b>B = 0 - spočteno</b>  |

|               |              |                    |
|---------------|--------------|--------------------|
| <b>move.l</b> | <b>d1,d4</b> | <b>D4 D1</b>       |
| <b>add.l</b>  | <b>d3,d4</b> | <b>D4 D3 + D4,</b> |
|               |              | <b>C = MIN + B</b> |

|               |                  |                                                         |
|---------------|------------------|---------------------------------------------------------|
| <b>move.l</b> | <b>d0, -(a7)</b> | <b>Schovej D0 na zásobník</b>                           |
| <b>move.l</b> | <b>d4,d0</b>     | <b>D0 D4</b>                                            |
| <b>mulu</b>   | <b>d0,d0</b>     | <b>C ^ 2... mulu je instrukce násobení (ale pomalá)</b> |
| <b>cmp.l</b>  | <b>(a7),d0</b>   | <b>C ^ 2 A?</b>                                         |
| <b>bcc.s</b>  | <b>CvA</b>       | <b>skok při CA</b>                                      |

|              |              |                      |
|--------------|--------------|----------------------|
| <b>add.l</b> | <b>d3,d1</b> | <b>D1 D1 + D3,</b>   |
|              |              | <b>MIN = MIN + B</b> |

|            |               |                   |                              |
|------------|---------------|-------------------|------------------------------|
| <b>rep</b> | <b>move.l</b> | <b>(a7) + ,d0</b> | <b>vyber D0 ze zásobníku</b> |
|            | <b>bra.s</b>  | <b>SQR_t</b>      |                              |

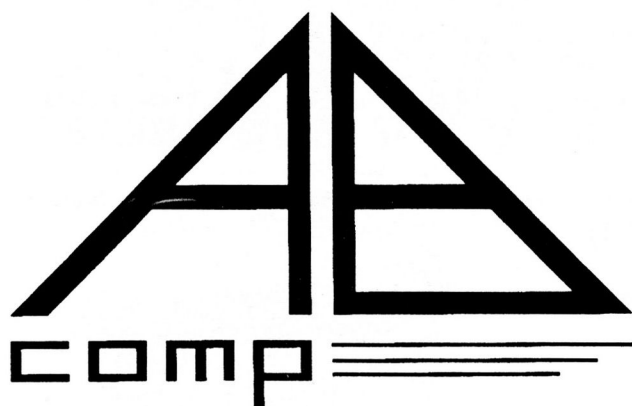
|            |              |              |                      |
|------------|--------------|--------------|----------------------|
| <b>CvA</b> | <b>sub.l</b> | <b>d3,d2</b> | <b>MAX = MAX - B</b> |
|            | <b>bra.s</b> | <b>rep</b>   |                      |

|              |            |                                   |
|--------------|------------|-----------------------------------|
| <b>SQR_e</b> | <b>rts</b> | <b>Spočtená odmocnina je v D1</b> |
|--------------|------------|-----------------------------------|

□

**MRP-NS-12**





## a COMMODORE

Naše redakce již půl roku velice dobře spolupracuje s firmou A - B - Comp, ale bohužel zatím jsme našim čtenářům neměli možnost tuto firmu podrobněji představit. Do tohoto čísla se nám podařilo díky právě této spolupráci uskutečnit rozhovor, který přiblíží našim čtenářům, čím se firma A - B - Comp zabývá a co nového přinese do oblasti počítačů "AMIGA" v Československu. Firma A - B - Comp byla založena v březnu 1990 jako soukromý podnik pro poskytování komplexních služeb v oblasti výpočetní techniky. V současnosti má divizní uspořádání. Konkrétně je složena ze tří následujících divizí:

|                                 |                                                            |
|---------------------------------|------------------------------------------------------------|
| <b>Divize domácích počítačů</b> | <b>(C64, AMIGA 500 až po CDTV)</b>                         |
| <b>Divize PC</b>                | <b>(SL 286-16 až po nejvýkonnější T 486-50 C)</b>          |
| <b>Divize grafických stanic</b> | <b>(AMIGA 2000, AMIGA 3000 + příslušné vybavení, CDTV)</b> |

Hlavními představiteli a majiteli uvedené firmy jsou Radek Bláha a ing. Jiří Vojtěch. O jejich kvalitách jednoznačně hovoří roční obrát, který dosáhl v roce 1991 úctyhodné výše 30 mil.Kčs.

Na tomto obrátu měla podstatný podíl právě divize domácích počítačů, která nabízí prostřednictvím třiceti dealerů po celém území Československa kompletní řadu výrobků firmy Commodore od domácího počítače C 64 až po Amigu s operačním systémem UNIX.

Vzhledem k tomu, že náš časopis je zaměřen na počítače řady AMIGA, zaměřili jsme i náš rozhovor tímto směrem.

**Co nového v oblasti počítačů řady AMIGA firma A - B - Comp připravuje?**

K závažné změně v postavení fy A - B - Comp došlo na letošním CeBITu, kde jsme podepsali smlouvu s firmou COMMODORE a stali se tak prvním autorizovaným distributorem v Československu. Co z této smlouvy vyplývá: V první řadě přímé dodávky hardwaru od fy COMMODORE, kvalita, serióznost a solidnější cena pro Vás jako zákazníky. Prohlédneme-li si nabídku firmy COMMODORE, v oblasti domácích počítačů a grafických stanic, tak prakticky všechno si mohou naši prodejci objednat.

Ať už se jedná o

**-klasický sortiment:** AMIGA 500, AMIGA 500 PLUS, CDTV, AMIGA 2000

**-speciální sortiment:** AMIGA 3000, AMIGA 3000 UX, AMIGA 3000 T

**-novinky:** AMIGA 600, AMIGA 600HD, AMIGA CDTV, CDROM jednotka A 570.  
**-příslušenství** pro AMIGY a CDTV.

**Jak jste se vypořádali se servisem pro počítače AMIGA?**

Firma AB Comp myslí i na tuto oblast a zajišťuje komplexní servisní služby pro své zákazníky.

**Co to znamená?**

Na stroje nakoupené prostřednictvím naší firmy pochopitelně poskytujeme záruku. Vyskytne-li se tedy nějaká závada v záruční době, řešíme ji většinou výměnou za nový přístroj.

Naši snahou je, aby zákazník mohl pracovat dále s nejmenší časovou prodlevou. Pozáruční opravy zajišťujeme samozřejmě také.

Máme vlastní tým servisních techniků, kteří zvládají dosud veškeré práce během několika dnů.

**Naše čtenáře bude ale zajímat i to, mohou-li se na Vás obrátit i v případě, že svůj počítač nakoupili u jiného prodejce, popřípadě si jej individuálně přivezli ze zahraničí?**

Záruku na výrobky ostatních prodejců nemůžeme zařizovat. V tom případě musí každý za tou organizací, která mu přístroj prodala. To pochopitelně někdy může znamenat cestu stovek kilometrů s nejistým výsledkem, a navíc další vyřizování celních formalit. To vše hovoří pro nákup v Československu. Pozáruční opravy strojů nezakoupených u našich prodejců se ujme a vyhovíme, pokud to bude v našich, zejména časových možnostech.

**Uzavřela, nebo připravuje Vaše firma některé další smlouvy?**

Samozřejmě, že fy A - B - Comp chce být v okruhu, který se úzce dotýká AMIGY nejlepší, musí tedy spolupracovat i s ostatními dodavateli. V oblasti grafických stanic v nejbližší době uzavře smlouvu s fy SILICON GRAPHICS a s fy THOMSON DIGITAL IMAGE. Dále pak v hardwaru, který vhodně doplňuje firmu COMMODORE pro AMIGY, je uzavřená distributorská smlouva s fy GVP, která nabízí harddisky a turbo karty nejvyšší kvality.

**Stále však mluvíme o hardware a jak vypadá situace v oblasti programového vybavení software?**

Nedostatek legálního softwaru si uvědomujeme, ale Československý trh je specifický nelegálním kopírováním. Přes tuto skutečnost se nám podařilo navázat obchodní kontakty s několika softwarovými firmami, a přesvědčit je o nutnosti poskytnout závaděcí ceny na našem trhu. V nejbližší době bude vyhlášena několikaměsíční amnestie při velice nízkých cenách (DeluxePaint IV v ceně cca 3500,-Kčs), po níž budou ceny upravené a uživatelé nelegálního programového vybavení mohou být postihováni podle platných zákonů.

**Co nového připravujete v oblasti grafických stanic?**

Nejvhodnějším počítačem pro videotechniku a animaci je Amiga 2000 a hlavně Amiga 3000, nejen pro své technické vlastnosti, ale i pro poměr výkon/cena. V současné době můžeme například nabídnout pro Amigu 2000 turbokartu 68030 na 50 MHz firmy GVP, a pro Amigu 3000 kartu VD 2001 s digitalizací v reálném čase s 16 milióny barev.

Se zástupci fy AB Comp neformálně diskutovali zástupci redakce "amiga".

Jako zajímavost pro naše čtenáře uvádíme, že jedním z prodejců této úspěšné firmy je redakce časopisu "amiga" se svým zásilkovým prodejem.

# **VELKÁ PREZENTACE FIRMY Commodore V PRAZE**

**Firma A - B - Comp pořádá  
první velkou prezentaci firmy Commodore v Československu**

**Na přehlídce bude představen kompletní sortiment výrobků firmy  
Commodore, které jsou již distribuovány prostřednictvím dealerů  
a naší prodejní sítě .**

**Předvedeny budou výrobky ze všech oblastí:**

**Domácí počítače  
PC počítače  
Grafické stanice  
Unixové stanice**

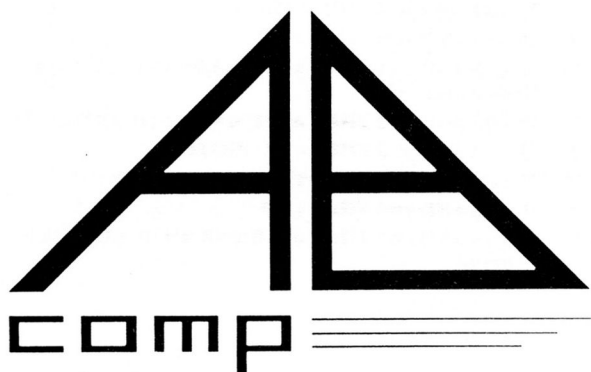
**Přehlídka se uskuteční ve čtvrtek 30. dubna 1992  
v Kaiserštejnském paláci na Malé Straně**

**Přehlídky se zúčastní viceprezident firmy Commodore  
a prezident evropské pobočky Commodore pan Helmut Jost,  
a další význační hosté.**

**Vstup na prezentaci bude pouze na základě pozvánky.  
Zájemci o účast na této zajímavé akci se mohou hlásit  
na tel. 72 51 41-7 l.16**

**Na shledanou s Vámi se těší firma  
A - B - Comp  
autorizovaný distributor Commodore v ČSFR**

**AUTORIZOVANÝ DISTRIBUTOR**



# POČTY

Program "Počty" je velmi jednoduchý, a slouží k procvičování základních matematických operací (sčítání, odčítání, násobení a dělení) z paměti.

Aby bylo procvičování zajímavější a přitažlivé, je v programu počítání chyb a času potřebného na výpočet příkladu. Pro kontrolu slouží rodičům výsledková tabulka.

```
' Program : Zkouseni na deleni, nasobeni, scitani a odcitani
'
' Pocitac : A500
' Jazyk : Amiga Basic
' Bytes : 9740
' Autor : MikySoft - Mikulecky 1991,
' (c) : Redakce časopisu Amiga 1992
```

```
95 MENU 1,3,1, " Nahodnych cisel do ..n "
100 MENU 1,5,1, " Vysledkova tabulka "
105 MENU 1,6,1, " Konec pocitani "
110 MENU 2,0,1, " Odcitani "
115 MENU 2,1,1, " Nahodnych malych cisel do 10 "
120 MENU 2,2,1, " Nahodnych velkych cisel do 100 "
125 MENU 2,3,1, " Nahodnych cisel do ..n "
130 MENU 2,5,1, " Vysledkova tabulka "
135 MENU 2,6,1, " Konec pocitani "
140 MENU 3,0,1, " Nasobeni "
145 MENU 3,1,1, " Mala nasobilka "
150 MENU 3,2,1, " Velka nasobilka "
155 MENU 3,3,1, " Nahodnych cisel do ..n "
160 MENU 3,5,1, " Vysledkova tabulka "
165 MENU 3,6,1, " Konec pocitani "
170 MENU 4,0,1, " Deleni "
175 MENU 4,1,1, " Nahodnych malych cisel do 10 "
180 MENU 4,2,1, " Nahodnych velkych cisel do 100 "
185 MENU 4,3,1, " Urceni delitele "
190 MENU 4,5,1, " Vysledkova tabulka "
195 MENU 4,6,1, " Konec pocitani "
200 ON MENU GOSUB 210
205 GOTO 200
210 MENU OFF
215 COLOR 1,0
220 x=MENU(0):y=MENU(1)
```

## Poety

Nasobeni nahodnymi cisly (1. cinitel do 24 , 2. cinitel do 47 ).

14 x 23 =

Pocet prikladu = 2

```
1 DIM d(5,5), e(5)
5 DEFINT lj
10 SCREEN 1,640,200,3,2
20 WINDOW 2," Poety ",(0,0)-(631,186),16,1
25 RANDOMIZE TIMER
30 PALETTE 1,.06,.8,.9 '--- svetle modra
35 PALETTE 2,.7,.7,.7 '--- seda
40 PALETTE 3,.8,.95,.15 '--- zluta
45 PALETTE 4,.1,.75,.15 '--- zelena
50 PALETTE 5,.1,.15,.3 '--- cervena
55 PALETTE 6,0,0,0 '--- cerna
60 PALETTE 7,.8,.55,.4 '--- pismo v menu
65 m=4
70 GOSUB 1110
75 MENU ON
80 MENU 1,0,1, " Scitani "
85 MENU 1,1,1, " Nahodnych malych cisel do 10 "
90 MENU 1,2,1, " Nahodnych malych cisel do 100 "
```

```
225 IF x=1 AND y=1 THEN a=10 :e=1 :GOTO 495 '-- scitani
230 IF x=1 AND y=2 THEN a=100 :e=1 :GOTO 495
235 IF x=1 AND y=3 THEN c=1 :e=1 :GOTO 335
240 IF x=1 AND y=5 THEN 1175
245 IF x=1 AND y=6 THEN 1155
250 IF x=2 AND y=1 THEN a=10 :e=2 :GOTO 545
'-- odcitani
255 IF x=2 AND y=2 THEN a=100 :e=2 :GOTO 545
260 IF x=2 AND y=3 THEN c=2 :e=2 :GOTO 335
265 IF x=2 AND y=5 THEN 1175
270 IF x=2 AND y=6 THEN 1155
275 IF x=3 AND y=1 THEN a=1 :e=3 :b=10 :GOTO 650
'-- nasobeni
280 IF x=3 AND y=2 THEN a=11 :e=3 :b=50 :GOTO 650
285 IF x=3 AND y=3 THEN e=3 :GOTO 380
290 IF x=3 AND y=5 THEN 1175
295 IF x=3 AND y=6 THEN 1155
300 IF x=4 AND y=1 THEN a=10 :e=4 :b=10 :GOTO 760
'-- deleni
```



```

305 IF x=4 AND y=2 THEN a=100 :e=4 :b=100 :GOTO 760
310 IF x=4 AND y=3 THEN e=4 :GOTO 450
315 IF x=4 AND y=4 THEN e=4 :GOTO 715
320 IF x=4 AND y=5 THEN 1175
325 IF x=4 AND y=6 THEN 1155
335 CLS
340 LOCATE 10,10 :INPUT "Jaka muze byt nejvyssi hodnota
    cisla:",a$
345 IF a$="" THEN 325
350 IF ASC(a$)<=57 AND ASC(a$)=>48 THEN 360
355 GOTO 335
360 a=INT(VAL(a$))
365 CLS
370 GOSUB 1110
375 ON e GOTO 495,545
380 CLS
385 LOCATE 10,10 :INPUT "Jaky muze byt nejvyssi
    cinitel:",a$
390 IF a$="" THEN 380
395 IF ASC(a$)<=57 AND ASC(a$)=>48 THEN 405
400 GOTO 380
405 a=INT(VAL(a$))
410 LOCATE 12,10 :INPUT "Jaky muze byt nejvyssi
    2. cinitel:",b$
415 IF b$="" THEN 405
420 IF ASC(b$)<=57 AND ASC(b$)=>48 THEN 430

```

```

510 e=1
515 i=INT(RND*a)
520 j=INT(RND*a)
525 GOSUB 825
530 IF k<>i+j THEN p=1 :GOTO 910
535 GOSUB 990
540 GOTO 515
545 LOCATE 2,1 :PRINT SPACE$(80)
550 LOCATE 2,3 :PRINT "Odecitani nahodnych cisel, nejvyssi
    cislo muze byt";a
555 f=45
560 e=2
565 i=INT(RND*a)
570 j=INT(RND*a)
575 IF i<j THEN g=i :i=j :j=g '--- prohozeni odecitance
    a odecitatele
580 GOSUB 825
585 IF k<>i-j THEN p=2 :GOTO 910
590 GOSUB 990
595 GOTO 565
600 LOCATE 2,1 :PRINT SPACE$(42)
605 LOCATE 2,3 :PRINT "Nasobeni nahodnymi cisly
    (1. cinitel do";a; ", 2. cinitel do";b; ")."
610 f=120 '--- volitelnymi nahodnymi cisly
615 e=3
620 i=INT(RND*a)

```

## Pocty

Deleni nahodnych cisel (delenec do 100 a delitel do 10 ).

**D O B R E** pro pokracovani stiskni ENTER

Uloha byla vyresena za 3.879492 sekundy.

Pocet prikladu = 8

Celkovy cas: 8 : 0 : 43

Pocet chyb = 0

Prumerny cas na priklad: 5.23939

```

425 GOTO 405
430 b=INT(VAL(b$))
435 CLS
440 GOSUB 1110
445 GOTO 600
450 CLS
455 LOCATE 12,10 :INPUT "Jaka muze byt nejvetsi hodnota
    delitele:",a$
460 IF a$="" THEN 70
465 IF ASC(a$)<=57 AND ASC(a$)=>48 THEN 475
470 GOTO 450
475 j=INT(VAL(a$))
480 CLS
485 GOSUB 1110
490 GOTO 715
495 LOCATE 2,1 :PRINT SPACE$(80)
500 LOCATE 2,3 :PRINT "Scitani nahodnych cisel, ktera nejsou
    vetsi nez";a
505 f=43

```

```

625 j=INT(RND*b)
630 GOSUB 825
635 IF k<>i*j THEN p=5 :GOTO 910
640 GOSUB 990
645 GOTO 620
650 LOCATE 2,1 :PRINT SPACE$(80)
655 IF a<10 THEN n$="Mala" ELSE n$="Velka"
660 LOCATE 2,3 :PRINT n$ " nasobilka"
665 f=120
670 e=3
675 FOR i=a TO b
680     FOR j=1 TO 10
685         GOSUB 825
690         IF k<>j*i THEN p=3 :GOTO 910
695         GOSUB 990
700     NEXT j
705 NEXT i
710 GOTO 675
715 LOCATE 2,1 :PRINT SPACE$(80)

```

```

720 LOCATE 2,3:PRINT "Deleni urcenym delitelem
      (delitel =";j;")."
725 f=58
730 e=4
735 i=INT(RND*10)*j
740 GOSUB 825
745 IF k<>i/j THEN p=4:GOTO 910
750 GOSUB 990
755 GOTO 735
760 LOCATE 2,1:PRINT SPACES$(80)
765 LOCATE 2,3:PRINT "Deleni nahodnych cisel
      (delenec do";b*b;"a delitel do";a;")."
770 f=58
775 e=4
780 i=INT(RND*a)
785 j=INT(RND*b)
790 IF i<j THEN g=i:i=j:j=g
795 IF j=0 THEN 780
800 i=i*j
805 GOSUB 825
810 IF k<>i/j THEN p=4:GOTO 910
815 GOSUB 990
820 GOTO 780
825 pr=pr+1
830 COLOR 6,7
835 LOCATE 20,10:PRINT "Pocet prikladu ="pr

```

```

930 COLOR 6,2
935 LOCATE 10,20:PRINT "S P A T N E  priklad spoctej
      znovu !!"
940 d(e,3)=d(e,3)+1
945 GOSUB 1135
950 ON TIMER(2) GOSUB 965
955 TIMER ON
960 GOTO 955
965 TIMER OFF
970 COLOR 6,7
975 LOCATE 22,10:PRINT "Pocet chyb ="ch
980 COLOR 6,2
985 ON p GOTO 525,580,685,805,630
990 w=TIMER
995 r=w+zz-v
1000 rz=rz+r
1005 min=INT(rz/60)
1010 hod=INT(min/60)
1015 sec=INT(rz-min*60)
1020 LOCATE 17,20:PRINT STRING$(LEN(STR$(rr))+32,32)
1025 IF r<5 THEN s$="y" ELSE s$=""
1030 LOCATE 17,20:PRINT "Uloha byla vyresena za
      "r"sekund";s$;".
1035 rr=r
1040 COLOR 6,7
1045 LOCATE 20,40:PRINT "Celkovy cas: ";hod;":"min;":"sec

```

## Pocty, vysledkove hodnoceni

### Vysledek tvoje snazeni \*\*\*\*\*

| PRIKLADY      | celkem    | z toho    |          | uspesnost | znanka   |
|---------------|-----------|-----------|----------|-----------|----------|
| na            |           | dobre     | spatne   | v %       |          |
| scitani       | 14        | 14        |          | 100       | 1        |
| odecitani     | 20        | 19        | 1        | 95        | 1        |
| nasobeni      | 25        | 23        | 2        | 92        | 2        |
| deleni        | 24        | 22        | 2        | 92        | 2        |
| <b>CELKEM</b> | <b>83</b> | <b>78</b> | <b>5</b> | <b>94</b> | <b>2</b> |

```

840 COLOR 6,2
845 LOCATE 10,20:PRINT SPACES$(42)
850 v=TIMER
855 LOCATE 10,32:PRINT t;CHR$(t);j"=
      ";STRING$(LEN(k$),32);
860 u=CSRLIN
865 s=POS(0)
870 LOCATE u,s-LEN(k$):INPUT " ",k$
875 IF k$="" THEN 850
880 IF ASC(k$)<=57 AND ASC(k$)>=48 THEN 890
885 GOTO 850
890 k=VAL(k$)
895 COLOR 6,7
900 LOCATE 22,10:PRINT "Pocet chyb ="ch
905 RETURN
910 z=TIMER
915 zz=zz+(z-v)
920 m=5
925 ch=ch+1

```

```

1050 LOCATE 22,40:PRINT "Prumerny cas na priklad:";rz/pr
1055 COLOR 6,2
1060 m=4
1065 LOCATE 10,20:PRINT "D O B R E  pro pokracovani
      stiskni ENTER"
1070 d(e,2)=d(e,2)+1
1075 GOSUB 1135
1080 MENU ON
1085 zz=0
1090 a$=INKEY$
1095 ON MENU GOSUB 210
1100 IF a$=CHR$(13) THEN RETURN
1105 GOTO 1090
1110 LINE (0,120)-(639,120),3
1115 LINE (0,141)-(639,141),3
1120 LINE (127,64)-(503,86),2,bf
1125 LINE (55,150)-(585,177),7,bf '--- ctverec pro vysledky
1130 LINE (120,125)-(510,136),7,bf
1135 LINE (120,60)-(510,90),m,b '--- zmena barvy ramecku

```

```

1140 LINE (126,63)-(504,87),m,b '---
1145 PAINT (121,61),m
1150 RETURN
1155 CLS
1160 WINDOW CLOSE 2
1165 SCREEN CLOSE 1
1170 WINDOW CLOSE 1: CLEAR: END
1175 WINDOW CLOSE 2
1180 WINDOW 2," Pocty, vysledkove hodnoceni ",(0,0)-
(631,186),16,1
1185 FOR i=110 TO 630 STEP 104 '--- tabulkove, cary
1190   FOR j=50 TO 162 STEP 16
1195     LINE (1,50)-(1,162)
1200     LINE (1,j)-(630,j)
1205   NEXT j
1210 NEXT i
1215 LINE(1,50)-(1,162)
1220 COLOR 3,0
1225 LINE(1,146)-(630,146)
1230 LINE(1,82)-(630,82)
1235 COLOR 1,0
1240 LOCATE 3,30:PRINT "Vysledek tvoho snazeni"
1245 COLOR 4,0
1250 PRINT PTAB(220)STRING$(25,42)
1255 COLOR 2,0
1260 LOCATE 8,1:PRINT PTAB(27)"PRIKLADY";
1265 PRINT PTAB(135)"celkem";

```

```

1365 NEXT i
1370 FOR i=1 TO 5 '--- soucet vodorovne a vypocet
prumeru
1375   d(1,1)=d(1,2)+d(1,3)
1380 IF d(1,2)>0 THEN d(1,4)=d(1,2)/d(1,1)*100
1385 IF d(1,3)=0 AND d(1,2)>0 THEN d(1,4)=100
1390 NEXT i
1395 FOR i=1 TO 5 '--- znamky
1400   d(1,5)=1:c$="V Y B O R N Y"
1405 IF d(1,4)<95 THEN d(1,5)=2:c$="CH V A L I T E B N Y"
1410 IF d(1,4)<70 THEN d(1,5)=3:c$="D O B R Y"
1415 IF d(1,4)<50 THEN d(1,5)=4:c$="D O S T A T E C N Y"
1420 IF d(1,4)<20 THEN d(1,5)=5
:c$="N E D O S T A T E C N Y"
1425 IF d(1,4)=0 THEN d(1,5)=0
1430 NEXT i
1435 s=0:t=0
1440 FOR i=12 TO 20 STEP 2 :s=s+1:t=0 '--- vypis tabulky
1445   FOR j=16 TO 68 STEP 13:t=t+1
1450     IF d(s,t)=0 THEN 1460
1455     LOCATE i,j:PRINT USING "#####";d(s,t)
1460   NEXT j
1465 NEXT i
1470 MENU ON
1475 MENU 1,0,1," Nabidka "
1480 MENU 1,1,1," Navrat zpatky k pocitani "
1485 MENU 1,3,1," Konec programu "

```

Pocty, vysledkove hodnoceni

Celkova znamka ze zkouseni je:

CH V A L I T E B N Y

```

1270 PRINT PTAB(293)"z toho";
1275 PRINT PTAB(437)"uspesnost";
1280 PRINT PTAB(553)"znamka"
1285 LOCATE 10,1:PRINT PTAB(52)"na";
1290 PRINT PTAB(245)"dobre";
1295 PRINT PTAB(345)"spatne";
1300 PRINT PTAB(458)"v %"
1305 LOCATE 12,5:PRINT "scitani"
1310 LOCATE 14,3:PRINT "odecitani"
1315 LOCATE 16,4:PRINT "nasobeni"
1320 LOCATE 18,6:PRINT "deleni"
1325 LOCATE 20,5:PRINT "CELKEM"
1330 FOR i=1 TO 3 '--- nulovani svisle celkem
1335   d(5,i)=0
1340 NEXT i
1345 FOR i=1 TO 3 '--- soucet svisle celkem
1350   FOR j=1 TO 4
1355     d(5,i)=d(5,i)+d(j,i)
1360   NEXT j

```

```

1490 MENU 2,0,1,SPACE$(66)
1495 ON MENU GOSUB 1505
1500 GOTO 1495
1505 MENU OFF
1510 COLOR 1,0
1515 x=MENU(0):y=MENU(1)
1520 IF x=1 AND y=1 THEN WINDOW CLOSE 2:GOTO 20
1525 IF x=1 AND y=3 THEN 1530
1530 MENU 1,0,1,SPACE$(79)
1535 CLS
1540 COLOR 3,0
1545 LOCATE 9,25:PRINT "Celkova znamka ze zkouseni je:"
1550 LOCATE 14,1:PRINT PTAB(310-(8*LEN(c$))/2) c$
1555 FOR i=1 TO 12 STEP 2
1560   LINE (190-(2*i),99-i)-(430+(2*i),114+i),5,b
1565 NEXT i
1570 d$=INKEY$
1575 IF d$<>" " THEN 1155
1580 GOTO 1570

```



# SAFE HEX INTERNATIONAL

*Ondrej Krebs*

Počítačové vírusy. Neexistuje, hádam, jediný človek, ktorý by sa s nimi už nestretol a ktorému by neznepríjemnili život. Škody po ich vyčítaní sú často nesmierne. Mohli by o tom rozprávať pracovníci jedného nemenovaného nemeckého vydavateľstva, kde vírus Dasa (alias Byte Warrior) napáchal škody v hodnote 170 000 DM. Našťastie sa už autora tohto vírusu podarilo chytiť a usvedčiť; podľa nemeckého zákonodarstva ho čaká až 5 rokov nepodmienečne. U nás... škoda papiera. A žiaľ, nie len u nás. Len veľmi málo krajín sa podujalo na skutočný boj proti počítačovému terorizmu. A pritom najtriezvejšie odhady hovoria o tom, že do roku 1995 sa zvýši počet PC vírusov na takmer neuveriteľných 25 000 (dvadsaťpäť tisíc), my Amigisti na tom budeme o niečo lepšie, ale aj tak, ak aktivita tvorcov vírusov neprestane naberať také tempo, ako je tomu v poslednom čase, vzrastie počet ich výtvorov na 10 000.

Nie všetko je však až také zlé. Stále našťastie existujú ľudia neľutujúci čas a ani prostriedky na boj proti vírusom. Obzvlášť v oblasti Public Domain programov práve na naše Amigy. V tomto čase už rozvinutú a plne funkčnú medzinárodnú sieť na šírenie antivírusových programov nám môže závidieť nemálo užívateľov iných typov počítačov. A to zásluhou jedného pána z Dánska. Mnohí z vás ho poznajú cez jeho antivírusové kompilácie "The New Superkillers", kde sa podpisuje roztomilým "Speedy Gonzales" (znalci komiksov vedia, že je to najrýchlejšia myška v celom Mexiku). Práve tento pán, Erik Loevandahl Soerensen, inicioval vznik Safe Hex International, siete regionálnych centier bojujúcich proti vírusom, rozmiestnených už prakticky vo všetkých európskych štátoch a tiež v USA a Austrálii.

Ale teraz niečo bližšie k Safe Hex... Jej prvoradým cieľom je šírenie ochrany proti vírusom formou distribúcie národných verzií disku "The New Superkillers", osveta v tejto oblasti a nesmierne dôležitým bodom našej činnosti je i získavanie nových vírusov, aby mohli byť v čo najkratšom čase doručené autorom najznámejších viruskilerov, ktorým to umožní pohodlnejšie a hlavne flexibilnejšie aktualizovanie ich diel. Takže, ak narazíte na nový vírus a najnovšie viruskily ho nedokážu presne alebo vôbec identifikovať, pošlite mi ho. Pomožete tým nám všetkým a na diskete sa vám spať vrátia najnovší "The New Superkillers" v slovenskej verzii.

Okrem toho ak náhodou poznáte meno a ďalšie informácie o niekom, kým ste si istý, že vyrába alebo zámerne rozširuje vírusy, dajte mi o ňom vedieť. V prípade dokázania viny vás neminie odmena vo výške minimálne 1000 US\$. Diskutabilnou zostáva hlavne otázka dokázania viny, keďže, pokiaľ viem, náš trestný systém sa počítačovou kriminalitou nijako obzvlášť nezaobrá, ale nepochybne sa

to raz musí zmeniť. Kompilácia "The New Superkillers"... Na tejto diskete sa nachádza viacero zaujímavých programov. Najlepším z nich je a zrejme ešte dlho ostane BootX belgického autora Petra Stuera. Rozoznáva už približne 200 vírusov a jeho obsluha je natoľko jednoduchá, že ju bez problémov zvládnu i úplní začiatčníci. Z autorskej dielne tohto 23 ročného čerstvého absolventa elektrotechniky pochádzajú i ďalšie programy. LVD, priamo zabudovaný do BootXu a z tohto i spustiteľný, je zaujímavý tým, že obsadí ten bod operačného systému Amigy, ktorým sú do pamäte naťahované všetky spustiteľné programy. Takýmto spôsobom musí každý spúšťaný program prejsť nekompromisnou kontrolou LVD a keď je objavené niečo podozrivé, eventuálne i známy vírus, programu, nie je dovolené rozbehnúť sa, čo zabezpečuje optimálnu ochranu proti šíreniu týchto malých beští. Za zmienku stojí i nemecký VT z dielne Heinera Schneegolda.

I on, podobne ako BootX, je odtý do slušivého 3D rúcha, čo i majiteľom Amig so systémom 1.2/1.3 umožňuje vychutnať prostredie podobné Kickstartu a Workbenchu 2.0. Obsluha VT je hádam ešte jednoduchšia ako je tomu u BootXu.

Od nórskeho autora Arthura Hagena pochádza CLI viruskiler VScan. Je naprogramovaný na prácu nie v okienkovom režime, ale v CLI a je obzvlášť vhodný na kontrolu harddiskov. Aj tento program obsahuje všetko, čo má obsahovať dobrý viruskiler, teda i účinný analyzácný mód na odhalenie neznámych súborových vírusov. Arthur Hagen je už dlhšiu dobu sužovaný reumatizmom, tak mu držme palce nech, svetlo sveta uzrie ešte veľa nových verzií VScanu.

Kapitolu samu o sebe predstavuje ZeroVirus austráľčana Jonathana Pottera. Tento program bol dlhý čas oslavovaný ako to najlepšie, čo bolo kedy vytvorené. V súčasnosti je už ZV naprosto neaktuálny, autor ho za posledných 9 mesiacov vôbec neaktualizoval. Vinu na tom však majú jednoznačne samotní užívatelia. A to nie je prípad iba ZeroVirusu.

Už mnoho shareware programov stroskotalo na neetickom prístupe užívateľov. Tu mi nedá trocha bližšie priblížiť ideu shareware programov. Je založená v prvom rade na férovom prístupe užívateľov. Autor svoj program dá voľne do obehu, môže byť kopírovaný a rozmnožovaný, ale ak sa ho niekto rozhodne seriózne používať, mal by zaplatiť autorovi onen skutočne veľmi nízky, skor symbolický poplatok, ktorý predstavuje len nepatrný zlomok skutočnej hodnoty programu.

Nebýva to viac ako 10 - 20 US dolárov, čo si myslím, že je suma, ktorú si môže dovoliť naprosto každý. Smutným faktom však stále zostáva, že túto etickú povinnosť si plní len veľmi, veľmi málo ľudí. A preto sa autorom nesmieme diviť, že sa na to vykašľú, veď i oni potrebujú z niečoho žiť a tisíce hodín strávené pri vytváraní programu sa im málokedy zhodnotia i finančne. Takže neváhaj už ani chvíľku!

Ak si aj ty jedným z tých, ktorí používajú shareware programy, uposlúchni čo ti káže etika a pošli autorom peniaze. Alebo chceš, aby už na svet nikdy neprišli žiadne nové verzie a nové programy?

**REGIONÁLNE VÍRUSOVÉ CENTRUM  
ČESKOSLOVENSKO**

Ondrej Krebs  
Síd. SNP 4  
908 51 Holíč

telefón 0801/3764

P.S. Na tejto adrese môžete získať vždy najnovšiu aktualizáciu "The New Superkillers", ktorá je v slovenskej verzii aktualizovaná spravidla každý mesiac a sem sa môžete či už telefonicky alebo písomne obrátiť s vašimi vírusovými problémami.

Cena diskety s týmto programom je 65.- Kčs vrátane poštovného.

Autor se vzdal honoráře ve prospěch konta Míša.

# Program 60Hz + SetPatch pro 1 MB Chip Ram.

*Ivo Janáček*

Jelikož mám Amigu s 1MB CHIP RAM, vznikají mi problémy se systémovými vektory, protože Kickstart1.3 je dělaný pro 0.5MB CHIP RAM. V důsledku toho se mi stávalo, že při zhroutení systému, když jsem měl nainstalován RamDrive, se tento nenávratně ztratil. Další problém vznikl při použití programu pro zobrazení v 60Hz. Proto jsem se poradil s vynikajícím coderem Přechkem a pak jsem se dal do práce. Nejdříve jsem změnil program 60Hz, který teď sedí na jiném vektoru. Dále následovala mravenčí práce s vytažením SetPatch pro 1MB CHIP RAM, což se mi nakonec povedlo. Výsledný program je krátký a je odladěný na překladači Genam2. Velkou výhodou je, že tento program se vždy spustí po startu systému a ihned opraví vektory, takže při použití RamDrivu nemusíte mít strach.

ProgStart

LEA \$100.W,A0 ;Počátek umístění programu  
LEA Start(PC),A1  
MOVEQ #Delka,D1

Copy MOVE.B (A1) + ,(A0) + ;Kopírování na \$100  
DBRA D1,Copy

MOVE.L 4.W,A6  
MOVE.L #\$100,\$2A(A6) ;Nastavení vektoru  
LEA \$22(A6),A0  
MOVEQ #0,D2  
MOVEQ #\$17,D1

KickSum ADD.W (A0) + ,D2 ; Obnovení kontrolního součtu

DBRA D1,KickSum

NOT.W D2  
MOVE.W D2,(A0)  
MOVE.B #0,\$DFF1DC ; Zapnutí 60Hz  
RTS ;Návrat

Start MOVE.L 4.W,A6  
MOVE.L #\$100,\$2A(A6) ;Nastavení vektoru  
MOVEQ #0,D1  
LEA \$22(A6),A0  
MOVE.W #\$16,D0

SumKick ADD.W (A0) + ,D1 ; Obnovení kontrolního součtu

DBRA D0,SumKick

NOT.W D1  
MOVE.W D1,\$52(A6)  
CMP.L #\$B7FC0004,\$1E(A5) ;SetPatch pro  
1 MB Chip RAM  
MOVE.B #0,\$DFF1DC ;Zapnutí 60Hz  
JMP (A5) ,Návrat na (A5)

Delka equ \*-Start

Program 50Hz + zrušení SetPatch pro 1 MB Chip Ram.

Toto je program, který provede přepnutí do 50Hz zobrazení a zároveň vypne i SetPatch. Ten kdo chce používat pouze část SetPatch a vyzná se trochu v assembleru, sám si může program upravit.

MOVE.B #\$20,D1  
MOVE.B D1,\$DFF1DC ;Zapnutí 50Hz  
MOVE.L 4.W,A6  
MOVE.L #0,\$2A(A6) ;Nulování vektoru  
RTS



## INZERCE

**Firma Antik Soft nabízí  
pro Amigu 500, 2000 a 3000.**

### MAX Sound

Sampler editor

Operace s bloky, úprava zvuku (zařízení je vybaveno hardwareovými filtry), efekty v reálném čase, sampling, přehrávání zvuku a záznam zvuku.

Informace a objednávky na adrese:

**ANTIK Soft**  
S. N. P. 16,  
040 11 Košice  
tel: 095/34356

# Deluxe Paint III

*Ivo Janáček*

Část 6.

Opět se scházíme u dalšího pokračování návodu, jak pracovat s DeluxePaintem III. Dnes nastala ta doba, kdy se konečně pustíme do probírání animace.

Nejdříve si ale povíme několik základních informací, které by každý začínající animátor měl vědět.

Nejjednodušší animace lze vytvořit pomocí rotace barev. V tomto případě musíte kreslit jen jeden obrázek, kde vhodným umístěním barev vyvoláte dojem pohybu. Pro tuto práci vám bude stačit dokonce jen 0.5MB paměti. Pokud však budete chtít vytvořit doopravdivou animaci, budete potřebovat nejméně 1MB paměti, která vám vystačí jen na velmi krátkou animaci v malém rozlišení a s minimem barev. Takže zde platí přímá úměra, že čím víc, tím líp. Musíte si totiž uvědomit, že v paměti je uložen každý obrázek animace. Další věc, kterou byste měli vědět, je, že celá animace může mít pouze jednu paletu barev a že všechny obrázky musí mít stejnou velikost a počet barev. To platí i pro dotahování již hotových výtvarů do rozpracované animace. Dále se zde chci zmínit o tzv. AnimBrush, čili animovaném štětcu. Nejedná se o celoobrazovou animaci, ale o animaci štětce, která je vhodná pro přípravu menších animací, které pak lze převést do animace skutečné. Ukázky animovaných štětců jsou na disketě ukázek v adresáři AnimBrush a spouští se klávesami 7 a 8. Na konec ještě jedno doporučení: Provádějte si zálohování animace po každém větším úspěšném zásahu do ní a pro zkoušení kresby použijte obrazovku Spare.

## Hlavní menu Anim

### Menu Load

Toto menu slouží k natažení animace z disku. Okno je shodné s oknem pro natažení jednoho obrázku. Jediný rozdíl je v tom případě, máte-li část animace hotové na disku a chcete ji přihrát k animaci, kterou právě máte v počítači. Pro tento případ je zde volba "Append", která provede přihrání animace z disku do paměti počítače. Pozor na velikost animací, jejich palety a také na rychlost přehrávání! Natažená animace se připojí za dosavadní. Nemáte-li v počítači animaci, volba "Append" se neobjeví.

### Menu Save

Toto menu slouží k uložení animace nebo její části na disk. Okno je téměř shodné s oknem pro uložení jednoho obrázku. Navíc je tady ještě možnost uložit na disk pouze určité obrázky pomocí volby frames, kde zadáte, od kterého do kterého obrázku se má animace uložit.

## Menu Move, "M"

Toto je velmi důležité menu, pomocí kterého lze provádět nejrůznější pohyby a zoomy (změna ohniskové vzdálenosti objektivu). Brzy zjistíte, že nebýt tohoto menu, tak by DeluxePaint III neměl takový úspěch, jaký má nyní. Tak tolik na úvod a teď si toto menu popíšeme podrobně. Menu lze použít i při tvorbě jednoho obrázku, tzn. bez animace.

110) Distance, neboli Vzdálenost předmětu. Tato volba určuje pohyb předmětu v prostoru ve třech směrech. Volba Brush určuje, jak se bude animace provádět.

111) Angle, čili Úhel. Tato volba určuje, jak se bude předmět otáčet. Zadává se vždy úhel, o který se má předmět otočit za celou animaci. Volba Brush určuje, jakým způsobem se bude rotace provádět.

112) Go Back, Jdi Zpátky, slouží k nastavení posledních hodnot pro ovládání animace.

113) Clear, čili Smaž, provede vynulování hodnot pro řízení animace.

114) Cyclic, neboli Opakovaně. Tato volba způsobí, že se animace bude opakovat tak dlouho, dokud se nezaplní všechny obrázky animace.

115) Ease-Out provede zpomalení začátku animace, které se bude postupně zmenšovat až do obrázku s hodnotou, která je uvedena za touto funkcí.

116) Ease-In provede zrychlení začátku animace, které se bude postupně zmenšovat až do obrázku s hodnotou, která je uvedena za touto funkcí. Obě tyto funkce se aktivují zapsáním hodnot do volných míst. Najednou lze použít pouze jednu z nich.

117) Counter neboli Počítadlo udává, do kolika obrázků se má pohyb rozkouskovat. Maximem je délka aktuální animace.

118) Direction, čili řízení. Volba Move udává, jak se bude provádět pohyb animovaného předmětu. Levé nastavení znamená, že se pohyb uskuteční od aktuálního umístění směrem k umístění, určenému zadanými hodnotami pro animaci.

Pravé nastavení znamená, že se animace bude zaznamenávat od místa určeného hodnotami pro animaci až po aktuální umístění. Druhá volba Record udává, jak se bude animace vypočítávat (ne zaznamenávat). Levá volba znamená, že se bude počítat od aktuální pozice směrem ke koncové pozici. Prostřední volba znamená, že se celá animace zaznamená pouze do aktuálního obrázku animace. Pravá volba znamená, že se nejdříve zjistí, jak bude animovaný objekt vypadat na konci animace, a pak se bude zpětně vypočítávat pohyb směrem k aktuální pozici.



Pomocí volby Record lze provést přesné umístění začátku nebo konce animace.

119) Preview je asi nejčastěji používaná volba v okně Move, neboť slouží k prohlédnutí animace ve formě drátěného modelu ještě před vlastním výpočtem.

Pomocí této volby si můžete ušetřit spoustu drahocenného času, protože vypočtení animace trvá dost dlouho. Takže, kdo nemá turbo kartu aspoň na 14 MHz, ať raději používá Preview. Brzy by se mi měla pravděpodobně do Amigy dostat první Moravská TurboBoard na 16 MHz, takže uvidím, jak to potom bude šlapat.

120) Draw slouží ke startu výpočtu animace. Výsledek výpočtů můžete průběžně sledovat na obrazovce. Zde bych chtěl upozornit na to, že čím větší máte horizontální rozměr obrazovky (rozlišení LoRes nebo HiRes, netýká se Interlace), tím pomaleji probíhají výpočty, protože Agnus blokuje MC68000, aby mohl z paměti číst údaje, které má zobrazovat. Takže pokud si chcete ušetřit čas a nemáte doopravdivou FAST RAM, přepněte na jinou obrazovku, která má rozlišení LoRes (320 bodů).

121) Cancel slouží k návratu do programu bez zapamatování provedených změn v řízení animace. Při dalším otevření okna budou nastavena taková čísla jako při minulém otevření.

122) Exit provede zapamatování provedených změn a návrat do programu. Při dalším otevření okna budou nastavena taková čísla jako při použití Exit.

123) Trails provede výpočet animace s tím rozdílem, že nejdříve okopíruje poslední vypočtený obrázek do dalšího, a pak teprve počítá další obrázek.

124) Fill provede vyplnění obrázku animovaným objektem tak, že se pohybuje jakoby celá obrazovka. Tato volba je časově velmi náročná.

Tak jsme si popsali všechny funkce okna Move, ale abyste je uměli dobře používat, bude třeba si všechno důkladně odzkoušet a "osahat". Doporučuji nezačínat pozdě večer, protože se nevyspíte a brzy zjistíte, že už je ráno.

Abych vás trochu navnadil, uvedu nyní jeden příklad, ve kterém si provedeme některé z dříve popsaných funkcí menu Move. Nejdříve si však popíšeme menu Frames a menu Control, abyste mohli nastavit délku animace a abyste si ji pak uměli přehrát.

## Menu Frames

### Funkce Add Frame

Tato funkce provede přidání jednoho obrázku za aktuální obrázek. Obsah nového obrázku bude shodný s aktuálním obrázkem.

## Menu Set #

Toto menu otevře okno "Set Frame Count", které má jen jednu nastavovací volbu "Count:". Tato volba nastavuje celkový počet obrázků v animaci. Pomocí ní lze provést přidání nebo i ubrání obrázků v animaci od aktuální pozice. Jestliže zvětšíte počet obrázků v animaci, provede se přidání v aktuálním místě zároveň s okopírováním aktuálního obrázku do všech přidávaných. Provedete-li zmenšení počtu obrázků v animaci, provede se smazání v aktuálním místě animace.

### Funkce Copy to all

Tato funkce provede okopírování aktuálního obrázku do všech obrázků animace.

### Funkce Delete Frame

Tato funkce provede smazání aktuálního obrázku animace. Po tom, co tuto funkci zvolíte, budete ještě otázaní, zda tuto operaci chcete doopravdy provést.

### Funkce Delete all

Tato funkce provede smazání celé animace. Po zvolení této funkce musíte opět své rozhodnutí potvrdit, (co kdybyste se z toho všeho nakonec zbláznili!).

## Menu Control

### Menu Set Rate

V tomto okně "Set frames per second" se zadává rychlost přehrávání animace.

Maximální hodnota je 50 obrázků za sekundu.

### Menu Set Range

V tomto okně "Set Play Range" se provádí nastavení rozsahu animace, který se bude přehrávat. Do volných míst můžete nastavit rozsah animace, který se má přehrávat. Potom musíte aktivovat volbu "Range" a zmáčknout "OK". Chcete-li, aby se přehrávala celá animace, zmáčkněte volbu "All Frames" a "OK". Pro zrušení nastavení slouží samozřejmě "Cancel".

### Funkce Previous, "1"

Provede posun v animaci na předcházející obrázek.

### Funkce Next, "2"

Provede posun v animaci na následující obrázek.

### Menu Go to, "3"

Toto menu otevře okno "Go to Frame", ve kterém můžete nastavit obrázek na který se chcete přesunout. Pak

"OK", jinak "Cancel".

## Funkce Play, "4"

Provede přehrávání animace v nastaveném rozsahu obrázků. Animace se bude neustále opakovat až do zmáčknutí "ESC" nebo myši.

## Funkce Play once, "5"

Provede jedno přehrávání animace.

## Funkce Ping-pong, "6"

Tato funkce provede přehrávání animace od začátku do konce a potom od konce do začátku, což se bude opakovat do té doby, než animaci přerušíte již známými způsoby.

Tak jsme se prokousali dalšími příkazy hlavního menu Anim a nadešel čas na slíbený příklad. Dobře se posadte, spusťte počítač a spusťte DeluxePaint III.

Nastavte si rozlišení LoRes a dvě barvy. Teď si zadejte počet obrázků animace, kterých nám bude stačit asi tak padesát. Dále si najděte nějaký pěkný font, veliký asi tak 30 pixelů (bodů). Napište na obrazovku nějaký text (jeden řádek). Nyní tento text vystřihněte tak, aby se na obrázku

vymazal (pravou myší). Můžete si změnit paletu barev na černou a žlutou, aby to všechno vypadalo efektněji. Po té umístíte text pěkně doprostřed obrazovky a otevřete okno Move. Do volného místa Distance Z vepište číslo 5000, na místo Ease-Out vepište počet obrázků vaší animace a totéž vepište i na místo Count.

Potom již můžete zmáčknout Draw. Za chvíli bude vše hotovo a můžete zmáčknout na klávesnici čtyřku nebo šestku. Pokud jste se dost vynadávali a už vás to nebaví, vymažte obsah animace pomocí CLR z pravého grafického menu pomocí "All Frame". Teď napište na místo Angle X číslo 1080 a vypněte horní "Brush" v Distanci. Znovu dejte Draw a pak se podívejte na váš výtvar. Nezapomeňte před zadáním Draw opět umístit váš vystřižený štětec do prvního obrázku animace pomocí levé myši tak, aby se skutečně do tohoto obrázku nakreslil! Tak a víc vám nepovím, protože bude lepší, když si teď budete chvíli sami experimentovat.

Tak to bylo pro dnešek všechno. Doufám, že začnete vytvářet své animace, které mohou dosahovat kvality kreslených filmů. Příště si povíme něco o tom, jak používat AnimBrush a jak jej pak umístit do své animace. No a potom nás čeká předposlední menu Effect. Přeji hodně zdaru a pevné nervy při probdělých večerech u tvoření animací (je 2:10 v noci).

*Pokračování příště.*



## Nastavení znakové sazby pomocí DPaintu

SDPaintem můžeme v první řadě použít tu znakovou sazbu, která je uložena na bootovací disketě. Jde to ale i jinak. Překopírujte si všechny fonty, které jsou k dispozici na Workbenchu v adresáři "fonts", DPaint-disketě nebo na jiném disku na disketu novou, kterou nazvete např. "ALLFONTS".

Prostřednictvím textového editoru např. Ed nebo MicroEmacs natáhněte "startup-sequenci" originální DPaint-diskety, případně její bezpečnostní kopie a vytvořte řádek:

**assign fonts: allfonts:**

Startup-sequenci uložte. Po bootování takto upravené diskety požaduje Amiga vložení diskety fontové. K psaní pak mohou být použity veškeré znakové sazby.

## Request - jen s pauzou

Je-li aktivován Requester s funkcí Intuition Request() (obsahuje více struktur Image, String-gadgetů a jeden Boolean - Gadget), potom se funkce Request() zpětně ohlásí předčasně, což znamená, že program běží dál, ačkoliv Requester není ještě hotov.

Nepříjemné pak ale je, když v této souvislosti má být např. aktivován jeden ze String-gadgetů, což přirozeně není možné.

Pomoci si můžeme vynucenou přestávkou mezi Request() a ActivateGadget(), realizovanou prostřednictvím Delay().

### Příklad:

```
Request(&RequesterStructure1, my_window);
Delay (5L);
ActivateGadget(&Gadget1, my_window, &RequesterStructure1);
```

Hodnota, kterou předáme funkci Delay(), musí být podle potřeby odladěna na jiné konfiguraci počítače.

*(Volně zpracoval podle časopisu AMIGA M&T P.Š.)*

## 703 KB v Amiga Transformeru

Občas potřebuji něco udělat na ATěčku a nechce se mi kvůli tomu jezdit ke známým a zbytečně je obtěžovat. Proto občas používám Amiga Transformer, verzi 1.23 pro Amigu 2000. Jelikož mám 2.5 MB paměti, chtěl jsem si v preferencích nastavit aspoň 2 MB paměti pro DOS. Bohužel toto není možné provést, protože se na tuto možnost při tvorbě programu nepamatovalo.

Maximální hodnota je 640 KB, což není mnoho.

Proto jsem si řekl, že zkusím změnit soubor ATPREFS "natvrdo".

Natáhl jsem si FileMaster, našel příslušnou hodnotu a změnil na 1 MB. Potom jsem spustil Transformer. Ten najel a ukázal jedno mega volné paměti. Při vložení DOS diskety však systém házel stále chybu, že se nejedná o systémový disk. Po delších testech jsem zjistil, že lze zvětšit volnou paměť aspoň o 64 KB na 703 KB.

Je to sice malé zvětšení, ale někdy se vám může hodit. Takže si nahrajte soubor ATPREFS do FileMasteru (nebo jiného hexa/ascii editoru), najděte si 59 a 60 bajt v souboru (3B a 3C hexa), jejichž hodnotu změňte na 02BF hexa. Pak soubor uložte a spusťte Transformer. Na úvodní obrazovce se vám objeví údaj o 703 KB volné paměti. (I.J.)

# transgas NABÍZÍ:

## TRANSGAS je tu pro Vás

Pro řešení problematiky proudění plynu a jiných médií v potrubních systémech nabízíme:

- znalosti získané 18-letou zkušeností v oblasti vývoje metod simulace, optimalizace, sběru dat, výpočtů pro vyhodnocení provozu i jejich úspěšné aplikace na soustavě tranzitních plynovodů v ČSFR i v zahraničí,
- výkonný „software package“ pro simulaci proudění plynu SIMONE v ČSFR i v zahraničí (SRN, Maďarsko, Dánsko atd.),
- vlastní software typu SCADA pro sběr dat a řízení rozsáhlého technologického systému,
- programy pro optimalizaci provozních nákladů a dalších parametrů, optimalizaci provozních postupů atd.,
- veškerý aplikační software pro využití při dopravě plynu a jiných médií,
- aplikace zabezpečíme podle vašich požadavků dodávkou obecného software nebo dodávkami „na klíč“, náš software zařídíme do vašeho dosavadního systému, vyvineme vhodnou metodu pro řešení vašeho problému, poskytneme konzultaci.

**PIŠTE, VOLEJTE, NAVŠTIVTE NÁS!**

# transgas

Naše adresa zní: Tranzitní plynovod, k. p., Praha  
 Odbor technických služeb a zakázek  
 Štěpánská 28  
 113 94 Praha 1  
 tel. č.: 83 97 87  
 82 93 40



# LATTICE C

## Development System v. 5

*Martin Helmich*

Část 4.

Jestliže strojový podprogram definuje umístění dat přístupných z C-modulů, měly by se v datové sekci vyskytovat :

### CSECT data

Všechny datové elementy musí být deklarovány instrukcí XDEF, aby mohly být přístupné z C-modulů :

|      | XDEF | VAR1,VAR2,VAR3 |
|------|------|----------------|
| VAR1 | DC.L | \$4000         |
| VAR2 | DC.W | \$8000         |
| VAR3 | DC.L | D0             |

Registry D2 až D7 a A2 až A6 musí být modulem chráněny a návratová hodnota musí být uložena v příslušném registru.

Aby bylo možno zavolat C-funkci z assembleru, je jí nutno deklarovat instrukcí XREF (a samozřejmě připravit k předání všechny argumenty) :

|        |                                         |
|--------|-----------------------------------------|
| XREF   | c_funkce                                |
| MOVE.L | D0,-(A7) ;uložení argumentů             |
| MOVE.L | D1,-(A7)                                |
| JSR    | c_funkce ;volání funkce                 |
| ADDQ   | #8,A7 ;obnovení zásobníkového ukazatele |

Datové elementy definované v C-modulech mohou být dosaženy přes XREF :

|        |         |
|--------|---------|
| XREF   | XD2,XD3 |
| .      | .       |
| .      | .       |
| MOVE.L | XD2,D0  |

### Instrukce CSECT

Instrukce CSECT specifikuje, do které řídicí sekce bude zahrnut

následný kód. V předešlé kapitole byly za instrukcí uvedeny dva standardní parametry, a to "text" a "data". Pomocí volby -s kompilátoru (příkazu lc) můžete jednotlivé sekce pojmenovat (jinak nemají své jméno). Použijete-li pouze volbu -s, programová sekce bude pojmenována jako "text", datová sekce jako "data" a neinicializovaná datová sekce (bss) jako "udata". Je-li volba -s následována jedním z písmen b, c nebo d a znakem "=", je možno zadat jméno jiné (-sb = jm1 pro neinicializovanou datovou, -sc = jm2 pro programovou a -sd = jm3 pro datovou sekci. Formát instrukce je :

### CSECT name,type,align,rtype,rsize

Pouze parametr "name" je povinný pro určení jména řídicí sekce.

Parametr "type" nabývá hodnot 0, 1 a 2. Hodnota 0 definuje strojovou, parametr 1 datovou a parametr 2 neinicializovanou datovou sekci. Neuvedení parametru je ekvivalentní hodnotě 0.

Parametr "align" specifikuje požadavky řazení řídicí sekce, hodnota 1 zajišťuje adresování po slovech (2 bajty), hodnota 2 po 4 bajtech.

Parametr "rtype" určuje typ přemístění, který podmiňuje význam adres, nebo přemístění dat. (neuvedení parametru jako 0), viz. tabulka.

Parametr "rsize" určuje rozsah kódu (dat) v bajtech, který bude zahrnut do příslušné sekce.

| rtype | Rozsah v bajtech | Význam                                    |
|-------|------------------|-------------------------------------------|
| 0     | 2 nebo 4         | absolutní adresa (short nebo long)        |
| 1     | 2                | relativní vzhledem k PC (program counter) |
| 2     | 2                | relativní vzhledem k adrese v registru    |

### Relativní adresování

Funkce PC-relativní mohou být užívány následujícím způsobem :

|       |       |             |
|-------|-------|-------------|
|       | CSECT | TEXT,0,,1,2 |
|       | XREF  | cfunc       |
|       | XDEF  | afunc       |
| afunc | JSR   | cfunc(PC)   |
|       | RTS   |             |

Relativní adresování pomocí registru :

|        |              |             |
|--------|--------------|-------------|
|        | CSECT        | data,1,,2,2 |
|        | XREF         | xdata       |
| data1  | DC.W         | 10          |
| data2  | DC.L         | 20          |
| MOVE.W | data1(A5),D0 |             |
| MOVE.L | data2(A5),D1 |             |
| MOVE.L | d1,xdata(A5) |             |

### Makro assembler

Lattice assembler načte zdrojový text a vytvoří "object file" ve formátu souborů s koncovkou ".o". Zdrojový text musí mít koncovku ".a". Linker poté assembler s C spojí ve spustitelný soubor.

### Formát zdrojového textu

Tak jako každý assembler, i Lattice má následující formát řádku :

návěští: operace operandy ;komentář

Položka návěští není povinná. Je-li uvedena, musí se bezprostředně za ní, nacházet dvojtečka. Návěští může být maximálně 31 znaků dlouhý řetězec sestávající z písmen, číslic, podtržítka a dolaru ( \_ \$ ), začínající písmenem. Velká a malá písmena jsou považována za odlišné, t.j. mh je různé od MH.

Položka operace obsahuje jméno instrukce, příkazu assembleru nebo makra.

Řádek nesmí začínat operací, pokud nezačíná návěstím, musí začínat mezerou.

Velká a malá písmena jsou totožná.

Položka operandů neobsahuje nic, nebo obsahuje konstanty, proměnné a operátory.

Konstanta může být zapsána následujícími způsoby :

| Číslo        | Reprezentace                                  | Příklad   |
|--------------|-----------------------------------------------|-----------|
| Desítkové    | řetězec desítkových číslic                    | 1234      |
| Šestnáctkové | \$ následovaný řetězcem šestnáctkových číslic | \$89AF    |
| Osmičkové    | @ následovaný řetězcem osmičkových číslic     | @7541     |
| Binární      | % následované jedničkami a nulami             | %10101001 |
| ASCII        | maximálně čtyři ASCII znaky                   | "AC9T"    |

Proměnná je jméno návěští nebo jméno definované příkazem assembleru.

Operátor může být některý z těchto znaků :

| Priorita | Operator | Význam                            |
|----------|----------|-----------------------------------|
| 1        | -        | unární minus                      |
| 2        | >>, <<   | posunutí doprava (doleva)         |
| 3        | *, /     | násobení a dělení                 |
| 4        | +, -     | sčítání a odčítání                |
| 5        | <, <=    | menší,                            |
|          | >, >=    | menší nebo rovno                  |
|          | =, !=    | větší,                            |
|          |          | větší nebo rovno                  |
| 6        | &, !     | rovno, nerovno                    |
|          |          | AND, OR (logický součin a součet) |

Komentář je libovolný text za středníkem.

## Používání assembleru

Assembler je volán následujícím příkazem v CLI :

asm [>listfile] [options] filename

Položka ">listfile" určuje, do kterého souboru se má vypisovat chybové hlášení.

Položka "options" se sestává z pomlčky následované řídicími znaky. Jejich význam je následující :

-c

Tato volba specifikuje, která programová sekce adresovatelná uživatelským hardwarem má být nahrána do paměti CHIP RAM.

Volba -c musí být bezprostředně následována jedním nebo více písmeny :

- b - neinicializovaná datová sekce
- c - strojová sekce
- d - inicializovaná datová sekce

-d

Tato volba způsobuje vkládání informací pro debugging do objektového modulu (do souboru s koncovkou .o). Další využití je k definování symbolů :

-dABC

Definuje ABC tak, jako by na prvním řádku zdrojového souboru byl příkaz :

ABC EQU 1

-dABC = 10

Definuje ABC tak, jako by na prvním řádku zdrojového souboru byl příkaz :

ABC EQU 10

-h

opak volby -c, t.j. které sekce mají být nahrány do paměti FAST RAM. Není-li volba použita, vykoná se automaticky :

-hbcd

-iprefix

Tato volba specifikuje, ve kterých direktoráři mají být hledány soubory uvedené v příkaze INCLUDE. Použijete-li například :

asm -iMYINC/ -iYOURINC/ prog.a

a obsahuje-li zdrojový text příkaz :

INCLUDE ABC.I

tak assembler hledá nejprve soubor ABC.I v aktuálním direktoráři, a pokud ho nenajde, hledá MYINC/ABC.I a YOURINC/ABC.I

-l[list]

zajišťuje výpis zdrojového a přelože-

ného textu do standardního výstupu, volba může být následována :

- l - vypíše zdrojový text i text z INCLUDE souboru
- m - umožňuje několikanásobný výpis řádku
- x - vypíše rozvinutí makra

-m0

Umožní jen použití instrukcí procesoru 68000.

-m2

Je možno použít i instrukce procesoru 68020.

-oprefix

Specifikuje soubor, do kterého má být uložen výsledný kód.

-s

Tato volba ukládá název sekce na začátku každé části.

-u

Tato volba očekává před každou funkcí podtržítka (\_).

Položka "filename" specifikuje zdrojový soubor, a pokud není použita volba

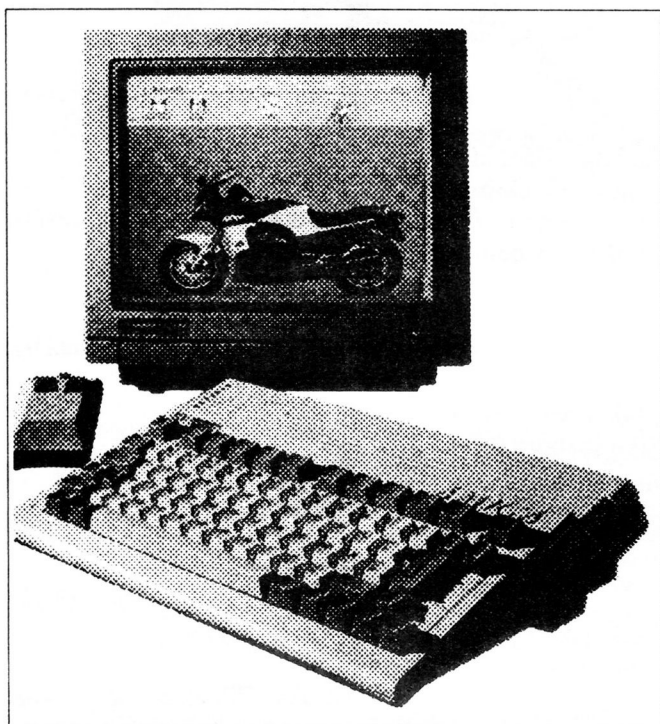
-o, tak i jméno cílového souboru (koncovka .a je nahrazena .o).

## Příkazy assembleru

- CNOP - podmínkový NOP pro řazení;
- DC.B - definuje konstantu v paměti, jejíž rozměr je 1 bajt;
- DC.L - definuje konstantu v paměti rozsahu long;
- DC.W - definuje konstantu v paměti o rozsahu jednoho slova (2 bajty);
- DS.B - definuje uložení bajtu;
- DS.L - definuje uložení dlouhého slova (4 bajty);
- DS.W - definuje uložení slova;
- DSECT - nastavuje lokální čítač na datový segment, ekvivalentní příkazu :  
CSECT data
- ELSE - "jinak", součást složeného podmíněného příkazu;
- END - konec programu;
- ENDC - konec podmínkového překladu;
- ENDM - konec definice makra;
- EQU - přiřadí stálou hodnotu;
- EXITM - opuštění makra;
- IDNT - jméno programové jednotky;
- IF - přeloží následující text až do prvního výskytu ELSE nebo ENDIF, pokud je podmínka splněna;

Pokračování příště.

# AMIGA 600 a AMIGA 600 HD



Firma COMMODORE představila na letošním CeBITu, mimo jiné, novinku v oblasti domácích počítačů a nazvala ji AMIGA 600.

Čím se A600 liší od A500.

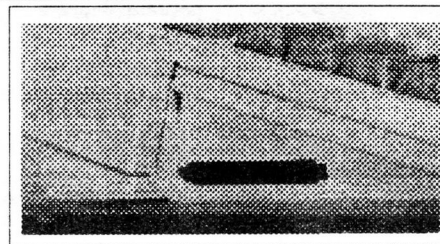
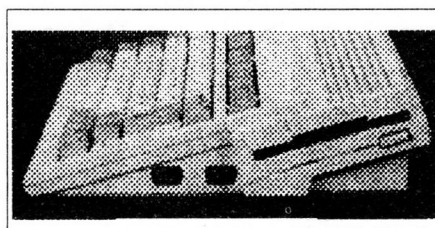
V první řadě je to designe, připomíná svoji koncepcí tak trochu laptop, zejména rozložením kláves na klávesnici. Dále pak u typu A600 HD je zabudován harddisk. Tímto krokem jsou tyto počítače tak trochu vyjímány mezi domácími computery. To, ale ještě není všechno, do počítače je možné zasouvat z levé strany speciální paměťové karty t.zv. Flash-Memory-Card, kterými bude možno značně rozšiřovat možnosti tohoto počítače. Není nutné se ale obávat toho, že by nová Amiga nebyla kompatibilní se starší řadou.

Nyní bych se chtěl ještě zastavit u konstrukce A600 a A600 HD, jak jsem se již zmínil je koncepčně nová. Co to mimo jiné znamená, je postavena novou technologií tzv. SMD technikou. Pro neznalé jedná se o technologii, která pájí součástky přímo na povrch tištěného spoje bez použití drátových vývodů. Domácím kutilům se tato koncepce příliš líbit nebude, protože bez speciálních doplňků nebude možné bezpečně pájet a tedy velice snadno dojde k poškození spoje a součástek.

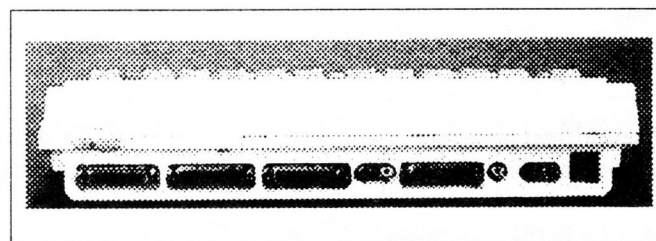
Znovu bych se vrátil k A600 HD, která jak jsem již uvedl má zabudován interní harddisk. Jedná se novou konstrukci 2,5" harddisků s kapacitou od 20 do 120 Mb.

Tímto a celou řadou dalších doplňků je tak trochu předurčena pro specialisty na grafiku. Jen namátkou uvedu nové zákaznické obvody AGNUS a DENISE, které jsou shodné s provedením v A3000. S obvodem ECS-Denise je možné zpracovávat obrazový signál v nových grafických módech např. HAM-, Super-Hires- a Produktivity-Modus.

Další dva obrázky ukazují A600 z pravého a levého boku, kde si můžete všimnout většího úhlu sklonu celého počítače. Vzhledem k tomu, že jsme neměli možnost s tímto počítačem delší dobu pracovat, tak vám nemůžeme říci je-li tato úprava výhodná.



Podíváme-li se na počítač v pohledu ze zadní strany zjistíme, že umístění konektorů jednotlivých rozhraní je prakticky stejné s A500.



Co je však u A600 zcela nové a stojí za zmínku tj. nový HF modulátor, který je umístěn uvnitř přístroje a je převzatý z CDTV. Pro ty, kteří nebudou mít speciální monitor a budou tedy používat pouze televizi, je tento modulátor velkým přínosem.

Pro úplnost je nutno říci, že A600 a A600 HD pracují pod Kickstart 2.0 a Workbench 2.0.

Na závěr vám můžeme sdělit, že uvedené počítače budou v nejbližší době díky Fy A-B-Comp uvedeny na náš trh a budou tedy běžně dodávány zásilkovým prodejem naší redakce.

Další podrobnosti spolu s testem uvedeme v některém z příštích čísel našeho časopisu.

(ZD)



# GENLOCK

## A 8802

*Tomáš Stibor*

Další z mnoha přídatných zařízení, nabízených k prodeji naší redakcí, jsou i genlocky. Dnes vás chceme seznámit s jedním z nich a pro laiky popsat, k čemu se tato zařízení vlastně používají.

Nejprve je nutno vzít na vědomí, že Amiga při zobrazování vyrábí normální televizní snímek. Výhodou toho je, že je možno jej sledovat na televizním přijímači, který má vstup RGB, popřípadě pomocí jednoduchého a levného modulátoru i na obyčejné televizi, či jej nahrát na video-magnetofon. Současně je také možno použít monitor (např. 1084) místo televizoru k videu. Nevýhodou je stejný obrazový kmitočet jako v televizním vysílání, při kterém po delší práci u obrazovky dochází k vyšší zrakové únavě. U počítačů typu PC-AT je obrazová i řádková frekvence vyšší, čímž je obraz klidnější, neblíká a není znatelné řádkování. Zařízení, které udělá na počítači PC televizní snímek, je však nesrovnatelně dražší a většinou převyšuje i cenu samotného PC. Z toho vyplývá, že Amiga je velmi vhodná pro ty, kteří se zabývají televizní tvorbou profesionálně, a také pro filmové amatéry, kteří přešli z klasického filmového materiálu na video a chtějí si své rodinné záběry vylepšit titulky či nějakou animací.

K čemu tedy Genlock je. Pomocí něj donutíte Amigu, aby se zasynchronizovala na přicházející televizní signál a do něj vkládala svůj obrázek. Zařízení A8802 je krabička, která se přírodním kabelem zasune do 23ti pólového konektoru Amigy místo monitoru. Následně monitor připojíte do zadní strany genlocku. Dále genlock obsahuje dva konektory typu BCN, označené Video IN a Video OUT. Do Video IN zapojíte zdroj signálu. Tím může být videokamera, videomagnetofon na kterém je vyladěno nějaké televizní vysílání nebo zapnuto přehrávání nahrané videokazety. Také je možno použít Laser Disk. Video OUT spojíte s videem, na které si své dílko nahrajete, a u něj je dobré mít připojenou televizi, přes kterou sledujete výsledný obrázek.

Genlock A8802 nemá žádné ovládací prvky, takže práce s ním je velmi jednoduchá. Po spojení všech zařízení, jak je uvedeno výše, začneme zapínat do sítě všechny přístroje tak, aby Amiga byla zapnutá jako poslední. Pokud zařízení připojené do Video IN nedodává žádný signál, je obrázek na monitoru Amigy nestabilní. Proto je dobré navolit pro zkoušku nějaký televizní vysílač se signálem PAL (např. OK3). Obraz se okamžitě srovná a je naprosto normální - bílý podklad a ruka s disketou. Na televizoru ale

uvidíte obrázek televizního vysílání a navíc také ruku s disketou. Vložíte-li nyní do Amigy např. disketu s Workbench, zjistíte, že vše, co vidíte na monitoru Amigy, je i na televizním přijímači, s tím rozdílem, že základní podkladová barva Workbench (modrá) je průhledná a v ní je vidět procházející televizní obrázek. V tomto případě pracuje genlock v Background módu, to znamená že Bitplane 0 počítače je průhledný.

Nyní můžete zkusit genlock s nějakým grafickým programem. Tak například DeluxePaint vám umožní malovat do televizního snímku vlastní obrázky nebo text a ten pomocí animace měnit a provádět s ním různá kouzla. Nastavením Overscan módu můžete zakrýt celou obrazovku a pak průhlednou barvou otevřít okno, v němž bude vidět televizní vysílání. Průhledná barva je ta první vlevo nahoře a nezáleží na její skutečné barevnosti. Je-li na monitoru bílá, černá nebo třeba zelená, je vždy v televizi průhledná. Všechny ostatní barvy jsou neprůhledné a tedy jsou vidět na televizi stejně jako na monitoru. Samozřejmě jsou vidět i myš a ovládací lišty, pokud je nevypnete (viz ovládání programu DeluxePaint).

V programu DigiPaint je průhledná barva první zleva. Navíc je možno v menu Prefš položkou Genlock průhlednost vypnout. Další program PhotonPaint má průhlednou kreslicí podložku, ale žádnou barvu, takže má jisté omezené možnosti. Samozřejmě též můžete vytvořit grafiku jiným způsobem, třeba i v Basicu.

Tímto základním módem genlocku je možné obsáhnout většinu požadavků na práci s televizním obrázkem. Pro náročnější aplikace lze ke genlocku A8802 připojit přepínač módů. Podle návodu a s přiloženým konektorem si uděláte přepínač dalších módů: Background, Foreground-1, Foreground-2, Pouze TV, Pouze Amiga. Krátký program z přiložené diskety umožňuje softwareové přepínání přes paralelní port.

V této chvíli již nebrání nic tomu, abychom mohli začít s videotvorbou. Bude-li to vylepšení rodinného záznamu z dovolené nebo profesionální titulkování zahraničních filmů, či tvorba reklamy pro televizi, to už záleží na vás.

Výše popsaný genlock je dodáván v zásilkovém prodeji naší redakce, za cenu 16 999,- Kčs.

### INZERCE

**Firma Antik Soft nabízí**

pro Amigu 500, 2000 a 3000

**MAX Editor**

Textový editor s slovníkem pre AMIGU.

Anglicky, Německy, Slovensky a Česky

Informace a objednávky na adrese:

**ANTIK Soft**

**S.N.P. 16**

**040 11 Košice**

**tel: 095/34356.**



# HRV

KARFI, GRIFIN, DOCTOR

## MIG-29 FULCRUM

Domark Soft. 1991

TEST 4/92

|              |     |                      |
|--------------|-----|----------------------|
| Grafika      | 70% | Celkem<br><b>58%</b> |
| Zvuk         | 64% |                      |
| Idea         | 55% |                      |
| Přitažlivost | 45% |                      |

Tvrdíte o sobě, že jste milovníkem leteckých simulátorů, a zároveň vás již omrzely lety v superelektronicky vybavených F-15, F-16, F-18, F-19 ...? Pak je simulátor firmy Domark právě to, co hledáte, a proto neváhejte a usedněte do cockpitu nejlepšího sovětského stíhacího letounu - MIGu 29 Fulcrum.

Nejdříve ale několik slov k vybavení stroje.

Počítačová identifikace nepřátelských letadel, kvalitní radar, mapa, autopilot, ILS (Instrumental Landing System), kamery pro zaměření cílů a další elektronické "zázraky", nepatří zrovna mezi ta zařízení, na která by byl Mig 29 štědrý (ono se toho s elektronikami ani moc dělat nedá).

A je také pravda, že palubní deska vzhledem svých "budíků" na první pohled spíše odrazuje. Ale již na ten druhý každý zkušený pilot pozná, že je na ní vše, co k letu nezbytně potřebuje, nic méně (a ani nic víc). Tak se na ni tedy pozorně koukněte.

**Popis palubní desky:**

A - flares

- B - chafs
- C - kulometné dávky
- D - neřízené rakety
- E - rakety vzduch-země
- F - rakety vzduch-vzduch
- G - Systémy letadla (podvozek, klapky, brzdy ...)
- H - umělý horizont
- I - směr a vzdálenost k cíli
- J - tah motoru
- K - waypoint
- L - poloha knyplu
- M - výškoměr
- N - vertikální rychlost
- O - radar
- P - tah levého a pravého motoru
- Q - palivo
- R - indikace poškození letounu
- S - indikace automatického vyrovnávání

Nyní, když se již vyznáte v palubní desce, vám nic nebrání v návštěvě Briefingu Roomu, pardon, místnosti předletové přípravy.

Zde na vás již čeká operační důstojník Aljoša, aby vám přidělil úkol.

1. Training (Tréning)
2. Blue (Modrý) - útok na v ledu uvízlou nepřátelskou ponorku (že by U.S. Navy nefasovala sekýrky?)
3. Yellow (Žlutý) - menší konflikt na čínské hranici
4. White (Bílý) - ach ten Perský záliv
5. Red (Červený) - nepřítel se přesunuje. Váš úkol - ZASTAVIT!
6. Final (Finále) - nálet na nukleární zařízení
7. Pilots (Piloti)

Úkoly si zvolíte klávesami 1 až 7 nebo kurzorovými klávesami a RETURNem. Všechny mise jsou přísně tajné, takže od Aljoši dostanete jen fotografii cíle a plavte (vlastně leťte).

### OVLÁDÁNÍ:

Letoun můžete pilotovat jak joystickem, tak i myší, nebo klávesnicí.

Hlavní klávesnice (německá):

CTRL +

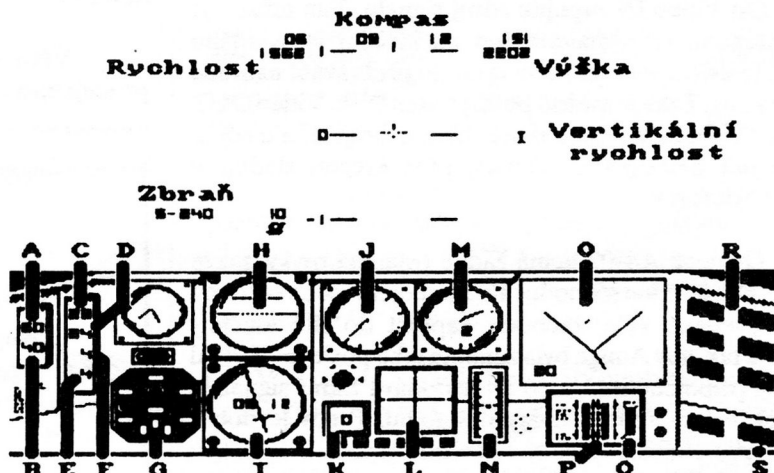
- D - opuštění mise
- E - katapultáž
- J - ovládání joystickem
- K - ovládání klávesnice
- M - ovládání myši

SHIFT +

- [dvojitě s] - úplné ubrání plynu
- plyn na maximum

Esc - vymazání programu

A - automatické vyrovnání letounu



- B - vysunutí/zasunutí brzdících klapek
- C - vypuštění staniolových proužků na odlákání radarově naváděné rakety
- E - vypnutí/zapnutí motoru
- F - vypuštění světlice (odvádí tepleně naváděné rakety)
- L - podvozek (velmi používán zejména při přistání)
- N - vypnutí/zapnutí zvuků motorů
- O - pohled na letadlo zvenku
- P - pauza
- Q - vypnutí/zapnutí veškerých zvuků
- V - pohled na letadlo zvenku
- W - brzdy podvozku
- - změna rozsahu radaru (30, 15 a 7 km)

[Přehlasované O] - změna Waypointu

#### DelBkSp

- (šipka doprava) - přepínání raket
- AA-8 - řízené vzduch-vzduch
- AS-7 - řízené vzduch-země
- S-240 - neřízené rakety

RETURN - zaměření cíle

SPACE  
HELP → vystřelení rakety

DEL - palba z kanónu

#### Kursorové

- klávesy - ovládání letu (jen, je-li zapnuto ovládací klávesnice
- ; - kormidlo doprava
- : - kormidlo doleva
- [dvojitě s] - ubrání plynu
- ' - přidání plynu

#### Přídavná klávesnice:

- 1 - pohled diagonálně vlevo dozadu
- 2 - pohled dozadu
- 3 - pohled diagonálně vpravo dozadu
- 4 - pohled doleva
- 5 - pohled dopředu bez palubní desky
- 6 - pohled doprava
- 7 - pohled diagonálně vlevo dopředu
- 8 - pohled dopředu
- 9 - pohled diagonálně vpravo dopředu

#### LET:

Létání s Migem je jednoduché a zvládne ho každý začínající GAMESNÍK [gamesník] (člověk věnující se počítačovým hrám). Během letu je jen důležité neustále si hlídat výšku.

K cíli se dostanete nejlépe nasměrováním letounu tak, aby červená ryska na kompasu byla uprostřed (viz obr.).

Navádění na jednotlivé cíle si volíte změnou Waypointu.

Vyhodnocení letu (po přistání), kdy se u běžných a tudíž nudných F-xx dostává náležitých poct, vyřešili u Domarků jak jinak, než originálně. Protože ať misi splníte či nikoliv, je to fuk - kromě bodů totiž nedostanete NIC. (Já vám doporučuji dát si jako odměnu malého panáka Vodky - to pro lepší mušku.)

#### ÚTOK:

Jsou dva typy útoků:

##### 1. útok na pozemní cíle

- a) řízenými raketami - velice jednoduché.

Provede se následovně:

- I. zvolte AS-7 (DelBkSpace)
- II. zaměřte cíl (RETURN)
- III. vypalte raketu (SPACE)

- b) neřízenými raketami - složitě
- I. zvolte S-240 (DelBkSpace)
- II. naletěte na cíl
- III. vypalte raketu (SPACE)

- c) kanónem - je to vůbec možné?
- I. naletět na cíl
- II. palte o sto šest (FIRE)

##### 2. Útok na letící cíle

- a) řízenými raketami - je jednoduché, ale většinou neúčinné
- I. zvolte AA-8 (DelBkSpace)
- II. zaměřte cíl (RETURN)
- III. vypalte raketu (SPACE)

- b) neřízenými raketami - obtížné, ale možné
- I. zvolte S-240 (DelBkSpace)
- II. naletěte na cíl
- III. vypalte raketu (SPACE)

- c) kanónem - vše, co není vyloučené, je možné
- I. naletět na cíl
- II. palte o sto šest (FIRE)

#### UKLÁDÁNÍ STAVU HRY:

Opět velice jednoduché. Stačí pouze odblokovat disketu, zapsat se do Pilots, a pak už jen létat, létat, létat a létat (a hlavně nespádnout).

Co říci závěrem? Pokud šlo programátorům o přiblížení sovětské techniky průměrnému západnímu člověku, tak se jim to povedlo. Avšak co se týká rychlosti simulace, kvality grafiky a zvuků, je Mig - 29 asi tak na úrovni Fighter Bomberu.

Jednu věc bych však programu speciálně vytkl - mám na mysli tu "rádoby" azbuku. Nás, co jsme strávili nejednu noc nad "bukvičkami", totiž dohání k šílenství (nedá se to vůbec přechíst).



#### OMLUVA

Zklamání příznivci, chtěli bychom se vám omluvit za nedodané pokračování vašeho oblíbeného programu Silence Service II, neboť napsané pokračování podlehlo zákeřnému viru a nebylo v našich silách je v krátkém termínu opět napsat.

Jako malou náplast tedy uveřejňujeme návod na program F-29. Příjemné chvíle u F-29 vám přeji Karfi, Griffin a Doctor.

# ALIEN BREED

.....1991

| TEST 4/92    |      |                       |
|--------------|------|-----------------------|
| Grafika      | 78 % | Celkem<br><b>73 %</b> |
| Zvuk         | 81 % |                       |
| Idea         | 69 % |                       |
| Přitažlivost | 65 % |                       |

Neznám člověka, který by neviděl nebo alespoň neslyšel o filmu Alien a jeho druhém pokračování Aliens - osobně (a určitě nejsem sám) řadím oba snímky mezi nejlepší sci-fi thrillery (drasťáky), jaké kdy byly natočeny.

Lze těžko říci, co je učinilo tak populárními. Jisté ale je, že jejich námětu (dramatického boje lidí proti mimozemské nestvůře) se stále v různých obměnách využívá nejen ve filmu či v literatuře, ale i v počítačových hrách (připomeňme si například Infestation).

Alien Breed, jak již název napovídá, také čerpá z módního kultu Vetřelců. Tentokrát v té nejjednodušší podobě - střílečky.

Děj hry se odehrává kdesi na samém konci vesmíru, kde okolo jedné planety obíhá kosmická základna. A právě v tomto vesmírném komplexu se jaksi "přemnožili" velice přítulní "potvoráci" (každého by nejraději samou láskou snědli). Nutno podotknout, že se jim to vcelku i podařilo.

Nic ale netrvá věčně a čas, kdy byly potvory pány základny, skončil - do akce právě nastupuje speciální dvoučlenný likvidační oddíl.

## To pravé drama začíná.

Celá hra je brána z ptačího pohledu, což působí velmi přehledně. V horní části obrazovky je čarou vyznačena vaše energie, počet životů, střeliva a klíčů (těch můžete mít maximálně šest).

Jedinou zbraní, kterou máte od začátku u sebe je sice jen samopal (vzor 92 'Stír'), ale v průběhu hry si

můžete dokoupit vše od střeliva až po extra život.

Pohyb po základně je normální - pomocí joysticku. Střelíte FIRE tlačítkem. A abych nezapomněl, základna se skládá z několika podlaží, přičemž platí pravidlo, že každé patro dolů je jeden "level" nahoru.

Pokud si chcete cokoli dokoupit, zjistit svou pozici v labyrintu chodeb, nebo se třeba jen podívat, kolik potvor jste už eliminovali, stačí jen přistoupit k počítači a ...

Na obrazovce se vám objeví INTEX SYSTEM MENU (monitory jsou poškozené a proto musíte mít chvíli strpení).

- INTEX WEAPON SUPPLIES : Umožní koupit nejrůznějších typů zbraní (od samopalu až po pulsující laser)

- INTEX TOOL SUPPLIES : Došlo-li vám střelivo, chybějí klíče nebo jste zraněni - zvolili jste dobrou volbu

- INTEX LEVEL SERVICE : Kurzorem označí vaši pozici v bludišti

- LEVEL INFORMATION UPDATE : Předá vám informace o daném levelu (podlaží) spolu s úkoly, které je nutné pro zdárné ukončení celé mise v tomto podlaží vyřídit.

- STATISTICS : Nudná statistika, podávající informace o tom, kolik "potvoráků" jste už poslali na onen svět, kolik jste na to potřebovali kulek, jakým druhem zbraně nyní operujete atd.

- EXIT INTEX NETWORK : Výstup ze systému, pokud vás již přestalo bavit Alien breed, můžete si zahrát televizní hru - tenis.

Jistě uznáte sami, cože by to bylo za hru (a střílečku zvlášť), kdyby neumožňovala hru dvou hráčů zároveň. V tomto směru vyšli programátoři všem "gamesníkům" a jejich kamarádům vstříc. Stačí si jen v úvodním menu navolit volbu '2 PLAYERS'.

A závěrečný verdikt? Spíše nadprůměrná střílečka, kterých bylo je a doufejme, že bude k nemalé radosti všech hráčů vždy dost.

# ELVIRA - ARC. GAME

Flair Soft. Ltd 1991

| TEST 4/92    |      |                       |
|--------------|------|-----------------------|
| Grafika      | 65 % | Celkem<br><b>60 %</b> |
| Zvuk         | 58 % |                       |
| Idea         | 57 % |                       |
| Přitažlivost | 62 % |                       |

Ne, nejedná se o pokračování úspěšného a podle mého názoru snad nejpropracovanějšího adventure na naší přítelkyni, ale, jak napovídá druhá část názvu, o klasickou chodičko-skákačko-střílečku (tedy spíše vrhačku, protože se v ní nestřílí, ale vrhají nože).

Ptáte se, co že má hra společného s Elvirou, s tou krásnou majitelkou strašidelného hradu? Nic míň než to, že místo nějakého hrdiny, který si své svalstvo vypěstoval v neustálých soubojích s netvory všeho druhu, proháníte po obrazovce sličnou brunetu s velkým výstřihem a rozparkem na sukni (nutno ale podotknout, že ani tyto aspekty neodradí tajemné mnichy či nebohé starce od toho, aby po ní nehodili ohnivou kouli).

Před začátkem samotné hry si musíte zvolit obtížnost (1,2,3) - liší se počtem životů, odolností a počtem potvor.

Pak následuje ještě výběr jazyka, kterým chcete, aby s vámi vaše Amiga komunikovala (čestina opět nikde).

Po těchto volbách vám zbývá ještě poslední, a to je vybrat si, zda chcete putovat v ohňovém nebo ledovém světě.

A vzhůru do světa, odkud již většinou není návratu ...

Hra se jako arcada nevyznačuje ničím převratným (grafika tak na úrovni ST, zvukové efekty jsou průměrně ucházející) a lehce by to zvládl kdejaký osmibit. Jediné, čím tato hra snad upoutá, je úvodní demo, i když musím říci, že jsem už viděl lepší.

Na Elviru II - Adventure se tedy těším mnohem více.



## HUGO II

.....1991

## TEST 4/92

|              |      |                       |
|--------------|------|-----------------------|
| Grafika      | 85 % | Celkem<br><b>81 %</b> |
| Zvuk         | 87 % |                       |
| Idea         | 65 % |                       |
| Přitažlivost | 90 % |                       |

Asi tak v polovině loňského roku se objevil na našich monitorech a TV obrazovkách Hugo - malý rozpustilý čertík (takový ošklivější čertík Bertík), který ve stejnojmenné hře bloudil v jakýchsi důlních šachtách. Přestože si tato hra (určená té nejmenší věkové kategorii amigahráčů) vysloužila slova chvály (od těch menších) i slova hany (to co na její adresu pronesl jeden "lamač joysticků", je neslušné), neusnuli její autoři na vavřínech a pracovali dál na novém příběhu rozpustilého čertíka. Dnes se podíváme, jak úspěšně si vedli tentokrát.

Nejdříve pár slov k příběhu: Zlá čarodějnice unesla Hugovy nejmilejší - manželku (až ji uvidíte, tak se nelekněte) a dvě rozpustilá dítk - na vysokou horu ležící za devatero železničními tratěmi a devatero hustými lesy. Je vám samozřejmě jasné, že to Hugo tak nenechá a vyřídí si to s ní.

Nejprve se před vámi objeví menu, kde si můžete vybrat ovládací zařízení (joystick nebo klávesnice), určit obtížnost hry (TV version je pro malé děti a Arcade dá zabrat i těm nejhráčům).

Dále si můžete zvolit typ grafiky (lituji, ale neumím dánsky, takže vám nepovím, v čem se liší - sám jsem žádný rozdíl neviděl). No a nakonec stačí stisknout klávesu F10 a tím začít dobrodružství.

Samotnou hru můžeme rozdělit do tří částí:

První (řekl bych nejlehčí) část hry je její začátek - cesta Huga přes devatero železnic. Na první pohled je to jednoduché, vždyť malá drezina může dovést Huga kamkoliv, má to však hned několik háčeků.

První je ten, že koleje jsou tři. To bysice nebyl pro Huga zas tak velký problém (drezině stačí v pohodě jedna), kdyby ale po nich nejezdily lokomotivy. Tím pádem mu nezbývá nic jiného, než na místech k tomu určených (výhybky) přejíždět z jedné do druhé či do třetí tak, aby neporazil žádnou z lokomotiv. Naštěstí má s sebou mapku, kde jsou vyznačeny jak koleje, tak i lokomotivy.

V druhé části to už má trochu těžší. Po úzké pěšince se musí dostat až na samý vrchol hory, kde zlá čarodějnice drží v zajetí jeho rodinku.

Při rychlém poklusu nahoru až na horu musí však ještě stihnout přeskakovat nejrůznější prolákliny a uhýbat kutálejícím se kamenům, které z dlouhé chvíle posílá dolů zlá čarodějka. To ale ještě není vše. V pravý okamžik musí navíc vystřelit speciální přísavné lano a vyšplhat se po něm vzhůru.

Jestli se Hugovi podaří "vydrápat" se až na samý vrchol, tak se ještě stejně nemá proč radovat. To nejtěžší ho totiž teprve čeká.

Po ne zrovna krátkém nahrávání se ocitne v místnosti, ve které jsou věznění i jeho nejbližší a nelmilejší. Před sebou vidí čtyři provazy a jen na něm nyní záleží, za který zatahne. Dobře si to ale musí rozmyslet, protože jenom jeden je ten pravý a ty ostatní

nespoutají čarodějnici, ale jeho (a můžete začít hrát znovu).

Jestliže se vám v roli Huga podaří zatahnout za ten správný provaz, bude čarodějnice spoutána a vyhozena ze svého hradu a vy se svou rozkošnou milou ženuškou prožijete nejednu hezkou chvíli.

A zazvonil zvonec a pohádky je konec.

Podíváme-li se na Huga II důkladně, zjistíme, že mu nelze prakticky nic vytknout. Nejenže udržuje pomocí vtipných (pravda dánských) dialogů neustále kontakt s hráčem, ale ani jeho grafická, zvuková a animační stránka nijak nepokulhávají za celkovým provedením a (i když je to smutné) snad nejlepší na celé této vynikající hře jsou nešťastné konce Huga. A proto, máte-li menšího bratříčka či sestřičku, neváhejte a sežeňte jim veselého čertíka.

Jednu vadu jsem ale přece jenom zaznamenal - hra je na pěti disketách (ale co to je pro ty, co mají harddisk).

Na závěr chci říct, že pevně doufám, že jste se při této hře dobře pobavili, užili si plno legrace a ocenili to, že máte doma právě svoji Amigu a ne třeba nějaký ten Sharp, Atari, Amstrad nebo třeba taky to komerční PC.

jedné disketě (oproti dvoudisketovým Lemmings).

A když už jsme u těch Lemmings, tak nedávno se objevila skutečná 'bomba' s názvem

## OPERATION LEMMINGS.

Její výrobce si nepřeje být jmenován a není se mu ani proč divit. Nejenže je program sestaven z částí jiných programů (česky řečeno to znamená, že autor vykradl vše, co mohl), ale ani idea hry není zrovna chvalyhodná. Vítězí ten, kdo svým odstřelovačem zlikviduje nejvíce zelených trpaslíků.

/psáno v lednu 1992/

## NOVINKY A ZAJÍMAVOSTI

Jedna z horkých novinek, která se již u nás objevila, jsou Lemmings - no more (Čech by to asi nazval Lemmings II, ale to víte Američani, ti musí mít vždy něco extra).

Jedná se o prakticky stejnou hru jako jsou Lemmings. Rozdíl je jen ten, že jsou různé levely (podle mého názoru propracovanější).

Snad největším plusem této hry (Lemmings II) je to, že jsou jenom na

# THE KILLING CLOUD

■ Vek.G.,Im.Works 91

| TEST 4/92    |      |                       |
|--------------|------|-----------------------|
| Grafika      | 92 % | Celkem<br><b>80 %</b> |
| Zvuk         | 68 % |                       |
| Idea         | 78 % |                       |
| Průtažlivost | 82 % |                       |

(2. část)

Vítejte všichni, kteří hodláte zatočit s démony zločinu v San Francisku. Dnes si povíme o tom, jak se vše ovládá.

## Start

Po nahrání se vás program zeptá, jakým jazykem si přejete, aby komunikoval (ne, neradujte se, čeština tam opět není). K dispozici máte angličtinu, němčinu a francouzštinu.

V dalším textu jsem vycházel z anglické mutace, protože se domnívám, že angličtinářů je mezi amigisty nejvíce. Ostatní ať mi laskavě promínou.

Po volbě jazyka se program ptá na kód mise. Pokud nezačáte nic, začínáte od začátku (první misí).

## Hlavní menu:

V hlavním menu (velká zasedací místnost) máte celkem pět možností. Jejich výběr se provádí pomocí myšky a LM (levé tlačítko, PM = pravé tlačítko na myši).

- \* ASSIGMENT
- \* BACK UP
- \* ARMOURY
- \* CRIME FILE
- \* CITY INFO

## ASSIGMENT:

Zde se dozvíte, jaký je (v dané misi) váš úkol. Návrat do hlavního menu provedete stlačením PM.

## BACK UP:

Pro úspěšné splnění většiny úkolů je třeba, abyste si do ulic San

Franciska umístili některé potřebné věci (P.U.P.y a řízené síťové střely zvané síťovky - jejichž popis je uveden dále).

POZOR! Počet P.U.P.ů a síťovek je omezen (10 a 10). Rozmíst'ujte je tedy opatrně a vždy jen v nejnutnějším počtu.

Umístění se provádí následovně:

1. Nejdříve si zvolte městskou čtvrť
2. Z menu: \* NETS (střely)  
\* PUPS (P.U.P.)  
\* CANCEL (zrušení)

vyberte požadovaný předmět a umístěte jej na mapu.

Vymazání se provádí obdobně:

1. Nejdříve si zvolte městskou čtvrť
2. Z menu: \* NETS  
\* PUPS  
\* CANCEL

zvolte CANCEL a ukazováním na jednotlivé předměty je "uklidíte" zpět.

Návrat do hlavního menu je opět pomocí PM.

**ARMOURY:** (POZOR!, po této volbě již není návratu do hl. menu)

Správné vyzbrojení hoverbiku je dalším důležitým krokem pro úspěšné splnění mise. Váš hoverbike může nést až 350 liber zátěže. Je tedy na vás, jak toho využijete. K dispozici máte kanóny, kulomety, ochranný oblek a přídatnou palivovou nádrž.

Výběr se provádí ve vyzbrojovacím automatu pomocí myši.

Zvolíte-li LAUNCH, ukončíte vyzbrojování a čeká vás vlastní start hoverbiku (viz dále).

## CRIME INFO:

Zde máte možnost seznámit se pomocí policejní kartotéky s těmi bídáky, kteří na vás v dané misi čekají.

Kartotéku lze ovládat jak cursorovými klávesami tak myši. PM opět zajišťuje návrat do hlavního menu.

Pozn.:

Zprávy z kartotéky neberte moc vážně. Třeba takový Eugen Cody je sice v databance označován jako

velice nebezpečný, ale při zatýkání se nezmůže ani na střelbu z kulometu.

## CITY INFO:

Další policejní databanka, tentokrát obsahuje vyobrazení budov (které lze použít jako orientační body), techniky SFPD a Black Angels.

Databanku lze opět ovládat cursorovými klávesami a myši.

## Technické vybavení SFPD (San Francisco Police Department)

Toto je seznam těch prostředků, jimiž disponuje SFPD a které máte (ev. budete) mít k dispozici i vy.

### XB500 Hoverbike

Automobily ve službách policejních hlídek již dávno zastaraly a na jejich místo nastoupily hoverbiky - víceúčelová vznášedla.

Typ XB500 je jednomístným postrachem veškerého podsvětí. Svou úctu si získal nejenom díky své rychlosti, manévrovací schopnostmi, pancéřováním či výzbrojí (kanóny a kulomety totiž mohou tvořit veškerou užitečnou zátěž stroje), ale hlavně možností nést tři řízené síťovky (Net Missile Pod), umožňující odchyt zločince přímo ze vzduchu i s jeho vznášedlem.

XB500 se může pohybovat jak směrem vpřed, tak i vzad a jeho manévrovací schopnosti jsou umocněny použitím dvou speciálních módů provozu.

V prvé řadě se jedná se o letový (Flight) mód, kdy se stroj chová jako letadlo (je-li tah motoru >0 či <0) nebo jako helikoptéra (tah = 0).

Druhý mód je tzv. vznášecí (Hover), kdy lze při nulovém tahu motoru volně stoupat či klesat v závislosti na orientaci podélné osy stroje. Je-li tah motoru nenulový, chová se XB500 stejně jako při letovém módu.

Na palubě XB500 je samozřejmě i palubní počítač, který průběžně podává zprávy o stavu zbraň, poloze, stavu paliva, ...

- A - stupeň poškození stroje
- B - ukazatel paliva
- C - zásoby střeliva (M - kulomety, C - kanón)
- D - zprávy palubního počítače
- E - palubní radar s proměnným rozsahem
- F - ukazatel tahu motoru (zpětný chod červeně)
- G - obsazení slotů na řízení síťovky
- H - výškoměr (zelená část - výška nad mrakem, hnědá - vzdálenost od země, červená - nadmořská výška povrchu)

### Ovládání XB500

(velká písmena v závorce uvádějí zprávu palubního počítače)

MYŠ - ovládání XB500

LEVÉ TLAČÍTKO NA MYŠI

- palba zvolenou zbraní

- F1 - pohled dopředu
- F2 - pohled dozadu
- F3 - pohled doleva
- F4 - pohled doprava
- F5 - pohled zezadu
- F6 - vnější pohled - možnost natáčení přídavnou klávesnicí nebo kurzorovými klávesami (/ znamená nebo)

- 8/šipka nahoru - natáčení do nadhledu
- 2/šipka dolů - natáčení do podhledu
- 4/šipka vlevo - natáčení doleva
- 6/šipka vpravo - natáčení doprava
- 9 - oddálení XB500
- 3 - přiblížení XB500

- E - zapnutí/vypnutí motorů, systémů (ENGINES ON/OFF) - pouze po přistání na pevné zemi či policejní stanici.
- R - zmenšení rozsahu palubního radaru
- T - zvětšení rozsahu palubního radaru
- S - zapnutí/vypnutí policejní sirény

- slouží k upoutání pozornosti
- D - odhození bomby (BOMB DROPPED)
- F - letový mód (FLIGHT MÓD ENGAGED)
- H - vznášecí mód (HOVER MÓD ENGAGED)
- L - nastoupení/vystoupení z XB500 - pouze při vypnutých motorech a tudíž po přistání
- C - aktivace kanónu (CANON ARMED)
- V - zpětný chod stroje
- B - chod stroje vpřed
- N - aktivace síťovek (NET MISSILE ARMED)
- M - aktivace kulometů (MACHINE GUNS ARMED)
- SPACE - brzda

Pozn.:

Létání s XB500 se zdá zpočátku strašné (tedy mně se alespoň takové zdálo), ale až poletíte po sedmé 10. misi, tak vám to ani nepřijde a dokážete s ním přistát i na pětiku.

Občas se vám stane, že se zaseknou kanóny (ev. kulomety) a nemůžete s nimi střílet. Nejedná se o nic vážného. Musíte jen chvíli počkat, než se vše automaticky opraví. To, že nepřítel nepočká, vám ale nemusím připomínat, že?

### P.U.P. (Pickup Pod)

Zastaraly nejen automobily, ale

i staré dobré antony, určené pro přepravu zločinců. Místo nich se používají tzv. Pickup Pody - sbírající kontejnery.

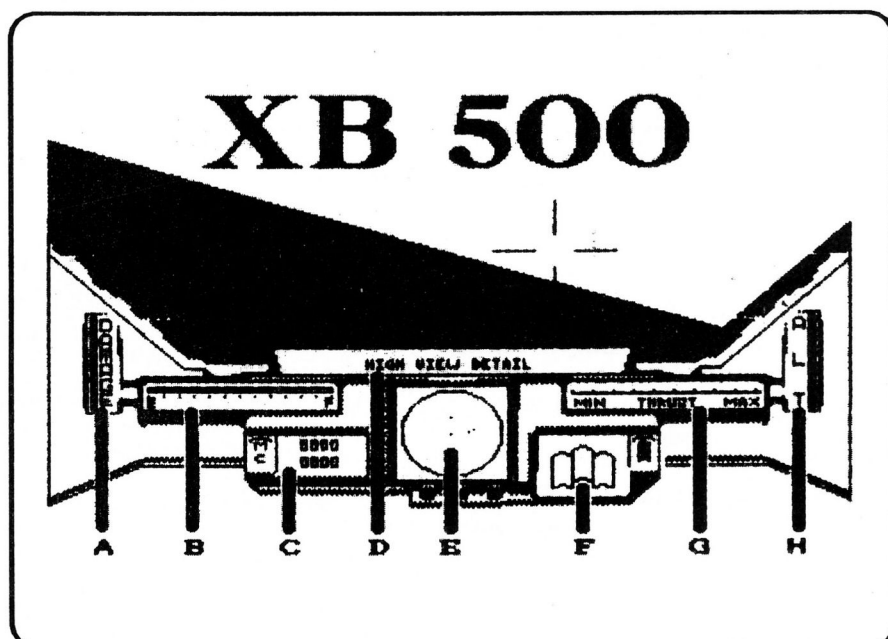
Použití:

1. Před začátkem mise si umístěte P.U.P. někde do ulic San Francisca.
2. Jakmile zatknete zločince, P.U.P. se k němu automaticky dopraví a "odveze" ho do chladku.

Pozn.:

Celý provoz P.U.P.u je plně automatický a nevyžaduje od vás žádnou aktivitu. Bližší informace naleznete v příštím čísle v části nazvané První mise.

No a jsme na konci našeho dnešního povídání. Doufám, že jste si zkusili létání s Hoverbikem a přišli jste mu, alespoň trochu, na chuť, a že se již těšíte na naše příští setkání, kde si povíme, jak pracovat se síťovkami, které z vás teprve učiní postrach podsvětí San Francisca a kde si společně splníme první z deseti misí. Zatím ale pilně cvičte létání a střelbu. Bude se vám to hodit.







Redakce našeho časopisu nabízí  
výrobky  
**firmy COMMODORE**

a z bohaté hardwarové nabídky vám v současné době můžeme  
dodat tento sortiment:

|                                           |              |
|-------------------------------------------|--------------|
| <b>Amiga 500 PLUS</b>                     | 18990,- Kčs  |
| CPU 68000, 7.14 MHz, 1 Mb RAM             |              |
| <b>Amiga 500</b>                          | 16249,- Kčs  |
| CPU 68000, 7.14 MHz, 512 Kb RAM           |              |
| <b>Color monitor 1084 S stereo</b>        | 11599,- Kčs  |
| 14", vstup RGB analog, RGBI               |              |
| <b>Rozšíření paměti A501, 512 KB</b>      | 2079,- Kčs   |
| <b>HF modulátor</b>                       | 1049,- Kčs   |
| <br>                                      |              |
| <b>Hard disk A590 - 20 MB</b>             | 14999,- Kčs  |
| rozšíření paměti až o 2 Mb, řadič SCSI    |              |
| <br>                                      |              |
| <b>Disketová jednotka 3,5" A1011</b>      | 4599,- Kčs   |
| externí disketová jednotka fy COMMODORE   |              |
| <br>                                      |              |
| <b>Amiga 2000</b>                         | 28499,- Kčs  |
| 8 MHz, 1 Mb RAM                           |              |
| <br>                                      |              |
| <b>Disketová jednotka pro A 2000/3000</b> | 4199,- Kčs   |
| interní disketová jednotka                |              |
| <b>A 2058/2 Paměť A2/A3 2MB-8MB</b>       | 11899,- Kčs  |
| <b>A 2091/2092 20 HD S kontrolerem</b>    | 13299,- Kčs  |
| <b>A 2091/2094 50 HD S kontrolerem</b>    | 20799,- Kčs  |
| <b>A 2620 68020/2MB/68881</b>             | 23499,- Kčs  |
| <b>A 2630 68030/2MB/68882</b>             | 28699,- Kčs  |
| <b>A 2630 68030/4MB/68882</b>             | 35899,- Kčs  |
| <b>A 2088 XT, 512 KB</b>                  | 9499,- Kčs   |
| <b>A 2286 AT, 1MB</b>                     | 23099,- Kčs  |
| <b>A 2300 GENLOCK</b>                     | 6149,- Kčs   |
| <b>A 2320 FLICKERFIXER</b>                | 10799,- Kčs  |
| <br>                                      |              |
| <b>Joystick SJ 119 Junior</b>             | 169,- Kčs    |
| <b>SJ 122 Auto fire</b>                   | 229,- Kčs    |
| <b>SJ 124 Auto fire-microsp.</b>          | 369,- Kčs    |
| <b>SJ 126 Profi</b>                       | 489,- Kčs    |
| <b>RMT122 Infra</b>                       | 919,- Kčs    |
| <br>                                      |              |
| <b>Amiga 3000 - 25 MHz, 2 MB, 100 HD</b>  | 109999,- Kčs |
| CPU 68030, 512 ROM, copr. 68882 25 MHz    |              |
| <br>                                      |              |
| <b>Amiga 3000 - 16 MHz, 2 MB, 50 HD</b>   | 101999,- Kčs |
| CPU 68030, 512 ROM, copr. 68881 16 MHz    |              |

|                                           |             |
|-------------------------------------------|-------------|
| <b>Color monitor C 1950</b>               | 20499,- Kčs |
| Multiscan, analog RGB                     |             |
| <br>                                      |             |
| <b>Tiskárna Star LC-20</b>                | 8399,- Kčs  |
| <b>Star LC-24-200</b>                     | 16990,- Kčs |
| <br>                                      |             |
| <b>Podložka pod myš</b>                   | 97,- Kčs    |
| <b>Diskety Commodore 3,5" 2DD á 10 ks</b> | 289,- Kčs   |
| <b>Diskbox 3,5" - 80 ks</b>               | 284,- Kčs   |
| <br>                                      |             |
| <b>C 64-II</b>                            | 5349,- Kčs  |
| <b>VC 1530</b>                            | 979,- Kčs   |
| <b>VC 1541-II</b>                         | 5699,- Kčs  |
| <br>                                      |             |
| <b>Komplet 1 - Amiga 500</b>              |             |
| <b>- Monitor 1084 S</b>                   | 27699,- Kčs |
| <br>                                      |             |
| <b>Komplet 2 - Amiga 500</b>              |             |
| <b>- A501</b>                             |             |
| <b>- HF modulátor</b>                     | 19299,- Kčs |
| <br>                                      |             |
| <b>NOVINKA:</b>                           |             |
| <b>CDTV</b>                               | 30999,- Kčs |
| <b>klávesnice</b>                         | 4249,- Kčs  |
| <b>disketová jednotka A1011</b>           | 4999,- Kčs  |
| <b>infra myš</b>                          | 2999,- Kčs  |
| <b>MY PAINT</b>                           | 1519,- Kčs  |
| <b>SNOOPY</b>                             | 2249,- Kčs  |
| <b>Literatura: AMIGA DOS v češtině</b>    | 79,- Kčs    |

Kniha je určena jak pro začátečníky, tak i pro pokročilé.

**Z obsahu vyjímáme:**

|               |                  |
|---------------|------------------|
| Amiga DOS 1.3 | Workbench v.1.3  |
| Extras 1.3    | Kickstart v. 1.3 |

**Vybrané zboží zasíláme poštou na dobírku  
do tří týdnů (Poštovné hradí zákazník)**

**Na dodávku je roční záruka, pozáruční  
servis zajištěn!**

**POPTÁVKY ZASÍLEJTE PODEPSANÉ  
NA ADRESU REDAKCE**