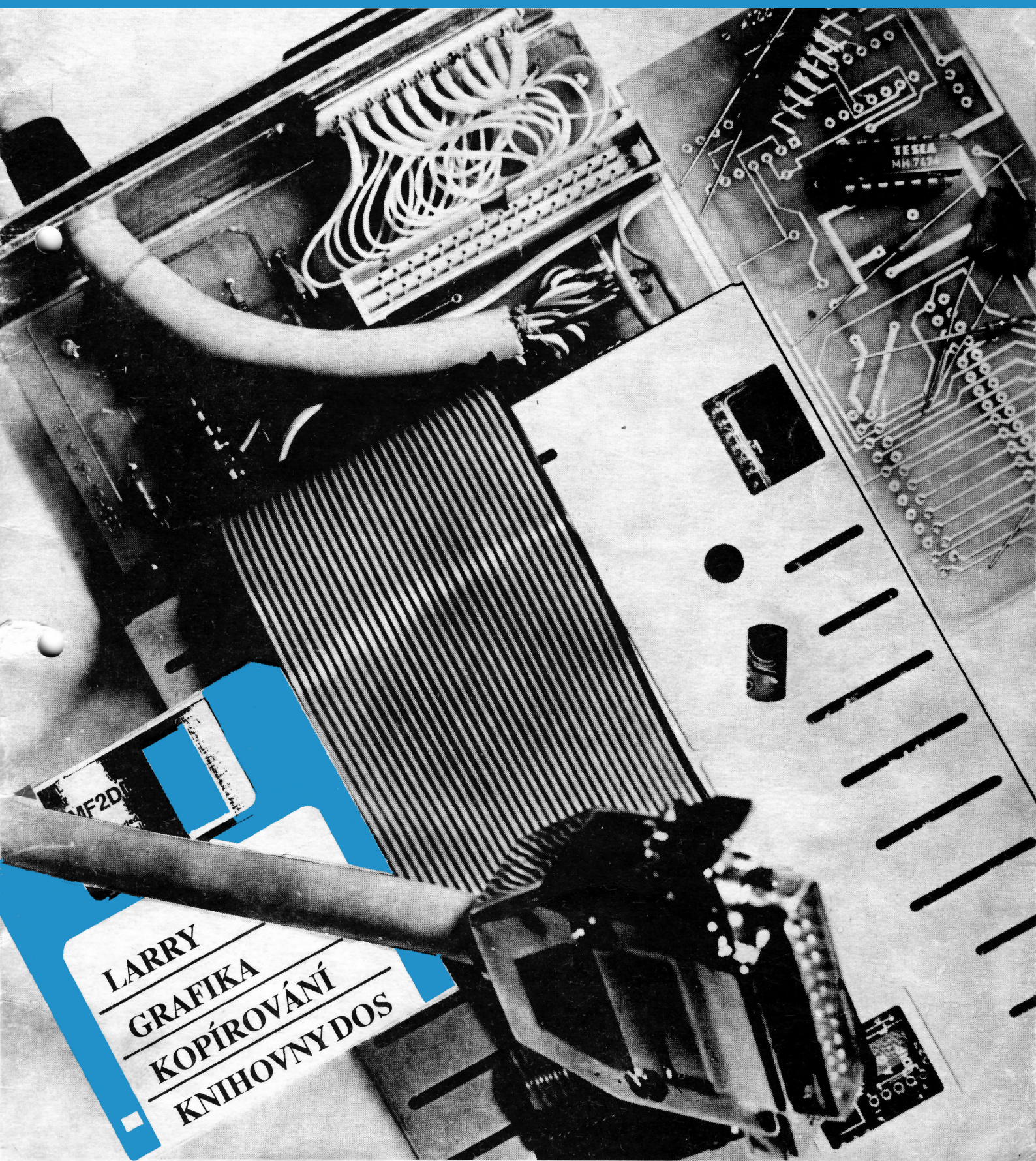


amiga

měsíčník pro uživatele počítačů amiga

20 Kčs - 2/91



ÚVODNÍK



Jak všichni určitě víte, je číslo v nadpisu významné hlavně pro uživatele a programátory, kteří se zabývají grafikou. Hodně se o tom už napsalo a hodně se také navyprávělo o neuvěřitelných možnostech počítačů Amiga. Myslím si ale, že málokterý článek napsal skutečnou pravdu o grafických módech a hlavně o těch 4096 barvách. Chtěl bych proto zde osvětlit, jak se věci skutečně mají i s rizikem, že budu muset občas na našeho miláčka nadávat. Chtěl bych také hned zdůraznit, že zde předkládám svoje subjektivní dojmy z pohledu programátora, který už se nějakou dobu zabývá programováním grafiky na počítačích řady Amiga.

Profesionální uživatelé grafických počítačových programů se hned zajímají o to, zda program má či nemá 24 bitů na pixel (tj. 2^{24} barev). Když zjistí, že má míň než těch 24 bitů tak tím opovrhnu a prohlásí, že s takovou hračkou dělat nebudou. Je situace opravdu taková, že bez takového programu nelze profesionálně pracovat? Odpověď je: "Ano i ne".

Hlavní problém totiž neleží v tom, kolik bitů máte na jeden pixel, ale jak je zobrazování grafiky uspořádáno. V počítačovém světě existují dva hlavní způsoby. Bud' přímý, kdy n bitů na pixel už přímo určuje barvu příslušného pixelu, anebo nepřímé zobrazování, kdy bity určují index do mapy barev, kde teprve zjistíte, jakou barvu má pixel. Tak a jsme u jádra celého problému. V prvním případě máme totiž "plynulou" paletu barev, kde můžeme jednoznačně říci, jak se od sebe liší pixel s hodnotou \$00 (bílá) a pixel s hodnotou \$FF (černá). V druhém případě nevíme, jaké barvy se pod těmito čísly skrývají v mapě barev a tudíž nemůžeme nic říci o barvách těchto pixelů. Dobrá, tak teď řeknete, že to nepotřebujete a že jste zcela spokojeni s druhou

variantou. Bohužel Vás poprosím, ať ještě vytrváte a dočtete úvodník dokonce, protože si myslím, že brzy pochopíte, o co mi jde.

Představte si, že máte nasnímaný obrázek (scanner, videokamera, apod.) z reálného světa a neztratily jste přitom skoro žádnou přesnost barev (dokonale to těžko zvládneme). Tento nasnímaný obrázek má nejspíše přímé zobrazování barev. Vy máte k dispozici mapu barev s určitým počtem položek, ale můžete si barvy vybírat z velké palety, např. ze 4096 barev. Nepřipomíná Vám to Amigu? Dejme tomu, že Vaše mapa barev má šestnáct položek, abychom mohli používat jemné rozlišení (640 x 512 bodů). Tak, a teď mi řekněte, jak vybrat těch šestnáct barev. Nebylo by lepší, kdybychom měli dvanáct bitů na pixel s přímým zobrazováním? Mohli bychom velmi jednoduše pomocí dithering (metoda jak rozšířit počet barev měněním odstínů sousedních bodů nebo zobrazováním obrázku s menším rozlišením na obrazovce s větším rozlišením) simulovat všechny barvičky skutečného nasnímaného obrázku i kdyby měl 24 bitů na pixel. A závěr? Velmi jednoduchý. Měli bychom apelovat na Commodore-Amiga, Inc., aby místo čipů, který odstraňují interlace (s ním se dá v grafice žít), vyráběli grafické čipy s rozlišením 640 x 512 bodů s přímým zobrazováním 4096 barev. Ideální by bylo samozřejmě obojí, ale myslím si, že ve světě je větší trh pro levné grafické stanice než DTP počítače, které vesměs pocházejí od firem jako jsou IBM a Apple. Do té doby Vám přeji hodně úspěchů s grafikou na našich Amigách a ať u nich všichni vydržíme a spolu přežijeme rok 2000 už s novými grafickými čipy.

Na závěr bych se chtěl vrátit ještě k jednomu problému, který nás poslední dobou velmi tlačí a kterému jsme také věnovali už hodně prostoru. Stále nám píšete o Vašich vesměs záporných zkušenostech s objednávkami časopisů Amiga Report a PCM. V mnohých dopisech nás prosíte, abychom se ujali celé záležitosti a podstoupili celou věc prokuratuře. Bohužel toto není v naší kompetenci. Nemůžeme podat hromadnou žalobu. Rozhodli jsme se ale pro jiné řešení a to, že podstoupíme Vaše dopisy Národnímu výboru, který vydal příslušné vydavatelské povolení, k posouzení celé záležitosti. Samozřejmě Vás budeme o všem informovat. Naše redakce si také objednala časopis Amiga Report u vydavatelství Fox Publishing, ale do dnešního dne (15. ledna 1991) jsme neobdrželi jedno jediné číslo. Z tohoto důvodu jsme se rozhodli podstoupit celou záležitost policii v Praze.

S pozdravem

Khaled Hussein

IMPRESSUM

(c) 1991 "amiga - měsíčník pro uživatele počítačů amiga." Nevyžádané příspěvky se nevracejí.

Ročník II. Cena 20 Kčs. Reg. č. MK ČR 5281. MIČ 46067.

Vydavatel: Jiří Próžr

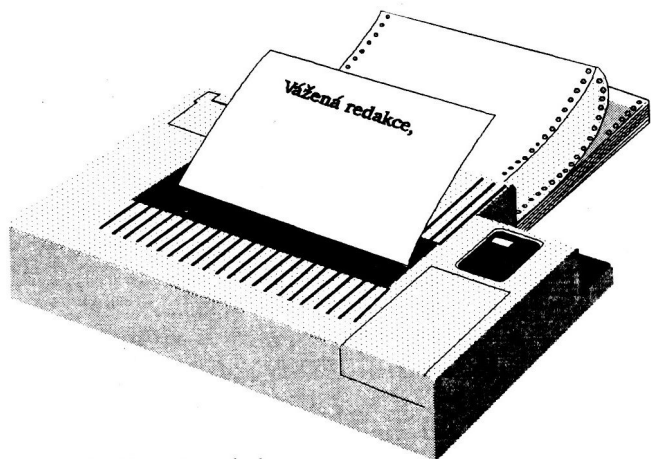
Šéfredaktor: Khaled Hussein Odpovědný redaktor: Tomáš Stibor

Fotografie: Petr Tůma Layout: Ing. Tomáš Daniček

Tisk: Ústav pro výzkum motorových vozidel

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

VAŠE DOPISY



Vážená redakce,

reaguji tímto na uveřejněné stížnosti některých čtenářů na pochybný časopis PCM, jehož otcem je "svěrázný podnikatel" pan M.L.

Reklamní akce propagující jeho časopis, podepřená navíc atraktivní soutěží (možnost vyhrát počítač Amiga 500), proběhla zhruba ve stejné době, kdy byl prezentován i vznik vašeho měsíčníku. Nulté číslo PCM, které jsem tehdy obdržel, se mi zdálo na solidní úrovni, a tak jsem podlehl "vtíravému" sloganu (na ty je zřejmě pan M.L. odborník), mluvícímu o denní radosti předplatitele. Radost z PCM sice nemám každý den, ale kdykoliv si vzpomenu na 96.- Kčs, věnovaných na podporu pana M.L., objeví se mi na tváři úsměv, který však ihned vystřídá obava, jak by to vypadalo, kdyby podobným způsobem "podnikalo" více lidí. Pevně však věřím, že registraci podobné činnosti bude místo Živnostenského úřadu nebo ministerstva provádět v budoucnosti pouze Prokuratura. Současně s PCM jsem si objednal na zkoušku i časopis Váš, který naštěstí neklamal. Přestože s úvodními čísly jsem po pravdě nebyl příliš nadšen, s postupující dobou se jeho informační hodnota i úprava viditelně zlepšuje. Proto si měsíčník Amiga objednáám i příští rok. Rozšíření počtu stránek velice vítám a doufám, že se tak dostane na podrobnější výklad v rubrice "Programování v jazyce C", a v budoucnu, že se dovíím i něco o GFA Basicu. Celkově si myslím, že by neškodilo ubrat něco stránek rozsáhlé kapitole "Hry" a věnovat je popisům kvalitních uživatelských programů, kterých je nepochybně rovněž velké množství. Hodně úspěchů do další práce v novém roce Vám přeje spokojený čtenář.

V.V.

České Budějovice

Vážená redakcia časopisu Amiga!

Využívam príležitosť a pri odoslaní predplatného by som Vám chcel poďakovať za to, čo robíte pre amigistov. Keď som si objednával Váš časopis rozmýšľal som, či nekupujem mačku vo vreci, pretože som mal skúsenosti s časopismi AMIGA REPORT a PCM (mimochodom dodnes mi ani po urgencii neprišlo ani jedno číslo z týchto časopisov a bohužiaľ nie som sám). Váš časopis ma zatiaľ neskla-

mal. A dúfam, že ma nesklame ani v budúcnosti. Aj keď ste zdvihli cenu, myslím si, že sa každému amigistovi oplatí objednať si Váš časopis. Verím, že aj naďalej bude Váš časopis písany takou formou, že aj začiatočník bude mať šancu väčšine článkov bez väčších problémov porozumieť. Želám Vám veľa elánu do Vašej užitočnej práce!

P.H.
Rožumberok

Amiga klub v Ostravě.

Členem se může stát každý, kdo vlastní počítač AMIGA, členské příspěvky se žádné zatím neplatí, ale v budoucnu se bude muset platit nájem budovy. Předseda, ani žádný vedoucí tam neexistuje. Průběh schůzky probíhá tak, že někdo donese počítač a po ukázce si povyměňujeme programy a různé informace. Pokud by někdo měl zájem přijít na schůzku, tak každý sudý pátek v 16.00 na adrese: Československá společnost elektroniků

Nádražní 44

Ostrava 1

Na této adrese se pouze scházíme, bližší informace na níže uvedené adrese:

Petr Hájek, Poštovní 581, Studénka II, 742 13

tel.:0655/71989

Vážení čtenáři,

vzhledem k tomu, že k nám do redakce dochází dopisy určené pro jiné adresáty uvádíme seznam adres a druh vyvíjené činnosti:

Redakce časopisu amiga -vydává časopis "amiga-měsíčník pro uživatele počítačů amiga"
Italská 29
Praha 2
120 00

Amiga Info -vydává literaturu pro počítače amiga na disketách
Post Box 729
Praha 1
111 21

Fox Publishing -měl by vydávat časopis Amiga Report (viz otištěné stížnosti)
P.O.Box 546
Praha 1
111 21

PCM -měl by vydávat časopis PCM a Počítačové noviny(viz otištěné stížnosti)
P.O.Box 414
Praha 1
111 21

Vaše redakce

Upozornění:Dne 23.2.19 91 pořádá Klub 602 v Raisově sále UKDŽ, Praha 2 Náměstí Míru "Sobotu s Commodorem" od 10.00 do 17.00 hodin. Zde bude provedeno veřejné slosování objednacích lístků našeho časopisu. Výherci budou uvědoměni i písemně. Inzeráty, které chcete zveřejnit v našem časopise (cena 9.- Kčs/cm2), musíme obdržet do 15 dne předcházejícího měsíce.

SOUROZENCI POWERPACKERU

Vlastimil Král

Ve svém článku "Kladivo na soubory" jsem vychválil cruncher PowerPacker. Tomu však (aspoň pokud se týká komprimace programových souborů) vyrostla konkurence v Imploderu 1.0 (v době, kdy toto píše, se připravuje článek o Imploderu), to však neznamená, že musíte PowerPacker "zahodit".

PowerPacker totiž kromě programových souborů může komprimovat i soubory datové. Stačí zvolit v PowerPackeru položku "Data Files" a soubory budou komprimovány jako data. Dokonce můžete data zakódovat. Kódování se zapíná položkou "Encrypt Data". Je-li kódování aktivováno, zeptá se PowerPacker před vlastní komprimací na heslo. Při zpětné dekompresi dat musíte toto heslo znát, jinak data "nerozbalíte".

Komprese dat je dost účinná, textové soubory lze zmenšit o 40%, někdy i o 50%.

S komprimovanými daty toho ovšem mnoho nepořídíte, pokud používáte standardní programy. Autor PowerPackeru Nico Francois dodává s tímto cruncherem i zdrojový kód funkce pro dekomprimaci dat. Funkce je napsaná v Aztec C a částečně i v assembleru. To dovoluje programátorům psát programy schopné přímo načítat komprimovaná data.

Ne každý je ovšem programátorem (zvláště pokud má Amigu). To si samozřejmě uvědomil i Nico Francois. Jeho odpovědi jsou další programy - "sourozenci PowerPackeru". Prvním z nich (a podle mého názoru důležitějším) je program PPMore.

Jak již název napovídá, je to ekvivalent programu More - slouží k prohlížení textových souborů. Jistě, různých ekvivalentů programu More je spousta - jmenujme aspoň Less a MuchMore.

Všechny tyto programy však dokáží prohlížet pouze normální textové soubory. To umí PPMore také, ale poradí si i se soubory zkomprimovanými PowerPackerem.

Chcete-li si prohlédnout nějaký soubor, stačí napsat

ppmore <soubor>

a otevře se okno "Listing of <soubor>" zobrazující zadaný soubor. Pokud zadáte neexistující soubor, objeví se na listě okna hlášení "Can't lock file !" a PPMore se ukončí. Velikost tohoto okna můžete měnit pomocí standardního gadgetu, pokud však okno zmenšíte příliš, zmizí část příkazové řádky. Po startu má okno pochopitelně maximální možnou velikost - zakrývá celou obrazovku.

Za "tvrdý" konec řádky považuje PPMore znak LF (0Ahex), jinak řádky láme podle aktuální šířky okna. Za stránku PPMore považuje aktuální výšku okna, pokud výšku okna změníte, změní se i délka stránky.

PPMore můžete volat pouze s názvem souboru a jen z CLI nebo Shellu. (Nezadáte-li název souboru, vypíše se obecný formát v angličtině.) Pokud však má textový soubor ikonu, můžete tam změnit DefaultTool na "<trasa>/ppmore". Naklapnete-li takovou ikonu, bude PPMore automaticky vyvolán (je to stejné jako u ostatních ekvivalentů programu More). Je-li text úspěšně zobrazen, můžete se jím pohybovat. Následující popis se týká verze 1.2, kterou mám k dispozici. O vyšších verzích se zmíním později. Chcete-li se pohybovat textem pomocí myši, můžete použít symbolů v nejspodnější řádce okna. Řádka nabízí tyto příkazy:

Next Page: Umožňuje posun na následující stránku.

Prev Page: Skok na předcházející stránku.

Next Line: Posun o řádku vpřed.

Prev Line: Posun na předchozí řádku.

Refresh: Je-li zobrazený text nějak poškozen, můžete ho tímto příkazem obnovit. Pokud například okno příliš zúžíte, budou nově zalomeny řádky. Obnovíte-li původní šířku, musíte použít tohoto příkazu, aby byly obno-

veny původní délky řádek.

Print File: Soubor bude vytisknuto přes zařízení "PRT:" (bude použit nastavený driver tiskárny).

Exit PPMore: Ukončí PPMore.

Samozřejmě můžete PPMore ovládat i pomocí klávesnice. Stisknete-li klávesu Help, "H" nebo "?", objeví se stručná anglická nápověda. Stisknete-li pak jakoukoli klávesu, zobrazí se opět prohlížený text. Příkazu Next page odpovídá stisk mezerníku nebo kombinace Shift + "kurzor dolů". Na předchozí stránku se posunete klávesou BACKSPACE nebo kombinací Shift + "kurzor nahoru". O řádku vpřed se dostanete klávesou Return nebo "kurzor dolů". Na předchozí řádku se posunete stiskem klávesy "kurzor nahoru". Text obnovíte klávesou "R", tisk textu se aktivuje stiskem "P". Na začátek textu se dostanete jednou z těchto dvou kláves: "<", "T". Chcete-li se rychle přesunout na konec textu, stiskněte buď ">", nebo "B". Klávesami Esc nebo "Q" PPMore ukončíte. PPMore také můžete ukončit, když zvolíte symbol pro uzavření okna ("close gadget").

PPMore podporuje standardní řídicí "escape" kódy, může tedy zobrazovat různobarevný text různých stylů (tučný text, kurzívu nebo podtržení). Protože soubor je tištěn přes zařízení PRT:, ovlivní tyto kódy i tisk (pokud je ovšem driver a tiskárna podporují). PPMore také podporuje české fonty - je-li v systému nastaven český font použije ho. Lze tedy bez problémů zobrazovat české (samozřejmě, i slovenské) komprimované texty. Pokud nastavujete font programem SetFont, musíte český font nastavit pro celý systém. PPMore dokonce podporuje i fonty různých rozměrů (s větší velikostí než 8), je-li však font příliš velký, mohou nastat problémy.

Tento popis se týká verze 1.2. Na disketě Fish #302 je verze 1.3, která by měla (podle recenzí v časopisech) umožňovat i skok na zadaný text. Tím by se PPMore již plně vyrovnal programu MuchMore. Na disketě Fish #334 je již verze 1.5, ta by se neměla příliš lišit od verze 1.3. Zatím však nemám žádnou z nich. PPMore načte celý soubor do paměti najednou (jinak by ho ani nemohl dekomprimovat). To je výhodné v tom, že při posunu textem není již

soubor znovu čten, je-li však soubor dlouhý, mohou nastat problémy s pamětí. PPMore je šířen ve zkomprimovaném stavu (pochopitelně, že je komprimován PowerPackerem) - pak má jen asi 6 KB. Na disketě tedy mnoho místa nezabere. Díky svým vlastnostem je to oblíbený program autorů disket utilit. Dokumentační soubory mohou být (díky kompresi) poměrně malé a vejde se jich tam více. Typickým příkladem jsou diskety řady Giant Utilities. PPMore by mohl být použit i na českých disketách utilit, které (jak doufám) se brzy objeví.

Druhým "PP programem" je program **PPShow**. Ten umožňuje prohlížet zkomprimované (ale i normální) obrázky v IFF formátu. Následující popis se týká verze 1.0. Volá se stejně jako PPMore:

ppshow < soubor >

Pokud zadáte soubor, který nemá požadovaný formát, objeví se hlášení "Not an IFF ILBM file!" a program se ukončí. Jestliže to obrázek je, program ho nejprve dekomprimuje (je-li to zapotřebí) a pak ho zobrazí na vlastní obrazovce. Příkaz "PPShow -h" vypíše krátkou anglickou nápovědu. Stisknete-li klávesu Tab, zapne se (nebo vypne) cyklická změna barev (tzv. color cycling). Grafici to jistě znají z DPaintu (mimochodem, tam se používá stejná klávesa pro aktivaci color cyclingu). Tento efekt umožňuje "pseudoanimaci". Máte-li klasickou hru "Defender of the Crown", můžete se podívat, co všechno tam tento efekt provádí (zajišťuje většinu pohybů ohně a vody). Stisknete-li klávesu "P", bude obrázek vytištěn s použitím parametrů nastavených v Preferences. Stisknete-li jakoukoli jinou klávesu nebo pravé tlačítko myši, program se ukončí a obrazovka se zavře.

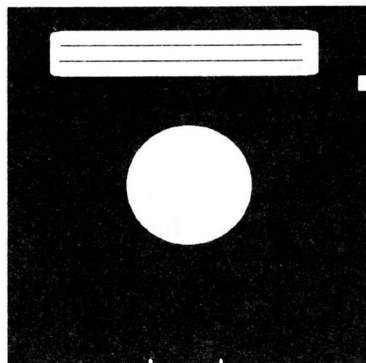
PPShow se uplatní především na demo disketách - zkomprimovaných obrázků se tam vejde více. Mají-li obrázky ikonu, můžete nastavit jako DefaultTool PPSHOW a po naklapnutí ikony bude PPSHOW automaticky vyvolán.

I PPSHOW je malý program - zkomprimovaný má jen 5KB. Tyto dva programy jsou Public Domain, je tedy možné je volně šířit. Nápadití amigisté jistě brzy přijdou na to, jak PPMore a PPSHOW využít - již se těším.



PRO ZAČÁTEČNÍKY

Kopírování



Tomáš Stibor

V tomto článku se pokusím nastínit problémy spojené s přesunem jednotlivých souborů z diskety na disketu a jejich řešení. Ponechme stranou etiku kopírování, protože ještě nenastala doba, kdy si budeme moci koupit program, který potřebujeme pro svoji práci nebo pro zábavu a nedat za něj dvě měsíční výplaty. Rady jsou určeny především začínajícím uživatelům, ale popis nadstavbových programů mohou, pokud není dobře znám, využít i ostatní.

Ke kopírování byl jistě každý z vás již donucen a je jedno jestli se jednalo o celou disketu nebo jen jeden soubor. Nejprve se budeme věnovat kopírování celé diskety. Pokud potřebujete zkopírovat úvodní diskety, jak vám radí v návodu k počítači - a originály uschovat na bezpečném místě, nejjednodušší je to provést přímo ve Workbench. Nejprve vyberete ikonu diskety, kterou chcete kopírovat, tzn. jednou na ní cvaknete levým tlačítkem myši. Okno diskety není třeba otevírat. Pak na liště Workbench Screen zvolíte v menu Workbench položku Duplicate. Nyní budete vyzváni ke vložení originální diskety (kterou kopírujete) do mechaniky df0:. Po nahrání několika stop z diskety do paměti počítače vás požádá o výměnu za novou disketu (na kterou kopírujete). Tak se to opakuje několikrát, až je celá disketa přenesena. Nová kopie má název rozšířený o slova "Copy of". Tento způsob kopírování

má určité nevýhody. Je pomalý, pracuje jen s jednou mechanikou i když jich máte připojeno více a kopie není naprosto identická s originálem. Dále to nemusí vždy fungovat. Místo vyzvání k vložení diskety se může nahoře objevit chybové hlášení "Error while opening SYS:System/DiskCopy 205:". To proto, že na disketě ze které jste spustili počítač a nahráli Workbench není direktorář (vysvětlíme si později) System a v něm program DiskCopy, jak kopírovaná disketa vyžaduje podle položky Default Tool. Toto uvidíte, když ve stejném menu jako je Duplicate zvolíte Info.

Z těchto důvodů bylo vytvořeno mnoho kopírovacích programů, které se liší komfortem obsluhy, možnostmi kopírovat nekopírovatelné, počtem použitelných mechanik a také zbytečnými efekty. Nyní si některé z nich popíšeme. Pro majitele pouze samotného počítače je důležité hledisko kolik místa v paměti kopírovací program umí využít. To je přímo úměrné počtu přendávání originálu a kopie v diskové mechanice. Na disketě je 880kB místa, ale v základní verzi Amigy jen 512kB, něco paměti spotřebuje systém, další kus vlastní kopírovací program a pokud hraje všemi barvami, tak na kopírování mnoho místa nezbyvá. Některé programy to řeší kompresí dat v počítači.

Tak například staříček **White Lighting** vám při nerozšířené paměti zkopíruje disketu na dvakrát a s

rozšířením na 1MB najednou. Další možnosti tohoto programu jsou:

Verifikace kopie - to znamená, že co počítač napíše na disketu, si také po sobě hned přečte a zkontroluje. Doba kopírování se ale prodlužuje.

Ignorování chyb čtení - při chybě čtení originálu se kopírování nepřerušuje.

Několikanásobné kopírování - použijte pokud kopírujete jeden originál na více disket.

Formátování diskety - účel si vysvětlíme později.

Kopírování jen určitých stop - těžko budete potřebovat.

Novější kopírovací programy mají další vymoženosti jako kopírovat i přidané stopy na disketě až do počtu 83, různý mód kopírování pro originální programy, které jsou chráněny proti rozmnožování a možnost nastavení dalších parametrů. Jako představitel této skupiny kopírovacích programů je nutno uvést např. X-Copy III nebo dále Blue Point, který sám mění mód kopírování podle potřeby.

Ochrana firem proti nežádoucímu množení však také přináší stále něco nového. Již od loňského roku máme v redakci originál hry Harrier. Tato disketa se zatím žádným programem nedala s úspěchem zkopírovat.

Ne vždy je však potřeba kopírovat celou disketu. Dejme tomu, že na některou disketu chcete přenést nějaký oblíbený program - třeba hodiny nebo odvírovač. Nyní je již důležité o jakou disketu (cílovou - target) se jedná. Jestli je na ní již něco nahráno, anebo je nová, ještě nepoužitá. Pokud je nová, je nutno ji nejdříve naformátovat. Toto neplatí, pokud kopírujete celou disketu najednou - formátování se provádí automaticky.

Co to vlastně to formátování je. Nepoužitá disketa je kolečko z magnetofonového pásku a aby se na ní počítač vyznal, musí si ji upravit asi tak, jako vypadá gramofonová deska, na které je větší počet skladeb. Počítač si na magnetickém kolečku připraví stopy, které však nejsou ve spirále, ale v kružnicích, právě tak, jako jednotlivé písničky na desce. Jinak by jednotlivé stopy nebylo možno rychle najít. Dále si počítač stopy očíslovuje a vyhradí si místo, kam si bude psát, kde má co uloženo a kde je ještě volné místo. To

aby nemusel prohledávat pokaždé celou disketu. Používáním se disketa opotřebovává, práší se na ni a i jinak se může poškodit. Pak jsou některé stopy těžko čitelné, což se projevuje delší dobou nahrávání programu a až úplnou nečitelností. Než to nastane je nutno disketu zkopírovat a znova naformátovat. Počítač si nyní označí vadná místa na disketě a je možno ji dále používat, ovšem již s menší kapacitou. Formátování se dělá buď ve Workbench v menu Disk položka Initialize. Opět musíte mít myši označen disk, který chcete formátovat a též může nastat chybná situace jako při kopírování celé diskety. Většina kopírovacích programů však též umí elegantně formátovat.

Na naformátovanou disketu je tedy možno kopírovat jednotlivé soubory. Opět nejjednoduší je to ve Workbench, ale ne vždy použitelné. Pokud má soubor, který chcete kopírovat ikonu, otevřete okno diskety, na které se nalézají. Pak do počítače vložte cílovou disketu, opět otevřete okno a pomocí myši přetáhněte ikonu souboru do nového okna. Počítač si bude střídavě žádat původní a cílovou disketu a nakonec se ikona souboru objeví v obou oknech. S programy a soubory, které ikonu nemají, je to těžší. Musíme kopírovat v CLI okně pomocí příkazů AmigaDOS nebo pomocí speciálních kopírovacích programů. Jestliže jste vlastníky druhé a další diskové mechaniky, kopírujete přímo, jinak přes RamDisk. RamDisk je část paměti počítače, která se chová jako další disketa. Je na něj možno něco nahrát, spouštět z něj programy, ale když vypnete počítač, vše se vymaže. (Pochopitelně to jde i bez RamDisku, ale je to prakticky neustálé přendávání disket v diskové mechanice.)

Nejdříve tedy kopírování v CLI (totéž platí pro Shell). Otevřete okno CLI a přesvědčte se, že soubor je opravdu na disketě a ve správném direktoráři. Nyní by asi pro některé uživatele bylo příhodné vysvětlit, co to jsou direktoráře. Z lektorské praxe vím, že to je věc laikům nejméně pochopitelná co do smyslu a využití. Vy ostatní následující odstavec přeskočte.

Direktorář. Představte si disketu třeba jako velkou zásuvku psacího

stolu. Pokud do ní všechny svoje věci nasypete přímo, těžko se tam pak vyznáte a budete něco hledat. A kdyby zásuvku používalo víc lidí, nastal by dokonalý zmatek. Navíc žádné dvě věci se nesmí stejně jmenovat. Proto si v šupletě uděláme přihrádky, každou nějak nazveme, a do nich teprve dáme svoje věci. Pochopitelně v přihrádce na tužky můžeme mít další přihrádku na nové tužky ještě neořezané. A tak pořád dále. V té chvíli, když budu mít ve své přihrádce dopis a kolega také ve své přihrádce dopis, tak, ač se jmenují stejně, nespletou se. Některé přihrádky - direktoráře jsou pro počítač důležité a jejich název se nesmí měnit.

Které soubory jsou v direktorářích zjistíte příkazem DIR. Po jeho spuštění se na obrazovku začnou vypisovat postupně názvy jednotlivých direktorářů diskety a názvy souborů, které nepatří do žádné přihrádky. Pokud chcete vědět jaké soubory máte třeba v přihrádce Tužky, musíte napsat příkaz CD tužky, kterým se počítač přesune do tohoto direktoráře. Po opětovném zadání DIR počítač vypíše, že zde je další přihrádka Nové-Tužky a vše ostatní. Soubory, které jsou mimo tuto přihrádku, jako by pro počítač neexistovaly. Zpět do základního direktoráře (šuplete) se dostanete napsáním např. CD DF0:. Když tedy chcete s nějakým souborem jakkoli pracovat (tedy i kopírovat), musíte se nejdříve přemístit pomocí příkazů CD až do té správné přihrádky, nebo počítači zadat tzv. úplnou cestu všemi direktoráři až k souboru. Např.: DF0:Tužky/Nové-Tužky/Modrá3. Pokud nepoužíváte názvy s čárkami a háčky je počítači jedno, jestli píšete malými nebo velkými písmeny.

To platí též pro příkazy. Když budete vyžadovat název s mezerou uprostřed - Nové Tužky - musíte jej používat v uvozovkách. A nyní již k vlastnímu kopírování. Provádí se příkazem COPY, za nějž musíte uvést co a kam kopírovat. Správný postup, jak psát příkazy a jejich parametry, se nazývá syntax příkazů. Způsob jeho vysvětlování v návodě k počítači - šablona - je poněkud šroubovaný. Proto se budu věnovat spíše příkladům. Co je v hranatých závorkách nemusí být použito.

COPY co kam [parametry]

co - to je název souboru, který chceme kopírovat

kam - název direktoráře, kam se má kopírovat, popř. i název souboru, pokud chceme, aby se kopie jmenovala jinak

Příklad:

COPY dopis ram: kopíruje soubor dopis do RamDisku. Totéž je možno vylepšit o anglická slovíčka FROM = z TO = do Pak to vypadá takto: COPY FROM dopis TO ram:. A nyní soubor kopírujeme zpět na jinou disketu: COPY ram:dopis Texty: kde Texty: je název diskety, na kterou chcete kopírovat. Pokud by jste napsali jen df0:, tak by to počítač chtěl nahrát na disketu, ze které jste ho spustili. A se vši parádou: COPY FROM ram:dopis TO Texty:Nyní kopírování s přejmenováním: COPY ram:dopis Texty:pismo Kopírování do konkrétního direktoráře (musí existovat): COPY ram:dopis Texty:nabidky/služby kde Texty: je název diskety, nabídka je název direktoráře a v něm je další direktorář s názvem služby.

Toto je tedy jeden způsob, jak zkopírovat jednotlivé soubory. V praxi se to však takhle krkolomně nedělá. Na první pohled vidíte, že to je dobré tak maximálně pro počítače PC. My ale máme Amigu a proto si zavoláme na pomoc nějaký program tzv. nadstavbu operačního systému. To znamená, že se v něm dají dělat skoro ty samé věci jako v okně CLI, jenže mnohem elegantněji a za pomoci myši. Jak jste si jistě všimli, myš v CLI není prakticky k ničemu. Jedna z nejjednodušších nadstavb systému je **CliMate**. Pro základní práci je plně dostačující. Popíšeme si verzi 1.201.

Na horní liště je v pravé části přepínač (přepíná se levým tlačítkem myši) Fastdir. Při zapnutí (ON) program vytváří soubor .fastdir, který obsahuje seznam všeho, co je v každém direktoráři. Hlavní efekt toho je, že tyto soubory zabírají na disku místo. Proto je lepší Fastdir na začátku práce vypnout (OFF). Hlavní částí obrazovky jsou dvě tabule, na kterých máme možnost si zobrazit obsah diskety ve zvolené mechanice. Levá tabule je označena Source - zdroj, pravá Destination - cílová. Ta,

kteřá je aktivní, má kolem názvu rozsvícené šipky vzhůru. Přepínání se (jako vše ostatní) děje levým tlačítkem myši. Mezi tabulemi je seznam všech diskových mechanik včetně RamDisku, které program objevil. Po vybrání mechaniky myši se do aktivního okna vypíší soubory z disku černě a direktoráře bíle. Nad tabulí je napsáno kolik zbývá na disku volného místa. V seznamu je možné listovat pomocí šipek na boku tabule. Pokud na některý soubor umístíte myš, řádek zmodrá a v horním políčku File se zobrazí jeho délka. Když ještě navíc cvaknete levým tlačítkem myši řádek zčervená a je vybrán k dalšímu použití. Na direktoráře platí naopak pravé tlačítko myši, aby ukázali co se v nich skrývá. Zpět z direktorářů se dostanete pomocí volby Parent. V dolní části obrazovky jsou čtyři velké modré rámečky s nápisy BATCH, RENAME, DISPLAY a SHOWPIC, které určují hlavní okruhy použití programu. Pro kopírování je nutná první volba - BATCH. V závislosti na ní je kolem několik menších položek s konkrétními příkazy. Začneme úplně vlevo:

Format - formátování diskety. Po jeho zvolení se nakreslí další okno na podrobné parametry - v které mechanice formátovat, jak se má disketa jmenovat, zda provádět verifikaci a jestli chcete nahrát na disketu direktorář s ikonou popelnice. Pokud je vše v pořádku a zvolíte OK, tak se program ještě jednou ujistí, že jste si to nerozmysleli a začne formátovat. Dvojí odsouhlasení je zde proto, že formátování je proces, při kterém všechny soubory na disketě vezmou za své a není možná náprava. **All** - tím se všechny soubory aktivní tabule stanou vybrané, včetně direktorářů a jejich obsahů. Jste-li ale již sám v nějakém direktoráři, nezahrnuje to soubory a direktoráře vně vašeho umístění.

Clear - opak příkazu All - žádný soubor není vybraný.

Copy - kopírování vybraných souborů z levé tabule (Source) do zařízení na pravé tabuli (např. Ram:). Během kopírování program podává hlášení o tom co je zrovna na řadě. Po zkopírování se program ptá, jestli má zrušit označení vybraných souborů. Pokud se všechny vybrané programy na disketu nebo do RamDisku neve-

dou program to ohlásí a u nezkopírovaných souborů označení nezruší. **Move** - totéž co kopírování s tím rozdílem, že zkopírované soubory se na zdrojové straně vymažou.

Delete - smaže vybrané soubory ze zvolené diskety.

Pattern - označí za vybrané soubory, které vyhovují předloze. Pozor, pracuje se znaky jako PC, které jsou poněkud pochopitelnější. * - znamená jakoukoliv skupinu znaků, ? - znamená jakékoliv písmeno napíšete-li P* vyberou se všechny soubory, které začínají na P, a např. ?A? označí ty soubory, mající název ze tří znaků, z nichž prostřední je A.

Print - vytisknutí vybraného souboru. Opět následuje okno s podrobným nastavením, které se přebírají z preferencí a je možno je měnit tak, že na položce stlačíme tlačítko myši a současně pohybujeme myší nahoru nebo dolů.

Makedir - vytvoří na aktivní disketě direktorář se zadaným jménem.

Rename - příkaz přejmenovává soubory na název, který se vám bude zdát nejlepší.

Display - program se zeptá, zda chcete výpis v hexadecimálním kódu nebo dekadicky.

Showpic - ukáže soubor jako obrázek, pokud tak byl vytvořen. (viz článek o souborech IFF).

Podobných programů existuje celá řada a liší se v možnostech nastavení barev, písma, programy editovat nebo přímo spouštět, ale čím více funkcí, tím bývá i program nepřehlednější. Pokud jste jen uživatelé počítače, používejte ho ke své práci nebo k zábavě a složité problémy nechte stranou. Doba, kdy uživatel musel být i programátor již naštěstí začíná být minulostí a může se snadno stát, že po mnoha hodinách zkoušení nějakého programu přijdete na něco, co je odborníkům samozřejmé a vy jste si mohli ušetřit čas a nervy. A jestliže vám něco hned nepůjde tak, jak byste si přáli, napište do redakce. Pokusíme se odpovědět na dotazy a při větším zájmu vytvoříme rubriku pro "experty všech oborů kromě výpočetní techniky", která by byla takovou vaší První pomocí. Příště si popovídáme o programu Preferences a o prohlížení obsahu souborů.





Bezesporu nejpoužívanějším a snad i nejefektivnějším programovacím jazykem pro Commodore Amiga je jazyk C. Sám o sobě, tak jak je standardizován, by však nedovedl skoro nic. Čím je tedy způsobena možnost vytvořit bezmála celého systému využívající kvalitní program právě v tomto jazyce? Důvodem není nic jiného než použití systémových knihoven, které skoro úplně obstarají obsluhu Amigy.

Vzhledem k tomu, že je C jazykem na nižší úrovni než Basic, je zde použití knihoven poněkud složitější. Pochopení výkladu této problematiky je podmíněno alespoň základními znalostmi o používání zmíněného jazyka ve spojení s počítačem Amiga, přesněji s jeho operačním systémem.

SOUBORY INCLUDE

Pro zachování či předávání rozsáhlých informací, týkajících se prvků operačního systému, používá tento systém takzvaných struktur. Pokud Vám v tuto chvíli vytanuly na mysli již kdysi v podobném sloupku zmíněné RastPort nebo ViewPort struktury, stalo se tak správně. Systémové struktury se ale samozřejmě netýkají jen grafiky.

Pro řadového smrtelného programátora je nemožné si stavbu těchto struktur zapamatovat. Je potom možno tvrdit, že tedy nebyl pro Amigu vyroben žádný program? Ne, není tomu tak díky tzv. INCLUDE souborům, ve kterých jsou snad všechny potřebné a používané struktury definovány popsáním. INCLUDE soubory se nacházejí na Vaší jistě legální C - disketě v adresáři INCLUDES a jsou rozděleny podle jednotlivých kategorií použití. V těchto souborech nejsou pochopitelně definovány jenom systémové struktury, ale i často používané konstanty apod.

Aktivace určitého INCLUDE souboru se provede jeho zahrnutím, připsáním k programu. Právě k tomu jsou tyto soubory předurčeny (include - zahrnout). K tomu slouží C - instrukce #include. Před kompilací se INCLUDE souborem tato instrukce nahradí. V jeho jméně se

neuvádí adresář 'INCLUDES'. Do svého programu zahrňte vždy jen ten soubor, ve kterém je Vám potřebná struktura. Vykonání příkazu include a následná kompilace jsou poměrně dost časově náročné. Instrukce include se obvykle používá na úplném počátku zdrojového textu C - programu.

STAVBA STRUKTURY, UKAZATELE NA STRUKTURY

Stavbu struktury lze vysvětlit na krátkém příkladě. (I přesto, že se bude jednat o určité zjednodušení.)

```
struct Image{
SHORT LeftEdge;
SHORT TopEdge;
SHORT Width, Depth;
USHORT *ImageData
UBYTE PlanePick, PlaneOnOff;
struct Image *NextImage;}
```

Za identifikátorem 'struct' následuje název struktury a uzavřen v závorkách je vždy uveden typ proměnné, přesněji prvku a jeho název, popřípadě více názvů prvků stejného typu. Prvek může být i ukazatelem (*ImageData) nebo odkazem na další strukturu (v tomto případě na stejnou) s uvedením paměťového ukazatele.

Budeme-li používat např. strukturu 'Window', v níž se uchovávají informace o již existujícím okně,
- zavedeme ji do programu příkazem

```
#include <intuition/intuition.h >
```

- a deklarujeme na ni ukazatel tímto způsobem:

```
struct Window *w;
```

Jméno ukazatele je 'w' a 'Window' je odkaz na strukturu.

Chceme-li vytvořit novou strukturu dle napsané strukturové definice s vlastními údaji, budeme postupovat následovně:

```
struct JmenoDefiniceStruktury JmenoVlastniStruktury
{
hodnoty odpovídající prvkům struktury
}
```

Např. když budeme vytvářet vlastní strukturu podle definice Image bude mít instrukce následující podobu:

```
struct Image MyImage
{
6,
7,
8,9
... atd ...
}
```

MyImage je název nové struktury. Prvku LeftEdge odpovídá hodnota 6, TopEdge odpovídá 7 atd. Hodnotu prvku ze struktury získáme pomocí znaku '->' s touto syntaxí:

```
proměnná = (NázevStruktury - NázevPrvku);
```

Jestliže chceme kupříkladu zjistit ukazatel na aktivní okno ze struktury IntuitionBase (tedy hodnotu prvku Acti-

veWindow), instrukce bude vypadat následovně:

```
w = (IntuitionBase -> ActiveWindow);
```

OTEVŘENÍ A POUŽITÍ KNIHOVEN

Některé knihovny musíme před jejich použitím otevřít. To vykonáme funkcí `UL = OpenLibrary("JmenoKnihovny", VerzeKnihovny)` nebo `UL = OldOpenLibrary("JmenoKnihovny")`. Funkce vrátí ukazatel na strukturu, do které si knihovna ukládá informace. Název této struktury je zakončeno příponou 'Base'. (Např. `intuition.library` - `IntuitionBase`, `graphics.library` - `GfxBase`) Otevření a použití knihoven si opět osvětlíme na příkladě.

```
#include <intuition/intuitionbase.h>
#include <intuition/intuition.h>
struct IntuitionBase *IntuitionBase;
struct GfxBase *GfxBase;
struct RastPort *rp;
struct Window *w;

static BYTE AlertStr[] =
{
    0,80,20,
    " Stla c le ve tlacitko m y si.",0
};

struct NewWindow nw1 = {
    100, 50,
    400, 150,
    0, 1,
    NULL, NULL,
    NULL, NULL,
    NULL, NULL, NULL,
    5, 5,
    640, 200,
    WBENCHSCREEN };

main()
{
    int a;
    if (!(IntuitionBase = OpenLibrary("intuition.library",0L)
    )) {exit();}
    if !(GfxBase = OpenLibrary("graphics.library",0L) )
    {CloseLibrary(IntuitionBase);exit();}
    if !(w = OpenWindow(&nw1) )
    {;CloseLibrary(GfxBase);CloseLibrary(IntuitionBase);}
    rp = w->RPort;
    SetAPen(rp,2);
    RectFill(rp,0,0,400,150);
    SetAPen(rp,3);
    Move(rp,120,20); Text(rp,"TOTO JE O K N O",16);
    DisplayAlert(RECOVERY_ALERT,&AlertStr[0],40L);
    CloseWindow(w);
    CloseLibrary(GfxBase);
    CloseLibrary(IntuitionBase);
}
```

Program nejprve zahrne `INCLUDE` - soubory, deklaruje ukazatele, vyplní potřebné struktury, otevře knihovny, otevře okno, které vyplní, a použije funkci `DisplayAlert()` pro zobrazení známého červeně ohraničeného textu na vrcholu obrazovky. Nakonec zavře okno a knihovny.

GRAFIKA Z POHLEDU PROGRAMÁTORA V JAZYCE C

Khaled Hussein

Tento měsíc bych se chtěl podívat na systémové struktury a knihovní funkce, které jsou zapotřebí na napsání programu, který by zobrazoval IFF soubory typu ILBM (viz článek o standardu vzájemného přenosu IFF, který vyšel v lednovém čísle amiga). Všechny struktury a funkce budou popsány z hlediska začátečníka, který je ještě nikdy neviděl, a tudíž vynechám nepotřebné detaily, ale na druhou stranu zase vše bude popsáno alespoň jednou větou. Nebudete-li rozumět všemu, nemusíte zoufat, protože chystám samostatný článek o různých detailech zobrazování. Tady se teda zaměřím na to, jak vůbec otevřít obrazovku, okno, vyměnit paletu barev a popřípadě něco nakreslit vkládáním nějakých hodnot na příslušné bitplanes. (Bitplanes - tento pojem byl vysvětlen v článku `Hold and Modify`, amiga 2/90.) Účelem tohoto článku je seznámit čtenáře s grafickým zobrazováním, abyste mohli jednoduše pochopit program, který zobrazuje ILBM obrázky, který v nejbližší době otiskneme.

BEZ OBRAZOVKY TO NEJDE

Abychom mohli otevřít svoji vlastní obrazovku (a později na ni svá okna), musíme udělat několik přípravných prací. Za prvé musíme počítači sdělit jakou obrazovku si přejeme a potom tyto údaje předat knihovní funkci, která obrazovku otevře, je-li v systému dostatek volné CHIP paměti. Otevřeli se obrazovka, vrátí nám funkce ukazatel na množství dat popisující otevřenou obrazovku. A teď už konkrétněji. Následující strukturu budeme potřebovat na to abychom popsali naši obrazovku systému.

```
struct NewScreen {
    SHORT LeftEdge, TopEdge, Width, Height, Depth;
    UBYTE DetailPen, BlockPen;
    USHORT ViewModes;
    USHORT Type;
    struct TextAttr *Font;
    UBYTE *DefaultTitle;
    struct Gadget *Gadgets;
    struct BitMap *CustomBitMap;
};
```

LeftEdge - počáteční x-ová souřadnice obrazovky. Momentálně není využívána. Inicializujeme vždy hodnotou 0.

TopEdge - počáteční y-ová souřadnice obrazovky. Zadáme číslo řádku, na kterém má obrazovka začínat. Většinou chceme úplně horní řádek, tudíž zase 0.

Width - šířka obrazovky. Buď 320 nebo 640.

Height - výška obrazovky. Buď 256 nebo 512.
Depth - počet bitplanes. Můžeme použít hodnot od 1 do 6, ale musíme dávat pozor, aby odpovídaly zvolenému rozlišení.
DetailPen - číslo registru barvy, které chceme použít pro detaily, jako je název obrazovky.
BlockPen - číslo registru barvy, které chceme použít pro vyplňování objektů, jako je lišta menu.
ViewModes - módy zobrazení. Můžeme použít jen jedno nebo kombinace několika z nich:
HIRES - vysoké horizontální rozlišení (640). Standardně je nastaveno 320.
LACE - nastaví prokládání řádek (512 řádek).
SPRITES - nastavíme tehdy, chceme-li použít sprites (pohyblivé objekty).
DUALPF - nastavíme tehdy, chceme-li dvě playfields (na sobě nezávislé "obrazovky", které se zobrazují na jednu fyzickou obrazovku).
HAM - nastavíme tehdy, chceme-li HAM mód.
EXTRA_HALFBRITE - nastavíme tehdy, chceme-li extra-halfbrite mód.
Type - typ obrazovky:
CUSTOMSCREEN - znamená, že naše obrazovka bude skutečně naše vlastní. Všechny obrazovky, co kdy vytvoříme, budou tohoto typu.
SCREENBEHIND - znamená, že se naše obrazovka otevře za všechny už otevřené obrazovky.
SCREENQUIET - znamená, že systém nenakreslí systémové věci na obrazovku, jako je například lišta menu.
CUSTOMBITMAP - znamená, že chceme vše obhospodařovat sami (paměť, bitmap atd.).
Font - ukazatel na strukturu popisující font, který se má použít pro tuto obrazovku. My budeme používat vždy systémový font a tudíž inicializujeme tento ukazatel na NULL.
DefaultTitle - titulek obrazovky, resp. ukazatel na text titulku, který je zakončen nulovým bytem (0).
CustomBitMap - ukazatel na strukturu BitMap. Souvisí těsně s typem obrazovky CUSTOMBITMAP.

Až si inicializujeme tuto strukturu podle svých přání, můžeme zavolat funkci **OpenScreen()** z knihovny **intuition.library**, aby otevřela naši obrazovku.

```
struct Screen *ms;
ms = OpenScreen( &myscreen );
```

kde **myscreen** je název naší inicializované struktury.

Jak jste si už určitě všimli, vrací funkce **OpenScreen()** nějakou hodnotu. Nemohla-li z nějakého důvodu obrazovku otevřít, vrátí hodnotu NULL. V opačném případě vrátí ukazatel na strukturu **Screen**, kterou si teď popíšeme. (Z toho tedy vyplývá, že bychom se měli vždy přesvědčit, zda vrácená hodnota je nenulová.)

```
struct Screen {
    struct Screen *NextScreen;
    /* spojový seznam všech obrazovek */

    struct Window *FirstWindow;
    /* spojový seznam všech oken této obrazovky */

    SHORT LeftEdge, TopEdge;
```

```
/* x-ová a y-ová souřadnice levého horního rohu obrazovky */
```

```
SHORT Width, Height; /* šířka a výška obrazovky */
```

```
SHORT MouseX, MouseY; /* poloha ukazatele myši vztahená k levému hornímu rohu obrazovky */
```

```
USHORT Flags; /* přepínače, který nastavuje systém, podle našich přání; například chceme či nechceme vidět lištu menu */
```

```
UBYTE *Title; /* titulek obrazovky */
```

```
UBYTE *DefaultTitle; /* pro okna, které nemají ScreenTitle */
```

```
BYTE BarHeight, BarVBorder, BarHBorder;
```

```
BYTE MenuVBorder, MenuHBorder;
```

```
BYTE WBotTop, WBotLeft;
```

```
BYTE WBotRight, WBotBottom;
```

```
/* velikost listů a rámeček této obrazovky a všech oken na této obrazovce */
```

```
struct TextAttr *Font; /* font této obrazovky */
```

```
struct ViewPort ViewPort; /* popisuje zobrazení této obrazovky (display) */
```

```
struct RastPort RastPort; /* popisuje zobrazování v této obrazovce (rendering) */
```

```
struct BitMap BitMap; /* kopie navíc struktury BitMap v RastPort */
```

```
struct Layer_Info LayerInfo; /* LayerInfo této obrazovky */
```

```
struct Gadget *FirstGadget; /* spojový seznam gadgetů této obrazovky; momentálně obsahuje jen systémové gadgety */
```

```
UBYTE DetailPen, BlockPen; /* čísla registru barev pro psaní textu a vyplňování objektů */
```

```
/* následující struktury jsou pro vnitřní potřebu systému */
```

```
USHORT SaveColor0;
struct Layer *BarLayer;
UBYTE *ExtData;
UBYTE *UserData;
```

```
};
```

Tato struktura nás bude zajímat jen z hlediska toho, že některé údaje poskytneme určitým funkcím, které nám vykonají určité služby, ale nebudeme se zajímat o to, co vlastně je v těchto údajích zakódováno až na jednu výjimku. Tím je struktura **RastPort**, která obsahuje ještě jednu strukturu a to strukturu **BitMap**.

```
typedef UBYTE *PLANEPTR;
```

```
struct BitMap {
    UWORD BytesPerRow;
    UWORD Rows;
    UBYTE Flags;
    UBYTE Depth;
    UWORD pad;
    PLANEPTR Planes[8];
};
```

Zde nás bude zajímat poslední položka, která obsahuje ukazatele na všechny bitplanes. Sem budeme zapisovat naše data, které chceme zobrazovat.

BEZ OKNA JE TMA

Když už máme svoji obrazovku otevřenou, musíme otevřít ještě okno, kde budeme kreslit. Otvírání oken se děje úplně stejně jako otvírání obrazovky, až na to, že použijeme jiných struktur a jiné funkce. Tady také vrací funkce nějakou hodnotu, ale zde nás bude zajímat jen to, zda není rovna NULL, tedy zda se naše okno otevřelo. Dokonce při popisu struktury, vynecháme většinu detailů, protože si zaslouží vlastní články.

```
struct NewWindow {
    SHORT      LeftEdge, TopEdge;
    SHORT      Width, Height;
    UBYTE      DetailPen, BlockPen;
    ULONG      IDCMPFlags;
    ULONG      Flags;
    struct Gadget *FirstGadget;
    struct Image *CheckMark;
    UBYTE      *Title;
    struct Screen *Screen;
    struct Bitmap *BitMap;
    SHORT      MinWidth, MinHeight;
    USHORT     MaxWidth, MaxHeight;
    USHORT     Type;
};
```

LeftEdge - počáteční x-ová souřadnice levého horního rohu okna.

TopEdge - počáteční y-ová souřadnice levého horního rohu okna.

Width - šířka okna v pixels.

Height - výška okna v pixels.

DetailPen, BlockPen - jako u struktury **NewScreen**.

Flags - určují vzhled a funkce okna (neuvádím všechny):

WINDOWIZING - je možné měnit velikost okna.

WINDOWDEPTH - je možné posílat okno za nebo před jiná okna.

WINDOWCLOSE - je možné zavřít okno pomocí tomu určeného gadgetu.

WINDOWDRAG - je možné posunovat okno po obrazovce.

GIMMEZEROZERO - všechny souřadnice budou vztaženy k hornímu levému rohu plochy, na kterou se kreslí.

SIMPLE_REFRESH - musíte se starat o obnovení obrazovky, když bude překryta nějakým jiným oknem a zase odhalena.

SMART_REFRESH - o obnovení obsahu okna se stará systém.

ACTIVATE - okno bude aktivní v okamžiku otevření.

IDCMPFlags - přepínače pro systémovou komunikaci s oknem. Zde neuvádím.

FirstGadget - spojový seznam gadgetů.

CheckMark - popis znaku, který ukazuje zvolenou položku v menu.

Title - titulek okna.

Type - Buď **WBENCHSCREEN** nebo **CUSTOMSCREEN**. V prvním případě se okno otevře na Workbench. V druhém případě se okno otevře na obrazovku určenou položkou **Screen**.

Screen - ukazatel na obrazovku, na kterou se má okno otevřít; je-li to Workbench, inicializujeme na NULL.

BitMap - chcete-li obhospodařovat vlastní bitmap, je toto ukazatel na něj. Musíte potom nastavit ve **Flags** přepínač **SUPER_BITMAP**.

MinWidth - minimální šířka okna v pixels.

MinHeight - minimální výška okna v pixels.

MaxWidth - maximální šířka okna v pixels.

MaxHeight - maximální výška okna v pixels.

Po inicializování této struktury otevřeme okno následovně (**mywindow** je název inicializované struktury):

```
struct Window *mw;

mywindow.Screen = ms;
mw = OpenWindow( &mywindow );
```

Tak a teď už máme otevřenou obrazovku a na ní okno.

BEZ BAREV JE TMA

Nyní si ukážeme jak si můžeme nastavit barvičky obrazovky. Pro náš příklad budeme uvažovat, že máme tři bitplanes, což znamená osm barev. Napřed musíme nadeklarovat strukturu, která bude obsahovat definice našich barev a inicializovat ji s příslušnými hodnotami:

```
UWORD ColorMap[] = {
    0x000, /* bílá */
    0x00f, /* modrá */
    0x0f0, /* zelená */
    0xf00, /* červená */
    0xfff, /* černá */
    /* a teď už jen mícháme a mícháme */
    0xbbb,
    0xc00,
    0xb45
};
```

Jak vidíme zadáváme hodnoty od 0 (bílé - 0x000) do 4095 (černá - 0xfff). To znamená, že máme 12 bitů na pixel, což zase znamená, že máme čtyři bity pro každou složku barvy (červená, zelená, modrá). Tedy jednoduše řečeno můžeme si vybrat z palety 4096 barev.

A teď pomocí funkce **LoadRGB4()** z knihovny **graphics.library** požádáme systém o tyto barvy následujícím způsobem:

```
LoadRGB4(ms->ViewPort, ColorMap, 8);
```

kde 8 znamená počet barev.

Teď už máme vše připraveno pro kreslení, které proběhne podrobně příště. Ale přeci jen Vám ještě něco ukáži. Víte jak nakreslíme osm červených bodů do horního levého rohu obrazovky? Nevíte. Tak se dobře dívejte a do příště se pokuste přijít na to, jak se to vlastně dělá.

```
ms->RastPort.BitMap->Planes[0] = 255;
ms->RastPort.BitMap->Planes[1] = 255;
ms->RastPort.BitMap->Planes[2] = 0;
```

Pamatujte si, když o tom budete přemýšlet, že máme obrazovku s třemi bitplanes, tudíž máme jen osm barev.

PROGRAM

Značení rezistorů

Breina Breyl

V elektrotechnice se již před mnoha lety přešlo na značení rezistorů barevným kódem. Má to výhodu v tom, že v každé pozici rezistoru na desce plošného spoje je možné určit velikost odporu, aniž by bylo nutné rezistor otáčet a hledat vytištěnou značku. Háček je však v tom, že pro určení odporu je nutné znát kódové značení rezistorů. A právě zde vám pomůže náš program. Ze zadaného odporu a tolerance (třídy přesnosti) určí jednotlivé barevné proužky a nebo naopak, ze zadaných barev určí odpor a třídu přesnosti. Proužky jsou zobrazeny v barvách a navíc opatřeny popisem, aby byl program použitelný i na nebarevných monitorech.

Program je napsán v jazyce Amiga-Basic a pokud by jej chtěl někdo přepsat pro jiný počítač, pak CHR\$(8) v řádku 620 a 695 značí klávesu Backspace a CHR\$(129), CHR\$(130), CHR\$(131) v řádcích 545 až 555 funkční klávesy F1 až F3 a CHR\$(177) znak plus-mínus.

Příklad použití programu. Chcete zjistit barevné označení rezistoru o hodnotě odporu 56 kohm v třídě přesnosti $\pm 5\%$. Po spuštění programu stisknete funkční klávesu F1, zadejte hodnotu odporu v ohmech, tzn. "56 000" (nebo "56e3") a odešlete klávesou RETURN (nebo Enter). Pak se program ptá na toleranci. Zadejte "5" a opět odešlete s RETURN. Na obrazovce se objeví na maketě rezistoru proužky 1 až 4 v barvách zelená, modrá, oranžová a zlatá, což je barevný kód pro rezistor o hodnotě odporu 56 kohm $\pm 5\%$.

Chcete-li znát odpor rezistoru, který jste např. vypá-jeli z vašeho magnetofonu, stisknete funkční klávesu F2. Pak postupně zadejte čísla barev jednotlivých proužků podle vašeho rezistoru. Čísla barev vám pomůže určit tabulka, která je zobrazena v horní části obrazovky. Po zadání čísla barvy čtvrtého proužku vám počítač oznámí hodnotu odporu.

Pokud má rezistor jen tři barevné proužky (4. chybí), znamená to, že se jedná o rezistor třídy přesnosti 20 %, což se nevyznačuje. V tomto případě na žádost programu o zadání čísla barvy čtvrtého proužku stisknete pouze RETURN (bez zápisu).

```

10 REM Barevné značení rezistoru
10 REM (c) Breyl 27.11.1990
20 REM
30 REM ***** inicializace *****
40 CLS: CLEAR: DIM b$(15): maz$ = SPACE$(60)
44 lhx = 200: lhy = 102: pdx = 220: pdy = 128
50 SCREEN 2, 640, 250, 4, 2
60 WINDOW 2, "Značení rezistoru", 7, 2
70 PALETTE 0, 0, 0: REM cerna
80 PALETTE 1, 73, 53, 27: REM hneda
90 PALETTE 2, 1, 13, 13: REM ruda
100 PALETTE 3, 1, 53, 0: REM oranž
110 PALETTE 4, 93, 93, 13: REM zluta
120 PALETTE 5, 0, 8, 0: REM zelena
130 PALETTE 6, 47, 6, 1: REM modra
140 PALETTE 7, 1, 13, 93: REM fialova
150 PALETTE 8, 67, 67, 67: REM seda
160 PALETTE 9, 1, 1, 1: REM bila
170 PALETTE 10, 67, 73, 47: REM sribro
180 PALETTE 11, 8, 73, 33: REM zlata
190 PALETTE 12, 1, 87, 73: REM okr
195 PALETTE 13, 53, 93, 1: REM azurova
220 DATA cerna, hneda, cervena, oranž, zluta, zelena
230 DATA modra, fialova, seda, bila, sribro, zlata
240 FOR i = 0 TO 11: READ b$(i): NEXT i
250 b$(12) = " - ": COLOR 9, 0
260 LOCATE 2, 32: PRINT "Cisla barev"
265 LINE (146, 18) - (458, 78), 6, b
275 LINE (150, 20) - (454, 76), 13, bf
280 FOR i = 0 TO 5
282 LOCATE 4 + i, 28: COLOR 1, i: PRINT " ";
284 COLOR 1, i + 6: PRINT TAB(44) " "
290 LOCATE 4 + i, 23: COLOR 0, 13: PRINT i; " - ";
300 COLOR 0, i: IF i = 0 THEN COLOR 9, 0
310 PRINT " " b$(i) TAB(39): COLOR 0, 13
330 PRINT USING "##"; i + 6: PRINT " - ";
340 COLOR 0, i + 6: PRINT " b$(i + 6)
350 NEXT i
360 LINE (169, 100) - (435, 130), 6, b
365 LINE (172, 102) - (432, 128), 12, bf
370 LINE (50, 115) - (169, 115), 6
380 LINE (435, 115) - (550, 115), 6
400 GOSUB inic: COLOR 9, 0
500 LOCATE 12, 26: PRINT "1. 2. 3. 4."
510 LOCATE 28, 17: COLOR 0, 13
520 PRINT " F1 = BARVY ";
525 PRINT SPC(1) " F2 = ODPOR ";
530 PRINT SPC(1) " F3 = KONEC "
540 v$ = INKEY$: IF v$ = "" THEN 540
545 IF v$ = CHR$(129) THEN GOSUB inic: GOTO 1000
550 IF v$ = CHR$(130) THEN GOSUB inic: GOTO 590
555 IF v$ = CHR$(131) THEN konec
560 GOTO 540
585 REM ***** odpor z barev *****
590 FOR i = 0 TO 3

```

```

610 GOSUB maz:LOCATE 25,24:COLOR 9,0
620 PRINT"Cislo barvy ";i + 1;CHR$(8);". prouzku ";
630 INPUT x$
635 IF i < 3 AND x$ < "0" OR x$ > "9" THEN
BEEP:GOTO 610
640 p$(i + 1) = x$:p = VAL(x$):bv = p
650 IF p > 11 THEN BEEP:GOTO 610
652 IF i < 3 AND x$ = "" THEN BEEP:GOTO 610
655 IF i = 3 AND p < > 1 AND p < > 2 AND p < > 10
AND p < > 11 THEN
657 p = 12:p$(4) = ""
659 END IF
660 LOCATE 20,12 + i*15
670 COLOR 9,0:PRINT i + 1;CHR$(8);".";
680 COLOR 0,9:PRINT" " + b$(p) + " "
700 bv = p:GOSUB prouzky:COLOR 9,0
710 NEXT i
720 a = VAL(p$(1)):b = VAL(p$(2))
730 c = VAL(p$(3)):d = VAL(p$(4))
740 GOSUB maz:
750 LOCATE 24,22:COLOR 4,0:PRINT "HODNOTA
ODPORU = ";
760 IF c = 10 THEN PRINT (a*10 + b)*.01;"ohm";
770 IF c = 11 THEN PRINT (a*10 + b)*.1;"ohm";
780 IF c = 0 THEN PRINT (a*10) + b;"ohm";
790 IF c = 1 THEN PRINT ((a*10) + b)*10;"ohm";
800 IF c = 2 THEN PRINT ((a*10) + b)/10;"kohm";
810 IF c = 3 THEN PRINT (a*10) + b;"kohm";
820 IF c = 4 THEN PRINT ((a*10) + b)*10;"kohm";
830 IF c = 5 THEN PRINT ((a*10) + b)/10;"Mohm";
840 IF c = 6 THEN PRINT ((a*10) + b);"Mohm";
842 IF c > 6 AND c < 10 THEN
843 PRINT"Mimo rozsah !":GOTO 510
844 END IF
845 PRINT SPC(3)CHR$(177)SPC(1);
846 IF d = 1 THEN PRINT "1 %"
850 IF d = 2 THEN PRINT "2 %"
860 IF d = 11 THEN PRINT "5 %"
870 IF d = 10 THEN PRINT "10 %"
880 IF d = 0 THEN PRINT "20 %"
890 GOTO 510
950 REM ***** barvy z odporu *****
1000 COLOR 4,0:GOSUB maz:LOCATE 24,25
1010 INPUT"Hodnota odporu [ohm] ";re$
1020 re = VAL(re$)
1022 IF re < = 0 OR re > = 1E + 08 THEN BEEP:
GOTO 1000
1026 IF re > = 1E + 07 THEN re = re/10:k = 1 ELSE k = 0
1028 a$ = STR$(re):a$ = RIGHT$(a$,LEN(a$)-1)
1030 LOCATE 25,25:INPUT "Tolerance [%] ";tr$
1035 tr = VAL(tr$)
1040 IF tr = 0 THEN p = 12:tr = 20
1050 IF tr = 1 THEN p = 1:tr = 1
1060 IF tr > 1 AND tr < 5 THEN p = 2:tr = 2
1070 IF tr > = 5 AND tr < 10 THEN p = 11:tr = 5

1080 IF tr > = 10 AND tr < 20 THEN p = 10:tr = 10
1090 IF tr > = 20 THEN p = 12:tr = 20
1100 LOCATE 20,57:COLOR 9,0:PRINT"4.";
1105 COLOR 0,9:PRINT" " + b$(p) + " "
1110 bv = p:i = 3:GOSUB prouzky:COLOR 4,0
1120 tr$ = STR$(tr):LOCATE 25,47:PRINT tr$ + " "
1130 IF LEFT$(a$,1) = "." THEN a$ = "0" + a$
1140 IF LEN(a$) = 1 THEN a$ = a$ + ".0"
1150 IF VAL(a$) < 1 AND LEN(a$) < 4 THEN
a$ = a$ + "0"
1160 IF LEFT$(a$,2) = "0." THEN
a$ = MID$(a$,3,LEN(a$)) + "st"
1170 IF MID$(a$,2,1) = "." THEN
a$ = LEFT$(a$,1) + MID$(a$,3,1) + "zl"
1180 a = VAL(LEFT$(a$,1)):b = VAL(MID$(a$,2,1))
1190 c$ = MID$(a$,3,LEN(a$)):COLOR 0,9
1200 i = 0:p = a:bv = a:LOCATE 20,12:COLOR
9,0:PRINT "1.";
1210 COLOR 0,9:PRINT" " + b$(p) + " ":GOSUB prouzky
1220 i = 1:p = b:bv = b:LOCATE 20,27:COLOR
9,0:PRINT"2.";
1225 COLOR 0,9:PRINT" " + b$(p) + " ":GOSUB prouzky
1230 IF c$ = "" THEN p = 0
1240 IF c$ = "0" THEN p = 1
1250 IF c$ = "00" THEN p = 2
1260 IF c$ = "000" THEN p = 3
1270 IF c$ = "0000" THEN p = 4
1280 IF c$ = "00000" THEN p = 5 + k
1290 IF c$ = "000000" THEN p = 6 + k
1300 IF c$ = "st" THEN p = 10
1310 IF c$ = "zl" THEN p = 11
1320 LOCATE 20,42:COLOR 9,0:PRINT"3.";
1325 COLOR 0,9:PRINT" " + b$(p) + " "
1330 i = 2:bv = p:GOSUB prouzky
1340 GOTO 890
1999 REM ***** podprogramy *****
2000 konec:
2010 WINDOW CLOSE 2:SCREEN CLOSE 2:END
2999 '
3000 inic:
3010 bv = 13:COLOR 9,0
3020 FOR i = 0 TO 3:GOSUB prouzky:NEXT i
3030 LOCATE 20,12:PRINT maz$
3040 RETURN
3999 '
4000 prouzky:
4010 LINE (lhx + i*45,lhy)-(pdx + i*45,pty),bv,bf
4020 RETURN
4999 '
5000 maz:
5010 FOR j = 0 TO 1
5020 LOCATE 24 + j,18
5030 PRINT maz$
5040 NEXT j
5050 RETURN

```


GURU!?

JE TO OPRAVDU TAK TAJEMNÉ

Khaled Husseini

Na světě není uživatele Amigy, který by v životě neviděl systémový requester s následujícím nápisem:

Software error - task held
(chyba softwaru - úloha pozastavena)
Finish ALL disk activity
(ukončete všechny diskové operace)
Select CANCEL to reset/debug
(zvolte CANCEL pro reset/debug)

Když jste to uviděli poprvé, tak jste si nejspíše mysleli, že jde o nějaký špatný vtíp. Chvilku jste se na to dívali (možná hledali dokonce i v anglicko-českém slovníku význam slov) a potom jste se odhodlali k stisknutí levého tlačítka myši na volbě CANCEL. A byli jste v šoku. Kde je Workbench? Co se ti stalo, ty moje Amigo? Před Vámi se objevila černá obrazovka s malými oranžovými písmeny v oranžovém orámování. A četli jste:

Software Failure.
(selhání softwaru)
Press left mouse button to continue.
(stiskněte levé tlačítko pro pokračování)
Guru Meditation #00000004.000C0401

V této chvíli mnozí z Vás pochopili, že s Vaším programem je amen. Chytali jste se za hlavu a zoufali jste, co bude s Vašimi daty, s prací několika týdnů, ne-li měsíců. Bohužel Vám mohu jen říci, že vše je nenávratně pryč. Leda, ano, Vy jste byl na tolik prozíravý, že jste si vždy během své práce nahrával všechno postupně na disk. Gratuluji. Jenom Vy nebudete tak zoufalý jako ti z Vás, kteří bezmezně důvěřovali svému miláčkovvi. V tomto článku bych vás chtěl seznámit s pozadím těchto hlášek. Jak jste si už určitě všimli skládá se chybové hlášení vždy ze dvou 32-bitových čísel oddělených tečkou. První 32-bitové číslo popisuje chybu, která nastala, a druhé 32-bitové číslo uvádí adresu úlohy v paměti RAM, která chybu způsobila. Zkoumáním prvního čísla můžeme zjistit chybu a později se pokusit odstranit příčiny, aby se to už neopakovalo (např. chybějící knihovny v adresáři LIBS:).

Ted' uvedu významy prvního čísla (popis zleva):

* prvním dvěma ciframi říkáme Subsystem number. Označuje nám, který z modulů operačního systému zaregistroval chybu. První z těchto dvou cifer může být zvláštním způsobem kódována. Je-li nejvýznamnější bit této cifry nastaven, jedná se tzv. dead-end alert. Jinými slovy

nemůže už systém žádným způsobem zachránit vzniklou situaci (po praktické stránce nejde zachránit ani normální alert, tudíž se jedná vlastně jen o oznámení, že už je více než zle).

Příklad: 02 znamená, že došlo k chybě v *graphics.library* a jedná se o normální (ne dead-end) alert. 82 znamená, že došlo též k chybě v *graphics.library*, ale jedná se o dead-end alert.

Zde jsou moduly operačního systému a jejich čísla:

Exec Library	01
Graphics Library	02
Layers Library	03
Intuition Library	04
Math Library	05
GList Library	06
DOS Library	07
RAM Library	08
Icon Library	09
Audio Device	10
Console Device	11
GamePort Device	12
Keyboard Device	13
TrackDisk Device	14
Timer Device	15
CIA Resource	20
Disk Resou rce	21
Misc Resource	22
Bootstrap	30
Workbench	31

* další dvě cifry nazýváme General Error Code. Popisují obecný typ vzniklé chyby. Většinou tyto cifry bývají rovny nule (00), co znamená, že žádný obecný typ chyby nevznikl. S následujícími typy chyb se můžete setkat:

Insufficient memory	01 (nedostatek paměti)
MakeLibrary error	02
OpenLibrary error	03
OpenDevice error	04
OpenResource error	05
I/O error	06

* poslední čtyři cifry popisují podrobně vzniklou chybu v daném modulu. Zde uvedu pro každý modul nejběžnější chybová čísla. Uvádím je kompletní ve 32-bitové formě, protože mnohé z nich se nevyskytují s žádnou jinou kombinací prvních čtyř cifer.

Exec Library

ExcptVect	81000001	kontrolní součet CPU trap vektoru
BaseChkSum	81000002	kontrolní součet ExecBase
LibChkSum	81000003	kontrolní součet knihovny
LibMem	81000004	chybí paměť pro vytvoření knihovny
MemCorrupt	81000005	pokazený seznam paměti
IntrMem	81000006	chybí paměť pro programy obsluhující přerušování

SemCorrupt	81000008	semafor je v i legálním stavu
FreeTwice	81000009	uvolnění paměti, která byla již uvolněna

Graphics Library

GfxNoMem	82010000	graphics došla paměť
CopDisplay	82010001	copper display list nemá paměť
CopInstr	82010002	copper instruction list nemá paměť
CopListOver	82000003	copper list je příliš dlouhý
CopIListOver	82000004	copper intermediate list je příliš dlouhý
CopListHead	82010005	copper list head nemá paměť
LongFrame	82010006	long frame nemá paměť
ShortFrame	82010007	short frame nemá paměť
FloodFill	82010008	flood fill nemá paměť
TextTmpRas	02010009	text, nedostatek paměti pro TmpRas
BlitBitMap	8201000A	BlitBitMap nemá paměť
MakeVPort	82010030	MakeVPort nemá paměť

Layers Library

LayersNoMem	83010000	layers došla paměť
--------------------	-----------------	--------------------

Intuition Library

GadgetType	84000001	neznámý typ gadgetu
CreatePort	84010002	create port nemá paměť
ItemAlloc	84010003	item plane alloc nemá paměť
SubAlloc	84010004	sub alloc nemá paměť
PlaneAlloc	84010005	plane alloc nemá paměť
ItemBoxTop	84000006	item box top < RelZero
OpenScreen	84010007	open screen nemá paměť
OpenScrnRast	84010008	open screen rast alloc nemá paměť
SysScrnType	8400 0009	open sys screen neznámý typ
AddSWGadget	8401000A	add SW gadgets nemá paměť
OpenWindow	8401000B	open window nemá paměť
BadState	8400000C	Bad State Return při vstupu do Intuition
BadMessage	8400000D	Bad Message přijato
NoConsole	8400000F	IDCMP nemohl otevřít Console Device

DOS Library

StartMem	07010001	chybí paměť při startu
EndTask	07000002	EndTask neuspěl
QpktFail	07000003	Qpkt selhal
AsyncPkt	07000004	neočekávaný packet
FreeVect	07000005	Freevec selhal
DiskBlkSeq	07000006	disk block sequence chyba
BitMap	07000007	zničený Bitmap
KeyFree	07000008	klávesa je už uvolněna
BadChkSum	07000009	ilegální kontrolní součet

DiskError	0700000A	disková chyba
KeyRange	0700000B	klávesa mimo rozsah
BadOverlay	0700000C	špatný overlay

RamLib Library

BadSegList	08000001	žadné overlaye v seglists knihovně
-------------------	-----------------	------------------------------------

TrackDisk Device

TDCalibSeek	14000001	calibrate: seek chyba
TDDelay	14000002	delay: chyba při timer wait

Timer Device

TMBadReq	15000001	špatný request
TMBadSupply	15000002	zdroj nedodává tlky

Disk Resource

DRHasDisk	21000001	get unit: už má disk
DRIntNoAct	21000002	přerušeni: žádná aktivní jednotka

Bootstrap

BootError	30000001	bootovací kód vrátil chybu
------------------	-----------------	----------------------------

Všechno co bylo výše popsáno se týká chyb rozpoznávaných Amiga ROM Kernel. Existuje ale další kategorie chyb, které jsou vyvolávány přímo procesorem 68000. Těmto stavům říkáme *68000 processor traps*.

Jedná se o chyby typu dělení nulou, vykonání nepřipustné instrukce apod. Většinou Exec způsobí na tyto chyby podobnou hlášku, jako byla popsána výše, ale programy mohou řídit tyto stavy i samy. Zde bych chtěl popsat možné chyby.

Nejlevější část prvního 32-bitového čísla (šest prvních cifer) bude obsahovat velmi malou hodnotu. Poslední dvě cifry mohou nabývat následujících hodnot (tyto čísla jsou součástí procesoru a nejsou přidělována ROM Kernel jako ty výše popsána):

Bus Error	02	chyba sběrnice
Address Error	03	chyba adresace
Illegal Instruction	04	nepřipustná instrukce
Divide by Zero	05	dělení nulou
CHK instruction	06	instrukce CHK
TRAPV (Overflow)	07	přetečení TRAPV
Priviledge Violation	08	porušení privilegovaného stavu
Instruction Trace	09	instruckce trasování
Line A Emulation	0A	emulace Line A
Line F Emulation	0B	emulace Line F
TRAP 0 .. 15	20 .. 2F	

Doufám, že všichni čtenáři vydrželi až dokonce. Mým cílem bylo poskytnout programátorům a "šfóuralům" přehled možných chybových hlášení, který by jim mohl při jejich práci pomoci. V blízké budoucnosti bych se chtěl věnovat i příčinám těchto chyb z hlediska programátora, a jak správně programovat, aby programy byly co nejvíce odolné proti takovýmto hláškám. Chtěl bych taky předvést malý program, který sám obsluhuje chyby procesoru 68000. Tak nashle někdy příště a doufám, že Vám aspoň v něčem tento článek pomůže.

STAVBA EXTERNÍ DISKOVÉ JEDNOTKY

Zdeněk Daniček

Každý šťastný majitel Amigy po určité době zjistí, že pracovat pouze s jednou diskovou jednotkou je značně omezující. Nabízí se dvojí řešení:

1) Nákup kompletní externí diskové jednotky speciálně vyrobené pro Amigu. Cena takového kompletu se pohybuje v rozmezí 220-300,- DM.

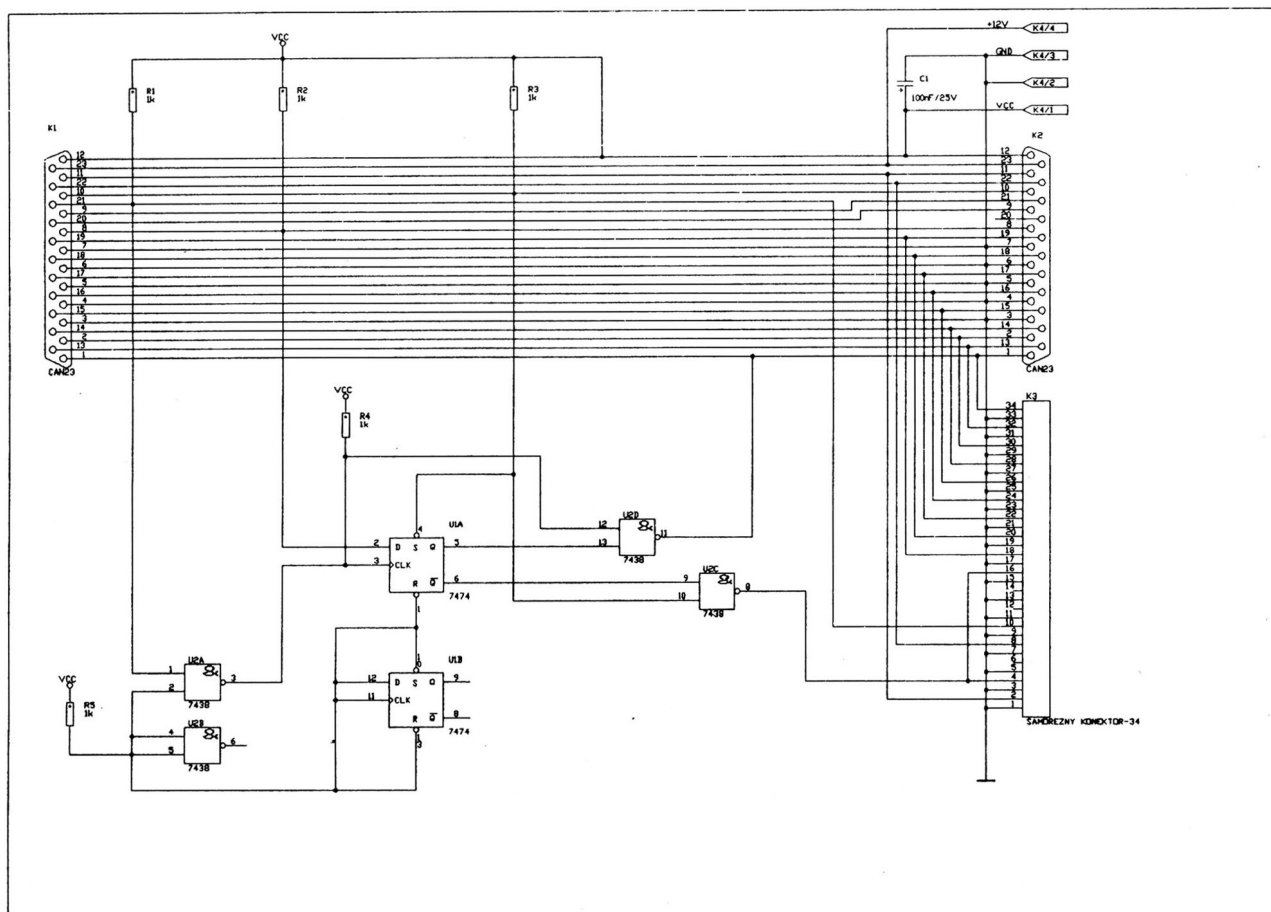
2) amatérská stavba externí diskové jednotky.

Pokud se rozhodnete pro amatérskou stavbu, tak se dá podstatně ušetřit a navíc získáte pocit, že jste udělali

něco navíc než jenom zastrčili konektor do příslušného místa.

K vlastní stavbě potřebujete následující součásti:

- disková jednotka 3.5" (NEC 1035, NEC 1036, Chinon FZ 357 nebo jakýkoliv jiný 3.5" dvoustranný)
- konektor 2 ks (CANON 23 - 1 ks SUB-D-Stiftleiste 1 ks SUB-D-Buchsenleiste)
- konektor 1 ks (pro plochý 34 žilový vodič typ Steckerleiste-Wannen 34 prořezávací)
- pouzdro 1 ks (pro CANON 23 typ Steckergehäuse für SUB-D-23)
- plochý 34 žil. vodič cca 150 mm dlouhý
- kabel 24 žil. kulatý typ H-ediflex-Computer-Kabel 24 cca 600 mm
- napájecí konektor pro příslušnou diskovou jednotku
- IC obvod SN 74LS74 1 ks
- SN 7438 1 ks
- rezistory 1 kOhm 5 ks
- kondenzátor 100 nF/25V 1 ks

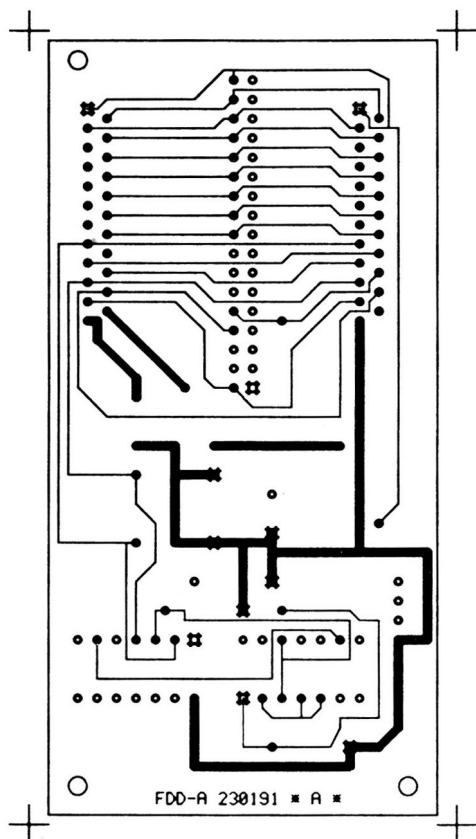


Obr. 1

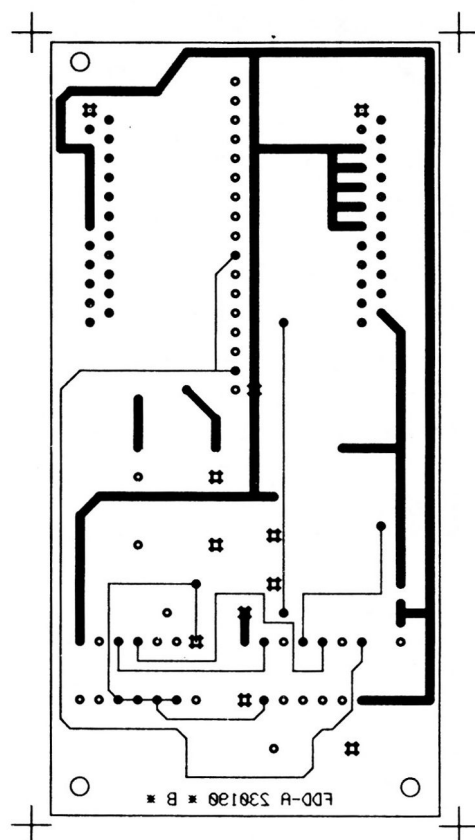
Konstrukce:

Disketová jednotka je navržena tak, aby vytvářela samostatný celek napájený z Amigy. Na obr.1 je znázorněno schema zapojení. Ze schematu je patrné, že konstrukce diskové jednotky pro Amigu vychází z podmínky vzájemné návaznosti připojovaných diskových jednotek na sebe. V prvním čísle našeho časopisu jsme podrobně popsali jednotlivé signály, proto je zde nebudeme znovu všechny rozebírat. Pro ty, kteří chtějí podrobněji znát jak probíhá připojení externí diskové jednotky, uvádím další podrobnosti. Jak je vidět z obrázku ukládá se signál na vodiči MTRX do Flip-flopu U1A když SEL 1 jde z 1 na 0. Jelikož U1A ukládá hodnotu na svém datovém vstupu s pozitivní hranou hodinového signálu,

musí být SEL 1 invertován. Toto provádí hradlo označené jako U2A. Z důvodu dostatečného výkonu na MTR je výstup Flip-flop posílen dalším hradlem U2D. Hradlo U2C nemá nic společného s řízením motoru. Slouží k identifikačnímu módu. Vždy, když se motor vypne a vodič SEL 1 je aktivní, tedy v 0, uvede hradlo U2C vodič RDY na 0. Tím pak rozezná Vaše Amiga, že druhý disk je 3.5" standardní s číslem DF1. Z důvodů snížení možnosti vzniku případných chyb, vzniklých při zapojování, jsme navrhli plošný spoj, který nám značně zjednoduší zapojení této disketové jednotky. Vzhledem k požadované velikosti jsme museli navrhnout dvoustrannou verzi. Na obr.2 je uvedena strana A a na obr.3 strana B.



Obr. 2



Obr. 3

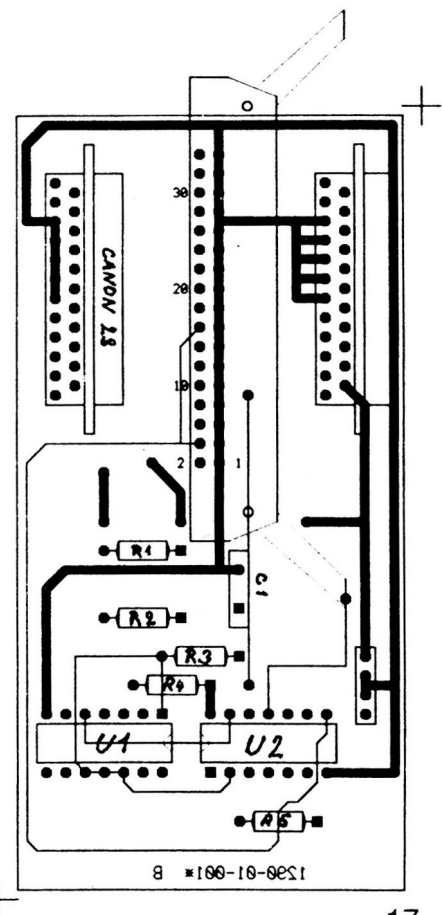
Pro snadné osazení je na obr.4 zobrazeno osazení součástkami. Seznam použitých součástek umístěných na desce plošného spoje:

Rezistory (TR 191 nebo TR 151)	R1-R5	1 kOhm
Kondenzátor	C1	100 nF
IC obvody	U1A-1B	74LS74 (7474)
	U2A-2D	74LS38(7438)

Zásuvka konektoru CANON 23 PIN Připojovací kabel se zástrčkou konektoru CANON 23 PIN Připojovací plochý vodič se zástrčkou do diskové jednotky 34 PIN Speciální napájecí vodič se zástrčkou do napájecí části diskové jednotky.

Závěr:

Stavba disketové jednotky pro Amigu je velmi jednoduchá a může se do ní pustit i méně zkušený amatér. Počítáme i s takovými, kteří rádi využijí případné služby a to ve formě částečné stavebnice, úplné stavebnice nebo kompletně hotové externí disketové jednotky. Pokud máte zájem, napište nám do redakce o jakou formu rozpracování byste se zajímali. Podle počtu zájemců Vám sdělíme podmínky a cenu dodávky.



Obr. 4

HRVY

Tomáš Adamec (TAD)

SUPER CARS

■ Gremlin Graphics 1990

TEST 2/91		
Grafika	83%	Celkem 76%
Zvuk	71%	
Idea	58%	
Přitažlivost	90%	

SUPER CARS potěší všechny milovníky rychlých a silných vozů. Na devíti okruzích proti sobě soupeří jezdcí v extratřídě sportovních automobilů. Máte-li odvalu (a peníze),

vstupte do jejich světa. Základem všeho je závodní automobil, který musíte zakoupit ve speciálním obchodě. Koupit můžete vozy Taraco Neoroder Turbo, Vaug Interceptor Turbo a Retron Parsec Turbo, každý ve dvou provedeních, podle kterého se liší i cena (informujte se o nich kliknutím na ten který automobil na základní obrazovce). A jelikož obchodník je také jenom člověk, zkuste s ním smlouvat. Některé repliky potěší každého angličtináře se smyslem pro humor. Nyní máte vůz, na který vám stačily finance. V garáži můžete dokupovat různá užitečná zařízení a můžete zde svůj vůz opravovat (stupeň poškození uveden u ENG (motor), BOD (karoserie), FUE (palivo) a

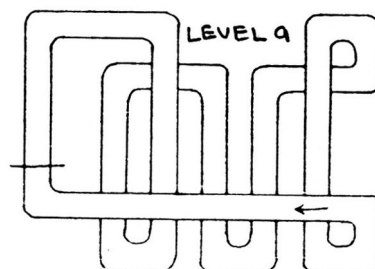
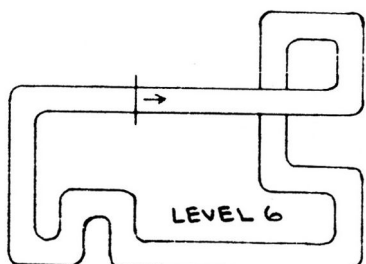
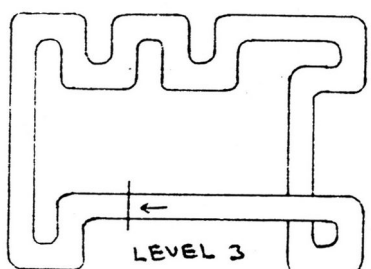
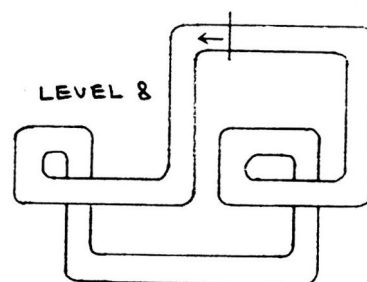
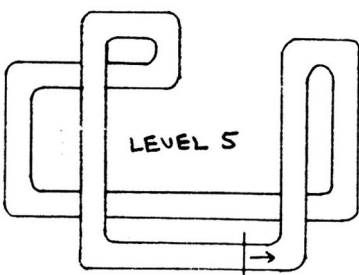
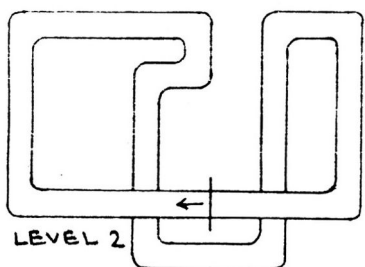
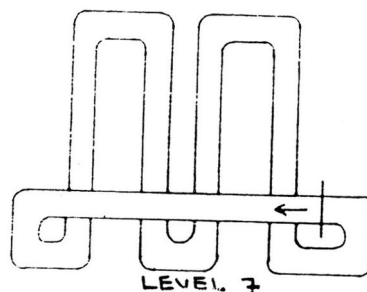
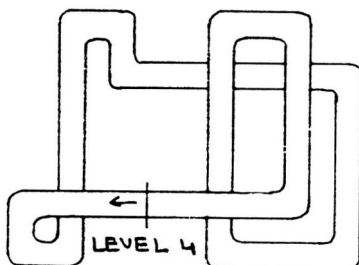
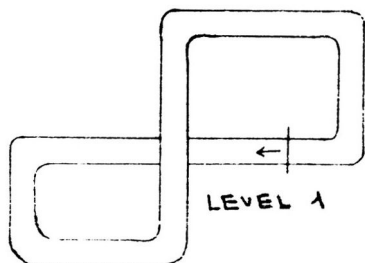
TYR (pneumatiky). Poté se vydejte na některý z devíti okruhů (musíte je projet všechny, i když nezáleží na pořadí, v jakém je pojedete). Pokud okruh dojedete alespoň na třetím místě, postupujete dále. Zdá-li se vám, že již máte dost peněz, pokuste se o koupi lepšího vozu. A ještě k ovládání: Střelou se rozjedete, zatáčíte doleva a doprava (uvědomte si, že sedíte v autě). Směrem dopředu a dozadu vypouštíte bomby (máte-li je), kterými můžete likvidovat své nepřátele (nesportovní, ale účinné).

NÁZOR MARTIN:

Hra SUPER CARS je trochu zvláštní: Na první pohled vypadala jako nic moc, potom se mi na krátkou dobu zalíbila, avšak stejně rychle mé sympatie opět opadly. Jinak mohu říct, že SUPER CARS jsou dobře propracovanou hrou, vynikající především v oblasti animace.

NÁZOR TOMÁŠ:

Pěkný design, grafika se smyslem pro detail (všimněte si stínování). Při hře výborné zvukové efekty, kombinované s neméně dobrou hudbou. Řízení je u SUPER CARS skutečným zážitkem, pohyb je velmi dynamický a věrohodný. Výborné!



Existuje mnoho typů HUDů - mění se totiž (odlišnosti jsou malé, ale přece) s každým nastaveným typem zbraně. Povězme si nyní o přístrojích, které jsou stále stejné, a potom o odchylkách jednotlivých HUDů.

1) AOA Indexer určuje "úhel útoku", naklonění letadla ve vztahu k rychlosti. Svítí-li horní šipka, letadlo je nakloněno v příliš velkém úhlu a letí pomalu. Svítí-li dolní šipka, letoun má malý úhel náklonu a příliš velkou rychlost. Svítí-li střední kolečko, vše je O.K. K AOA Indexeru se váže i AOA Indikátor, který se nalézá na přední desce a udává přesné údaje o náklonu letounu.

2) NWSS/LGSI panel obsahuje kontrolky, týkající se podvozku: RDY - značí, že celý podvozek je v pořádku a připraven k použití (vytažen) NWS - oznamuje, že přední směrové kolo podvozku je vytaženo a v pořádku - DISC - oznamuje, že letadlo dosáhlo rychlosti, potřebné k odpoutání se od země. Podvozek vytahujete/zatahujete klávesou "G". Rychlost letu s vytaženým podvozkem nesmí přesáhnout 300 uzlů, jinak dojde k jeho poškození. U vyšších úrovní pak dojde ke vzniku smrtelného ocasního víru. DISC oznamuje, že nastala vhodná chvíle k zatažení podvozku.

3) Rychloměr udává aktuální rychlost vašeho stroje v desítkách uzlů.

4) MACH indikátor udává aktuální rychlost jako procentuální vyjádření rychlosti zvuku (Mach 1).

5) Kompas HUDu udává směr letu letadla v desítkách stupňů. V módu zbraní vzduch-vzduch umístěn v dolní části HUDu, jinak vždy nahoře.

6) Indikátor G udává přetížení v kabině letadla. Od hodnoty major může dojít ke stavu "blackout" - odkrvení při G8, či ke stavu "redout" - překrvení při záporném G pod -2,5.

7) Zobrazuje nejvyšší dosažené přetížení. Nulujete klávesou "X".

(Pokračování příště)

NOVINKY + ZAJÍMAVOSTI

* Celý herní svět s nedočkavostí očekával příchod POPULOUS II. Ke konci roku 1990 se fandové POPULOUS (tedy i já) konečně dočkali. Nová hra od firmy Bullfrog se jmenuje POWERMONGER. Proti sobě bojují nikoliv dva, nýbrž až sedm bohů (sedm národů). Samotná hra je výborně graficky zpracována, strategie má být oproti POPULOUS mnohem složitější, POWERMONGER tedy bude skutečně hrou pro náročné.

* Konečně, konečně byla vytvořena amiga-verze hry CHAOS STRIKES BACK. Jedná se o pokračování v současné době našim časopisem popisovaného DUNGEON MASTERa. Vstup CHAOS STRIKES BACK na trh spadá do stejné doby jako POWERMONGER, tedy čtvrté čtvrtletí roku 1990. CHAOS STRIKES BACK již přes rok vévodí všem herním žebříčkům Atari ST a zřejmě tomu tak bude v blízké době i u Amigy.

* Ke konci roku 1990 vstoupily na trh hned dvě hry firmy Lucasfilm Games. Jedná se o třetí historický simulátor od Lucasfilm Games, THE SECRET WAFFEN OF LUFTWAFFE. Tou druhou hrou je další adventura v oblíbeném systému typu ZAK McKracken či INDIANA JONES 3 a jmenuje se THE SECRET OF MONKEY ISLAND. Děj vás zavede mezi piráty, tedy do světa dobrodružství a romantiky. Graficky se jedná ještě o větší zázrak, než tolik vychvalovaný LOOM.

* Pro zajímavost uveřejnil časopis Amiga-magazín M&T v čísle 12/90 soupis jím nejlépe hodnocených her roku 1990 v osmi kategoriích:

Action: Final Countdown, Flimbo's Quest, Flood, Projectyle, Lotus Esprit Turbo Challenge

Adventure: It came from the Desert,

Loom, Tomi & the Ghost

Logic: Atomix, Tower of Babel, Turn It

Sport: TV Sports Basketball, The Games: Summer Edition

Arcade: Paradroid '90, Pipe Mania, Rainbow Islands

Rolle-adventure: Cadaver, Champions of Krynn, Sword of Twilight

Simulation: 688 Attack Sub, Battle of Britain, Falcon Mission #2, Indianapolis 500

Strategy: Pirates!, Rings of Medusa Tyto údaje jsou nesporně zajímavé, avšak směrodatným bude teprve hlasování čtenářů samotných.

* Známa německá firma Data Becker věnovala ve svém bookware místo i hrám. Již dříve vyšly knihy DIE LARRY STORY a KING'S QUEST SAGA. Novinkou je třetí kniha zaměřená na adventure fy Sierra, DIE SPACE QUEST STORY. V této knize jsou popsány příběhy Rogera Wilca tak, jak je prožíval (a s ním mnoho adventuristů) ve hrách SPACE QUEST I, II a III. Kniha stojí stejně jako DIE LARRY STORY a KING'S QUEST SAGA, tedy 19,80DM.

* Hrou prosince '90 většiny herních magazínů se stal CADAVER od firmy Image Works. Jedná se o hru stylu arcade/adventure, která svým zpracováním (tzv. "rohovka") připomene všem bývalým Spectristům výborný KNIGHT'S LORE a spol. fy Ultimate.

* V posledním čtvrtletí roku 1990 vstoupila na trh novinka od fy Broderbund Software, akční adventura PRINCE OF PERSIA. Hrdinou je odvážný mladík, který se musí pokusit o záchranu své milé (perské princezny) před zlým Jaffarem. Tato hra je na dobré cestě stát se hrou s nejlepší animací za rok 1990.

LEISURE SUIT LARRY

IN THE LAND OF LOUNGE LIZARDS

■ Sierra On-Line 1986

TEST 2/91		
Grafika	35%	Celkem 51%
Zvuk	12%	
Idea	77%	
Přitažlivost	80%	

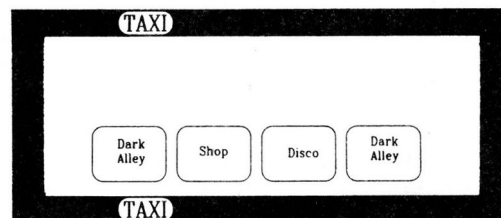
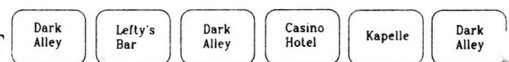
Už když jsem vám v minulém čísle ve článku o firmě Sierra On-Line sliboval, že brzy nějakou hru od této firmy popíši, nedovedl jsem si představit, že by tou první hrou mohla být nějaká jiná než LARRY. Popis bude super stručný a to ze dvou důvodů: 1) DUNGEON MASTER na vedlejší stránce se značně protáhl (mnoho věcí je vysvětlováno poprvé), zbylo tedy málo místa a 2) LARRY je velmi rozšířený a známý a mnoho dobrodruhů jej již dohrálo. Návod vyžaduje určitou znalost angličtiny a zapojení vlastního úsudku. Počáteční přístupové otázky odrušte kombinací Alt + X, jsou natolik americké, že na ně spatra neodpoví ani věkovitý profesor angličtiny.

Kasino aneb jak vydělat peníze: Můžete hrát blackjack (F4-začátek hry, F6-další karta, F8-stát) či zkusit štěstí na forbesu (F4-sníží částku, F6-zvýší částku, F8-pohne pákou). Nejvyšší sázka je \$20, maximálně můžete vyhrát \$250.

Před barem	- open door
V baru	- sit down
	- order whisky
	- stand up
Před toaletou	- give whisky
	- take rose
	- open door
Toaleta	- sit
	- get up
	- read wall (4x)
	- look at sink

Před barem	- take ring
	- taxi
V taxíku	- enter
	- shop
	- pay
	- get out
V obchodě	- take magazine
	- take wine
	- buy prophylactic
Před obchodem	- give wine to man
	- look at phone
	- use phone (555-6969)
Taxi	- bar
V baru	- knock on door
	- heslo "Ken sent me"
Za dveřmi	- use remote control
	- switch channel (6x)
Pokoj nad barem	- undress
	- use lubber
	- fuck
	- remove lubber
	- take candy
	- open window
	- climb through window
Kontejner	- take hammer
	- climb out
Taxi	- casino
V kasinu	- bud'to USE SLOT MACHINE, nebo PLAY BLACKJACK
Loby	- look into trash
	- take card
	- disco
Taxi	- show card
Před disco	- sit down
V budově	- look at girl
	- talk to girl
	- ask her to dance
	- sit down
	- talk to girl
	- give her candy
	- give her rose
	- give her ring
	- give her money
	- chapel
Taxi	- marry girl
Kaple	- buy apple (muž se
Před kasinem	

Kasino	objevuje náhodně)
Výtah	- opět vydělat peníze
	- four4. patro
	- knock door
Pokoj	- turn radio on
	- talk to girl
Taxi	- shop
Před obchodem	- answer phone
	- use phone 555-803
	- order wine
Taxi	- do kasina
Pokoj 4. patro	- lie down on bed
	- cut rope with knife
	- take ropeKasino
	- vydělat peníze
Taxi	- bar
Pokoj nad barem	- climb window
	- tie rope round waist
	- fasten rope to railing
	- climb over railing
	- smash window with hammer
	- take pills
	- climb back
	- untie rope
Taxi	- casino
Kasino 8.patro	- look at girl
	- talk to girl (několikrát)
	- give her pills
	- look at desk
	- push button
Skříň	- take doll
	- blow up doll
	- use doll
Balkón	- jump into tub
	- look at girl
	- talk to girl
	- give her apple



NÁZOR TOMÁŠ:

LEISURE SUIT LARRY je evergreenem v pravém slova smyslu. I když není tak kombinovaný a nenabízí takové akce jako novější adventure, je třeba ocenit neobyčejnou chytrost a vkus, kterým se tento první díl úspěšné řady může pochlubit. Nedokonalost provedení plně nahrazuje dokonalost obsahu.

DUNGEON MASTER

2. podlaží

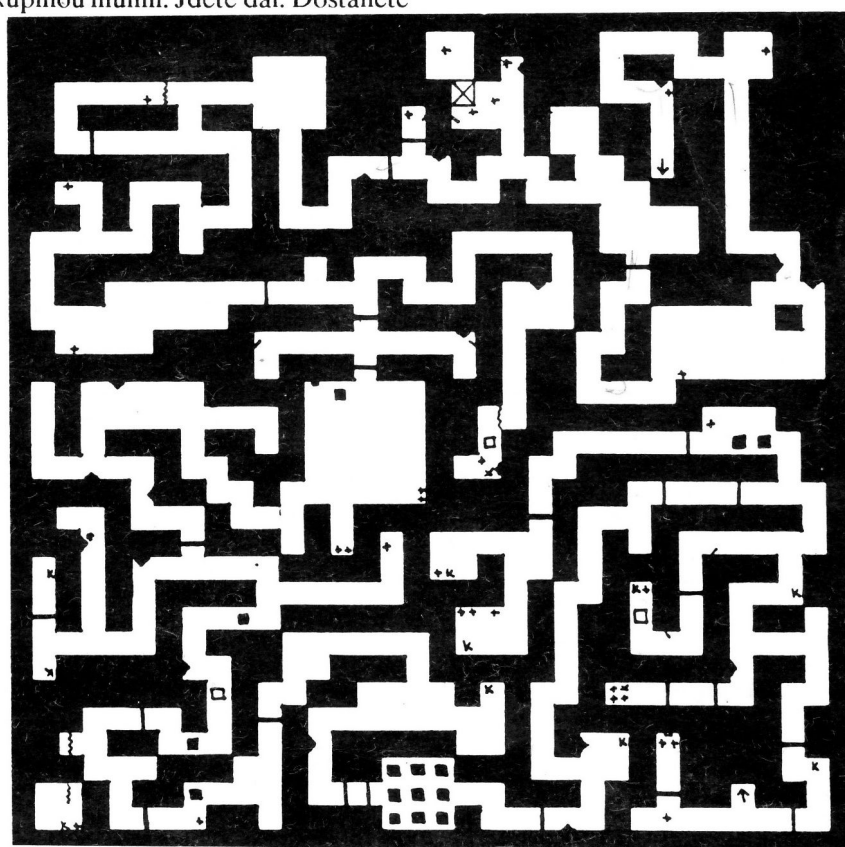
Nacházíme se tedy v druhém podlaží. Směr vaší cesty je jasný - musíte jít stále níž a níž, až se ve spodních sklepeních utkáte s Lordem Chaosem - míříte tedy ke schodům do třetího podlaží. Ihned jděte doleva. Za zavřenou mříží vidíte prvního nepřítele-mumii. Otevřete mříž a prodělejte svůj první boj. Když váš bojovník udeří, na místě ikon rukou se na chvíli objeví "počet bodů poškození", tedy míra zdraví, kterou váš útok potvoře ubral. Pokud se té legrační mumie bojíte, prozradím vám již zde základní zápasnický trik, který vám umožní v dalších úrovních ničit i takové nepřítele, kteří vás jinak roztrhají na kusy. Jakmile mumie vejde pod otevřené dveře, spusťte je na ni. Mříž do ní začne bušit, mezitím co vy můžete potvora zpracovávat svými zbraněmi. Je to velmi účinné. Po zabití mumie vkročte do chodbičky. Najdete tam jídlo a pergamen. Co to říká? Že malé detaily mohou přinést velké objevy? Stiskněte spínač ve stěně a vezměte meč!

Pokračujte v cestě. Až najdete první klíč, vezměte jej do ruky a vstrčte do klíčové dírk. Dveře se otevrou. Nyní se dostanete na první rozcestí - musíte nejprve doprava a získat dva klíče (můžete získat pouze dva), kterými otevřete dveře v chodbě nalevo. Při hledání jednoho z klíčů poprvé narazíte na pákové spínače. Jedním otevřete dveře, druhým uzavřete propadlo. Za otevřenou dvojicí dveří v levé části chodby pak vybavte své ninji. Nyní se s nově nalezeným klíčem odeberte do slepé chodby, ve které na vás (na rozvětveném konci) čekají dva klíče a další předměty. Se dvěma šedivými klíči postupujte dále. Během této cesty narazíte na dalšího nepřítele - "zpívající květinu". Nebojte se jí, je vítaným zdrojem potravy. Jděte dál, v místnosti s kašnou vezměte klíč. Pak musíte přejít přes pole spínačů a s

jejich pomocí otevřít dvojici dveří, které vám brání v postupu dál. Na to musíte přijít sami, je to snadné. Pokračujte v cestě, sbírejte předměty a masakrujte nepřátele. Dorazíte ke dveřím, které jsou otevřeny pouze tehdy, stojíte-li na blízkém spínači. Položte tedy na spínač kámen, ležící vedle. Potkáte jednu mumii a už jste u teleportu. Projděte jím, vezměte klíč a opět se vraťte. V rozlehlé čtvercové místnosti najdete pergameny s novými kouzly. Jedná se o kouzla zdravotnická: VI + BRO (do lahve) léčí otrávení, VI (do lahve) léčí poranění (HEALTH). Hned si obě kouzla vyzkoušejte. Kněz, který má kouzlo provést, musí v pravé ruce držet lahev - ty zde najdete také. Pokračujte dále. Na další veliké křižovatce (nejprve musíte otevřít dvoje dveře) odbočte doprava. Dostanete se ke cloně, vypadající jako teleport. Za ní se nachází propadlo. Vložte do clony předmět a propadlo se uzavře. Vezměte zde umístěné předměty, vraťte se a pokračujte dále. Ve velké podlouhlé místnosti vás čeká první větší boj - se skupinou mumií. Jděte dál. Dostanete

se k průhledné stěně, za kterou vidíte bednu. Stěnu nelze projít. Z mapy vidíte, že ji můžete obejít. Zavřené dveře, které vám stojí v cestě, jsou nejslabšími dveřmi ve hře. Jednou do nich bouchněte mečem a je po problému. Truhlu vyberete tak, že ji umístíte někomu do pravé ruky. Tato truhla obsahuje dva magické štíty v lahvích. Pokud některý z vašich lidí lahev vypije, bude po určitou dobu částečně chráněn magií. Jděte dál. Kašně, která vám chce splnit jedno přání, dejte penízek - dveře se otevrou. Přijdete do místnosti se spínačem. V tajné místnosti najdete mrazicí box. Chytrá věčička. Použijete-li ji v boji (někdo ji musí mít v pravé ruce), znehybníte na chvíli své protivníky. Na konci 2. podlaží v bedně najdete kouzlo na otrávení nepřítele: DES + VEN - první bojové kouzlo pro kouzelníky! Nechte své lidi, ať si odpočinou, nají se a napijí (připravte je k sestupu do třetího podlaží) a uschovejte si pozici.

(Nashledanou příště.)



- | | | |
|------------|--------------------|----------|
| + předmět | ⊗ tajný průchod | ● spínač |
| } teleport | ↓ schody dolů | ● páka |
| κ klíč | ↑ schody nahoru | ▾ kašna |
| □ propadlo | ■ podlahový spínač | |