

PIERWSI PROJEKTANCI KOMPUTERA AMIGA I SYSTEMU AMIGA OS



Jay Miner (31 maja 1932 - 20 czerwca 1994) - twórca Amigi, przy modelu A1000




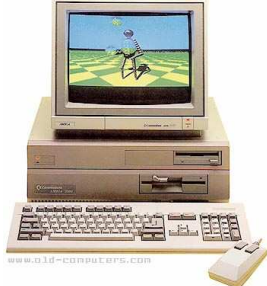




Amiga Team, lipiec 2007 – spotkanie po latach: od prawej, dół: **Glenn Keller** (HW: Paula chip), **Dave Needle** (HW: Agnus chip, płyta główna), od prawej, góra: **Carl Sassenrath** (SW: Exec kernel), **Dale Luck** (SW: grafika), **Sam Dicker** (SW: dźwięk), **RJ Mical** (SW: Intuition), **Bob Burns** (SW: drukarki i sterowniki I/O), **Hedley Davis** (HW: reprezentant Commodore)



Brak na zdjęciu: **Jay Miner** (HW: Team leader, Agnus chip), **David Dean** (HW: Denise chip), **Bob Pariseau** (SW: szef grupy), **Neil Katin** (SW: sterowniki urządzeń), **Jim Mackraz** (SW: grafika, sterowniki urządzeń), **Bart Whitebook** (SW: grafika)



KOMPUTERY AMIGA – LINIA CLASSIC



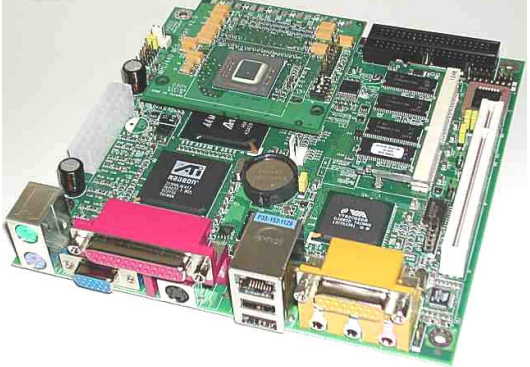
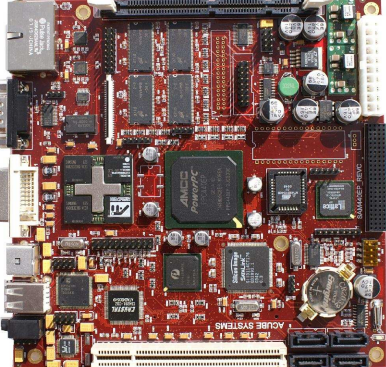
Amiga 1000	1985 – 1987 (Amiga Inc./Commodore Int.)	Procesor – Motorola 68000 (7.16 MHz – NTSC) Grafika – OCS (min. 320x256, od 2 do 32 kol. max. 640x512, od 2 do 16 kol. z 4096) Dźwięk – Paula (4x 8-bit, 28 kHz) I/O – Agnus, Denise, CIA, Gary ROM (Kickstart) – brak, z dyskietki RAM – 256 KB FDD 880 KB AmigaOS 1.0 – 1.3	
Amiga 2000	1986 – 1991 (Commodore Int.)	Procesor – Motorola 68000 (7.16 MHz – NTSC, 7.09 MHz – PAL) Grafika – OCS (min. 320x256, od 2 do 32 kol. max. 640x512, od 2 do 16 kol. z 4096) Dźwięk – Paula (4x 8-bit, 28 kHz) I/O – Agnus, Denise, CIA, Gary ROM (Kickstart) – 256 KB (1.2, 1.3) RAM – 512 KB FDD 880 KB HDD SCSI (opcja)	
Amiga 500	1987 – 1991 (Commodore Int.)	Procesor – Motorola 68000 (7.16 MHz – NTSC, 7.09 MHz – PAL) Grafika – OCS (min. 320x256, od 2 do 32 kol. max. 640x512, od 2 do 16 kol. z 4096) Dźwięk – Paula (4x 8-bit, 28 kHz) I/O – Agnus, Denise, CIA, Gary ROM (Kickstart) – 256 KB (1.2, 1.3) RAM – 512 KB FDD 880 KB	
Amiga 3000D	1990 - 1992 (Commodore Int.)	Procesor – Motorola 68030 (16 lub 25 MHz) + kooprocesor 68881 lub 68882 Grafika – ECS (min. 320x256, od 2 do 32 kol. max. 1280x256, od 2 do 4 kol. z 4096) Dźwięk – Paula (4x 8-bit, 28 kHz) I/O – Super Agnus, Super Denise, CIA, Gary ROM (Kickstart) – 2.0 RAM – 2 MB FDD 2x 880 KB HDD SCSI	
Amiga 3000T	1990 - 1992 (Commodore Int.)	Procesor – Motorola 68030 (16 lub 25 MHz) + kooprocesor 68881 lub 68882 Grafika – ECS (min. 320x256, od 2 do 32 kol. max. 1280x256, od 2 do 4 kol. z 4096) Dźwięk – Paula (4x 8-bit, 28 kHz) I/O – Super Agnus, Super Denise, CIA, Gary ROM (Kickstart) – 2.0 RAM – 2 MB FDD 2x 880 KB + 1x 5,25" HDD SCSI	
Amiga 500 Plus	1991 - 1992 (Commodore Int.)	Procesor – Motorola 68000 (7.16 MHz – NTSC, 7.09 MHz – PAL) Grafika – ECS (min. 320x256, od 2 do 32 kol. max. 1280x256, od 2 do 4 kol. z 4096) Dźwięk – Paula (4x 8-bit, 28 kHz) I/O – Super Agnus, Super Denise, CIA, Gary ROM (Kickstart) – 256 KB (2.04) RAM – 1 MB FDD 880 KB	

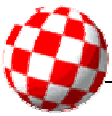


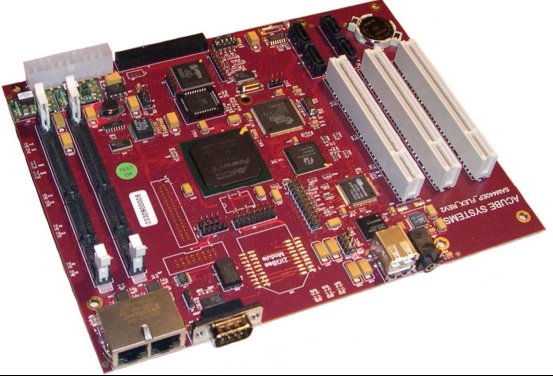


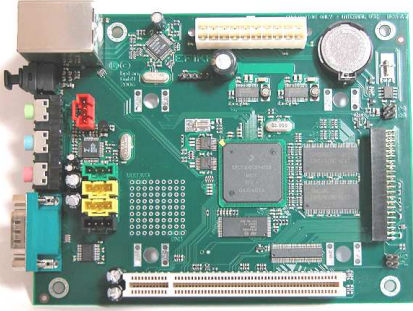

Amiga CDTV	1991 - 1993 (Commodore Int.)	Procesor – Motorola 68000 (7.16 MHz – NTSC, 7.09 MHz – PAL) Grafika – OCS (min. 320x256, od 2 do 32 kol. max. 640x512, od 2 do 16 kol. z 4096) Dźwięk – Paula (4x 8-bit, 28 kHz) I/O – Agnus, Denise, CIA, Gary ROM (Kickstart) – 256 KB (1.3) RAM – 1 MB FDD 880 KB HDD SCSI (opcja) CD-ROM, Wireless gamepad <i>Pierwszy na świecie komputer multimedialny</i>	
Amiga 600	1992 – 1993 (Commodore Int.)	Procesor – Motorola 68000 (7.16 MHz – NTSC, 7.09 MHz – PAL) Grafika – ECS (min. 320x256, od 2 do 32 kol. max. 1280x256, od 2 do 4 kol. z 4096) Dźwięk – Paula (4x 8-bit, 28 kHz) I/O – Super Agnus, Super Denise, CIA, Gary ROM (Kickstart) – 256 KB (2.05) RAM – 1 MB FDD 880 KB, PCMCIA HDD ATA 2,5" (opcja)	
Amiga 1200	1992 – 1996 (Commodore Int./Escorm)	Procesor – Motorola 68EC020 (14,28 MHz) Grafika – AGA (min. 320x200, od 2 do 256 kol. max. 1280x512, od 2 do 256 kol. z 16,8 mln) Dźwięk – Paula (4x 8-bit, 56 kHz) I/O – Alice, Lisa, Super Gary, Super Amber ROM (Kickstart) – 512 KB, (3.0, 3.1) RAM – 2 MB FDD 880 KB, PCMCIA HDD ATA 2,5" (opcja)	
Amiga 4000D	1992 – 1994 (Commodore Int.)	Procesor – Motorola 68EC030 lub 68040 (25 MHz) Grafika – AGA (min. 320x200, od 2 do 256 kol. max. 1280x512, od 2 do 256 kol. z 16,8 mln) Dźwięk – Paula (4x 8-bit, 56 kHz) I/O – Alice, Lisa, Super Gary, Super Amber, Super Ramsey ROM (Kickstart) – 512 KB (3.0) RAM – 2 MB FDD 1,76 MB HDD IN ZORRO	
Amiga CD32	1993 – 1994 (Commodore Int.)	Procesor – Motorola 68020 (14.3 MHz) Grafika – AGA (min. 320x200, od 2 do 256 kol. max. 1280x512, od 2 do 256 kol. z 16,8 mln) Dźwięk – Paula (4x 8-bit, 56 kHz) I/O – Alice, Lisa, Super Gary, Super Amber, Akiko ROM (Kickstart) – 512 KB (3.1) RAM – 2 MB CD-ROM <i>Konsola do gry</i>	
Amiga 4000T	1994 – 1997 (Escorm – Amiga Technologies)	Procesor – Motorola 68040 (25 MHz) lub 68060 (50 MHz) Grafika – AGA (min. 320x200, od 2 do 256 kol. max. 1280x512, od 2 do 256 kol. z 16,8 mln) Dźwięk – Paula (4x 8-bit, 56 kHz) I/O – Alice, Lisa, Super Gary, Super Amber, Super Ramsey ROM (Kickstart) – 512 KB (3.1) RAM – 6 MB FDD 1,76 MB HDD SCSI	



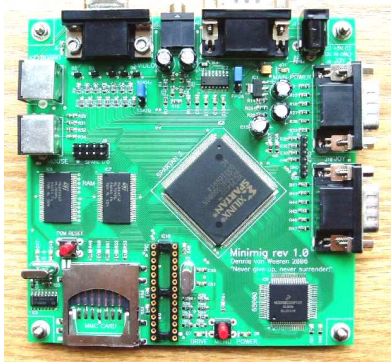
NOWE KOMPUTERY AMIGA I ICH KLONY

AmigaOne-SE (A1-SE)	2002 – 2004 (Eyeteck Group)	<p>Płyta ATX (305 x 244 mm) Procesor PowerPC G3 600 MHz (włutowany) RAM do 2 GB SDRAM DIMM (133 MHz) 2x IDE Northbridge - Articia S, Southbridge - VIA 82C686B 2 gniazda AGP x2, 4 gniazda PCI (3x 33 MHz, 1x 66 MHz) ethernet 100 Mb 3Com 920C 4x USB 1.1, 2x PS/2 2x port szeregowy, 1x port równoległy, IrDA</p>	
AmigaOne-XE (A1-XE)	2003 – 2004 (Eyeteck Group)	<p>Płyta ATX (305 x 244 mm) Moduł procesorowy PowerPC G3 800 MHz lub G4 800 MHz, 933 MHz, 1 GHz RAM do 2 GB SDRAM DIMM (133 MHz) 2x IDE Northbridge - Articia S, Southbridge - VIA 82C686B 2 gniazda AGP x2, 4 gniazda PCI (3x 33 MHz, 1x 66 MHz) ethernet 100 Mb 3Com 920C 4x USB 1.1, 2x PS/2 2x port szeregowy, 1x port równoległy, IrDA</p>	
Micro- AmigaOne (Micro-A1 C)	2004-2005 (Eyeteck Group)	<p>Standard płyty: Mini-ITX (170 x 170 mm) Procesor: PowerPC G3 750FX 800 MHz (na wyminnym module MegArray) Pamięć SO-DIMM PC133 do 2 GB pamięci Northbridge - Articia S, Southbridge - VIA 82C686B 2x port szeregowy, 1x port równoległy MIDI, 2x PS/2, 4x USB 1.1 Fast Ethernet (10/100 Mbit/s) na układzie 3Com 920C Dźwięk C-Media CMI8738 6-kanalowy 1x IDE 3,5", 1x IDE 2,5", 1x FDD, 1x PCI Grafika: ATI Radeon 7000 AGP z 32 MB pamięci - włutowany</p>	
Sam440ep	2008 – obecnie (A-Cube Systems)	<p>płyta format MiniITX (17x17 cm) procesor: AMCC 440ep (400, 533, 667 MHz) pamięć DDR DIMM 100 max 512 MB, 266 MHz zintegrowany kontroler PCI zintegrowany kontroler urządzeń pamięci Flash USB 1.1 i 2.0, zegar systemowy 2x ethernet 10/100, 4x port szeregowy kontroler USB 2.0 OHCI/EHCI NXP PCI dźwięk Cirrus Logic CS4281 i Realtek ALC655 Codec 4x Serial-ATA grafika: ATI RADEON Mobility M9 z 64 MB RAM mostek Pericom 8150B PCI 1x PCI - 32 bits, 33 MHz</p>	



Sam440ep Flex	2008 – obecnie (A-Cube System)	<p>plyta format FlexATX (21,6x17 cm) procesor: AMCC 440ep (667, 733, 800 MHz) pamięć DDR DIMM 100 max 1 GB, 266 MHz zintegrowany kontroler PCI zintegrowany kontroler urządzeń pamięci Flash USB 1.1 i 2.0, zegar systemowy 2x ethernet 10/100, 4x port szeregowy kontroler USB 2.0 OHCI/EHCI NXP PCI dźwięk Cirrus Logic CS4281 i Realtek ALC655 Codec 4x Serial-ATA mostek Pericom 8150B PCI 2x PCI – 32 bits, 33 MHz, 1x PCI – 32 bits, 66 MHz</p>	
Pegasos I G3	2001 – 2003 (bPlan/Genesi)	<p>plyta microATX (236 mm x 172 mm) mostek północny – ArticiaS procesor: IBM 750Cxe PowerPC G3 600 MHz 2x DIMM ECC PC133 max. 2 GB 1x AGP, 3x PCI, 4x IDE, FDD, USB 1.1 FireWire (IEEE 1394), IrDa, AC'97 port joysticka PC ethernet 10/100 Mbit port szeregowy port równoległy 2x gniazdo PS/2</p>	
Pegasos II G3 Pegasos II G4	2003 – 2006 (bPlan/Genesi)	<p>plyta microATX (236 mm x 172 mm) mostek północny – Marvell Discovery II procesor: IBM 750 Cxe PowerPC G3 600 MHz lub Freescale MPC7447 1.0 GHz (G4) 2x DIMM DDR PC2100 max. 8 GB 1x AGP (działający na szynie PCI-X), 3x PCI, 4x IDE, FDD, USB 1.1, FireWire (IEEE 1394), IrDa, AC'97 port joysticka PC 1x ethernet 10/100 Mbit, 1x ethernet 1 Gbit port szeregowy port równoległy 2x gniazdo PS/2</p>	
Efika	2005 – 2007 (bPlan/Genesi)	<p>plyta: nanoATX (153 mm x 118 mm) procesor: Freescale MPC5200B PowerPC SoC 400MHz (zintegr.) 128 MB DDR RAM (zintegr.) 1x PCI 2.2 + AGP ReiserCard HDD ATA 2,5" USB 1.1 AC'97 Ethernet 100/100 Mb RS232 IrDa</p>	
Mac Mini G4	2005.01 – 2005.09 (Apple)	<p>Procesor: PowerPC 7447a G4, 1.25, 1.33, 1.42, 1.5 GHz ROM: 256, 512 lub 1024 MB SDRAM DDR Grafika: ATI Radeon 9200, 32 lub 64 MB, 4x AGP Dźwięk: 2 kanały HDD 3,5", DVD Combo 2x USB 2.0, 1 x FireWire Wireless AirPort 802.11 b/g Ethernet 10/100 Mb, modem analogowy 56 kbps</p>	 www.apple.com



Minimig	2007 – obecnie (projekt - Dennis VanWeeren, produkcja – A-Cube)	sprzętowy emulator, uważany też za kłona Amigi 500 procesor: układ FPGA Spartan-3 Xilinx (emuluje RAM – 2 MB, procesor MC68SEC000, układy specjalizowane – Denise, Agnus, Paula, Gary i CIA oraz układ Amber do łączenia z VGA) slot kart pamięci MMC 2x PS/2 2x Joystick (DB9) 1x RS232 (DB9)	
Sam460	A-Cube	?	?
X1000	A-EON	?	?



CECHY

- wieloprocusowość (*multiprocessing*)
- wielozadaniowość (*multitasking*) z wyłączeniem
- olbrzymia szybkość reakcji (system czasu rzeczywistego – tzw. *real-time*)
- budowa modułowa - umożliwia prostą i bezproblemową instalację dodatkowych:
 - systemów plików (CDFS, FAT, HFS)
 - bibliotek
 - lokalizacji systemu i oprogramowania
 - powłok graficznych
 - datatypów do obsługi znacznej ilości formatów
- uproszczone łatanie funkcji systemowych przez systemy setpatch
- *deflcons* – dowolnie definiowane indywidualne ikony dla każdego pliku, a nie tylko po rozszerzeniu pliku
- system *datatypes* – szybkie rozszerzanie systemu o nowe formaty multimedialne
- dostęp do *CLI* (command line interpreter) opartym na AmigaShell
- dynamicznie tworzone wirtualne pulpity – *screen's*
- nie są wymagane rozszerzenia plików, pliki są rozpoznawane na podstawie liczby magicznej
- nazwy plików zachowują zapis małe/duże znaki, nie są jednak rozróżniane przez system
- opcjonalna pamięć wirtualna (od AOS 4.0) ze stronicowaniem pamięci
- podstawowa ochrona pamięci (od AOS 4.0) – brak ochrony pamięci dla uruchomionych procesów – kernel i programy działają na wspólnym obszarze, przy czym jedynie obszar zajmowany przez kernel staje się zabezpieczony przez zapisem
- ładowanie do pamięci tylko wymaganych sterowników i bibliotek, przy ich niewykorzystywaniu są usuwane z pamięci.

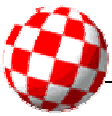
ARCHITEKTURA SYSTEMU

AmigaOS jest nowoczesnym systemem operacyjnym stworzonym przez firmy Amiga Inc i Commodore, który w roku premiery (1985 r.) wyprzedzał zastosowanymi rozwiązaniami, nowoczesnością, funkcjonalnością wszystkie inne znane systemy operacyjne. System składa się z trzech podstawowych elementów:

- jądra systemu (kernel) o nazwie **exec.library** – będącym w dzisiejszym rozumieniu mikrokerneliem
- **intuition** (obecnie zastąpiony przez **ReAction**) – biblioteki wykonawcze API oraz środowisko GUI - **Workbench**
- **AmigaDOS** – powłoka tekstowa systemu, bazująca na systemie Tripos firmy MetaComCo.

System od początku był 32-bitowy, napisany został dla procesora Motorola 68000. Obsługiwane procesory to: **MC68000**, **MC68010**, **MC68020**, **MC68030**, **MC68040**, **MC68060**. Systemy w wersjach 3.x obsługują również procesory PowerPC znane także jako PPC, dzięki podsystemom **WarpOS** albo **PowerUP**. System pracuje nadal na M68x00, istnieje jednak możliwość uruchamiania programów napisanych dla PPC. Konstrukcja i oprogramowanie kart procesorowych umożliwia jednoczesną pracę obu procesorów, przy czym PPC jest wykorzystywany jako koprocessor. W krótkim czasie od pojawienia się kart z układami PPC powstał projekt **MorphOS** – mający na celu przeniesienie systemu AmigaOS na architekturę PPC. Pracę programistów utrudniał brak dokumentacji – kody źródłowe systemu 3.0/3.1 stały się niedostępne krótko po upadku firmy Commodore. Istnieje także otwarta implementacja systemu AmigaOS pod nazwą **AROS** i jest dostępna między innymi na platformę x86 i PowerPC.

Obecne wersje AmigaOS 4.0 i 4.1 zostały przepisane całkowicie dla procesorów PowerPC i działają na komputerach **AmigaONE**, **microA1**, **Sam440ep**, **Pegasos II**, wkrótce **Sam460** i **X1000**. Oprogramowanie, które zostało napisane dla klasycznych Amig wyposażonych w procesory serii MC68x może być uruchamiane dzięki dwóm wbudowanym w system emulatorom: interpretowanemu, zapewniającemu wysoką zgodność z oryginalnymi układami sprzętowymi oraz JIT o nazwie Petunia zapewniającemu wysoką prędkość emulacji.

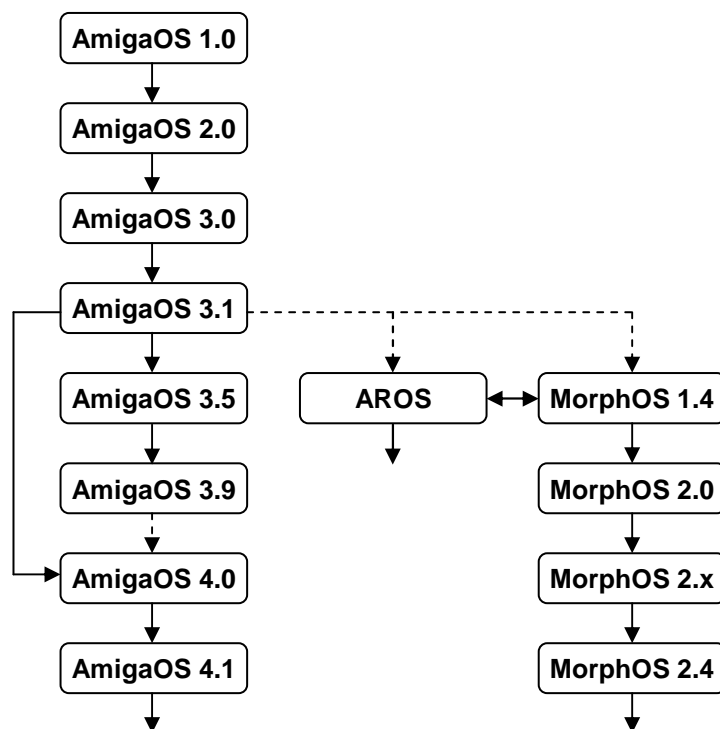


System operacyjny Amigi (zwany kiedyś omyłkowo **Workbench**) jest systemem graficznym, którego podstawowymi elementami są ikony, okna i rozwijane menu. Workbench standardowo uruchamia się w rozdzielczości 640x256 w 4 kolorach w systemie PAL (15 kHz, 50 Hz) na każdej Amidze niezależnie od wersji systemu. Wszystkie operacje wykonuje się za pomocą myszy. System ten jest wielozadaniowy (*multitasking*), tzn. że każda Amiga potrafi wykonywać jednocześnie kilka programów otwartych w oddzielnych oknach lub na oddzielnych ekranach (*screen'ach*). Wykorzystywanie ekranów jest charakterystyczne dla systemu Amigi. Aby przełączyć ekran należy kliknąć lewym przyciskiem myszy na gadżet umieszczony na górnej belce Workbencha po prawej stronie lub nacisnąć kombinację klawiszy: [**Lewa Amiga**] + [**M**]. Można powiedzieć, że screen jest pewnego rodzaju oknem wypełniającym całą powierzchnię monitora. Mechanizm screenów jest bardzo użyteczny w przypadku korzystania z kilku programów jednocześnie, np. na pierwszym ekranie jest Workbench, na drugim jest uruchomiony program graficzny, na trzecim edytor tekstu, a na czwartym przeglądarka Internetowa. W programie graficznym robimy rysunek, który wykorzystamy na naszej stronie WWW, w edytorze tekstu piszemy stronę WWW, a na przeglądarce oglądamy efekty naszej pracy, czyli gotową stronę.

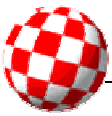
Lewy przycisk myszy służy do uruchamiania niemal wszystkich funkcji, prawy przycisk odpowiedzialny jest za obsługę rozwijanego menu umieszczonego na górnej belce ekranu (np. Workbench'a). W Amidze standardowo używa się myszy dwuprzyciskowych.

Pliki z rozszerzeniem **.info** zawierają ikonę - na ekranie Workbench'a można zobaczyć tylko pliki, które posiadają na dysku plik **.info**, tzn. jeżeli program nazywa się **Calculator** i widzimy jego ikonę na ekranie, to ta ikona znajduje się w pliku "**Calculator.info**". Pliki, które nie posiadają ikon są z poziomu Workbencha niewidoczne. W Workbenchu 2.0 lub nowszym w menu "Window" na górnej belce ekranu jest opcja "show/all files" - po jej uaktywnieniu widzimy wszystkie pliki znajdujące się na dysku. Pomijanie plików nieposiadających ikon pomaga utrzymać porządek w oknach i ułatwia orientację - wyświetlane są tylko ważne ikony. Jest kilka rodzajów ikon: **tool** (plik wykonywalny), **project** (dane), **drawer** (ikona katalogu), **disk** (ikona dysku), **garbage** (ikona śmietnika).

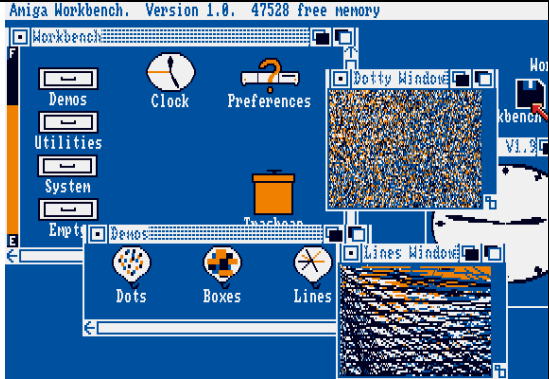
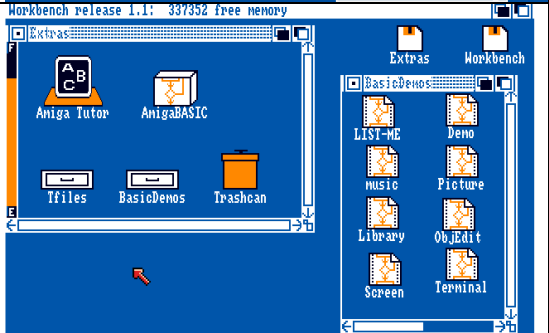
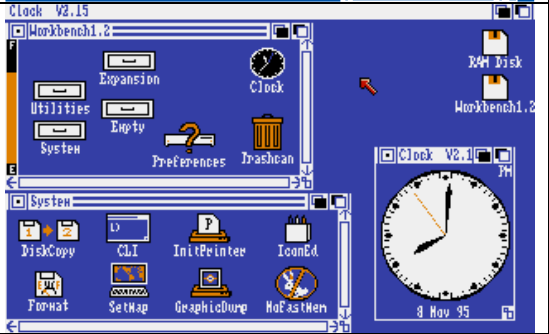
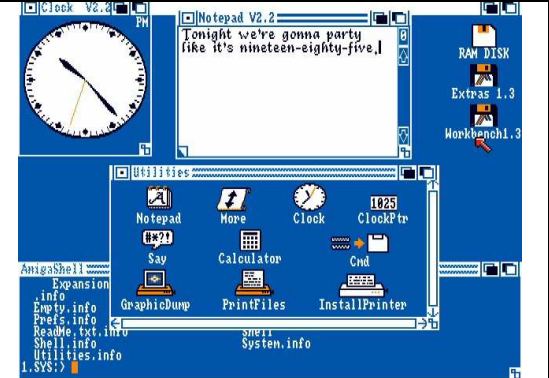
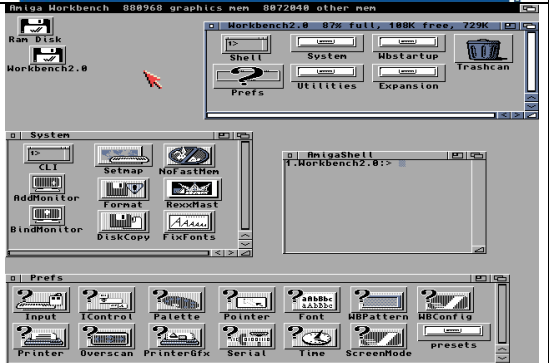
Dyskiety, z których startuje komputer, tzw. boot'owalne można podzielić na dwa rodzaje: dosowe i niedosowe (**NDOS**). Na dyskietkach NDOS nagrane są starsze gry nie korzystające z systemu operacyjnego - zawartości tych dyskietek nie da się obejrzeć pod Workbenchem (nie można zobaczyć jakie pliki znajdują się na dysku, ile miejsca zajmują i co zawierają), nie można też przegrać takiej dyskietki na dysk twardy.





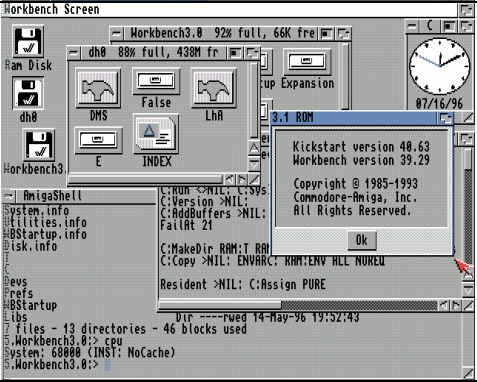
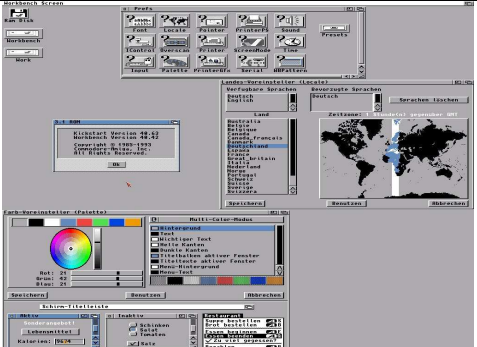
Rozwój systemu i stan obecny







WERSJE SYSTEMÓW

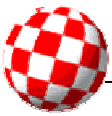
<p>AmigaOS 1.0</p>	<p>1985.10 (Amiga Inc., MetaComCo, Commodore Int.)</p>	<p>(Amiga 1000)</p> <ul style="list-style-type: none"> - w pełni konfigurowane okna i ich wygląd, - dwa stany wyglądu ikon: nieaktywna i aktywna - skróty klawiszowe do typowych działań myszą - wiele <i>screenów</i> dla aplikacji - oparcie <i>AmigaDOS</i> na systemie <i>Tripes</i> napisanym w języku <i>BCPL</i> - niejednolity wygląd systemu 	
<p>AmigaOS 1.1</p>	<p>1985.12 (Commodore Int.)</p>	<p>(Amiga 1000)</p>	
<p>AmigaOS 1.2</p>	<p>1986 (Commodore Int.)</p>	<p>(Amiga 500, Amiga 2000)</p>	
<p>AmigaOS 1.3</p>	<p>1988 (Commodore Int.)</p>	<p>(Amiga 500, Amiga 2000)</p>	
<p>AmigaOS 2.0</p>	<p>1990 (Commodore Int.)</p>	<p>(Amiga 3000)</p> <ul style="list-style-type: none"> - w pełni 32-bitowy - przepisanie <i>AmigaDOS</i> do <i>C</i> i <i>Assemblera</i> - ujednolicenie API dla obsługi wyglądu środowiska graficznego Workbench – jednolity wygląd widżetów - wsparcie dla <i>public screen</i> - wprowadzenie systemowego <i>Installera</i> aplikacji opartego na języku <i>Lisp</i> - wprowadzono system <i>class, commodities</i> - wprowadzono system pomocy – <i>AmigaGuide</i> opartego na hipertekście - usprawniono obsługę klawiatury i myszy 	



<p>AmigaOS 2.04</p>	<p>1991 (Commodore Int.)</p>	<p>(Amiga 600, Amiga 3000)</p>	
<p>AmigaOS 2.1</p>	<p>199x (Commodore Int.)</p>	<ul style="list-style-type: none"> - wprowadzono <i>locale.library</i> – system przetłumaczono na wiele języków 	
<p>AmigaOS 3.0</p>	<p>1992 (Commodore Int.)</p>	<ul style="list-style-type: none"> - wprowadzono system <i>datatypes</i> - obsługa nowych wersji Amig z układami AGA - zwiększenie stabilności i szybkości działania - wsparcie dla napędów CD, urządzeń PCMCIA 	
<p>AmigaOS 3.1</p>	<p>1994 (Commodore Int.)</p>	<p>(Amiga 1200, Amiga 4000)</p>	

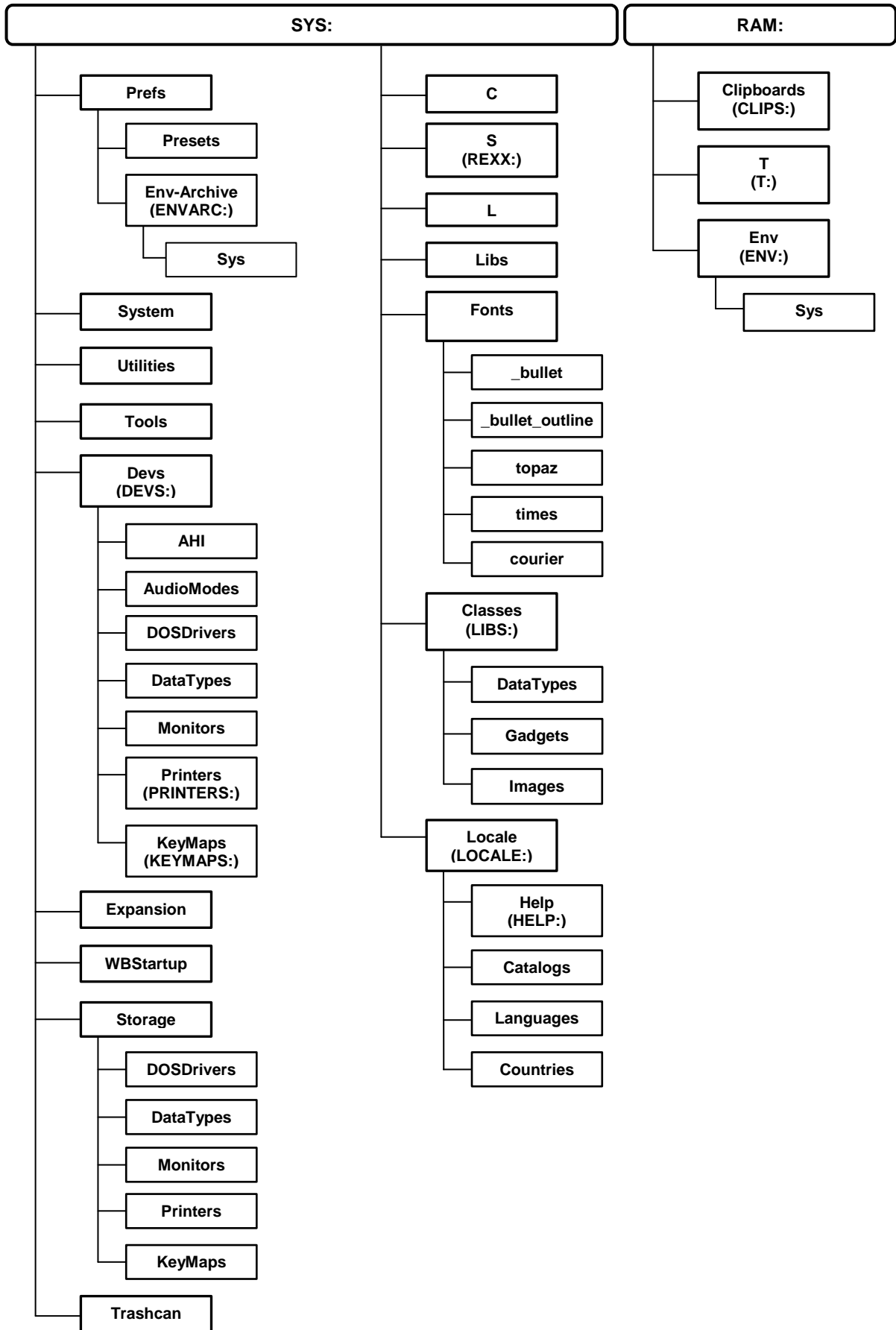
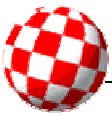


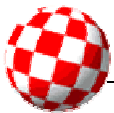
AmigaOS 3.5	1998.10.18 Haager and Partners	(Amiga 1200, Amiga 3000, Amiga 4000) <ul style="list-style-type: none">- całkowita zmiana Workbench'a – wsparcie dla wielu kolorów – paleta 4, 8 lub 16-bitowa oraz monitorów VGA powyżej 31 kHz- ikony 2D, 3D, izometryczne, wielokolorowe- wprowadzono nowe API – ClassACT- wsparcie dla dysków twardych powyżej 4 GB, napędów CD- zintegrowany stos TCP/IP	
AmigaOS 3.9	2000.12.14 Haager and Partners	(Amiga 1200, Amiga 3000, Amiga 4000) <ul style="list-style-type: none">- większe wsparcie dla <i>ClassATC (ReAction)</i>- wprowadzono <i>AmiDOCK</i> – pasek szybkiego uruchamiania programów- nowe datatypy, system wyboru kolorów – koło- kontrola rozmiarów rozdzielczości ekranu	
AmigaOS 4.0	2004.05.26 – Pre Release 2006.12.24 – Final (Hyperion Entertainment CVBA)	(Amiga 1200, Amiga 3000, Amiga 4000, AmigaOne, Micro-A1) <ul style="list-style-type: none">- inteligentny system przydzielania pamięci programom- defragmentacja systemu plików w locie podczas nieaktywności systemu- wsparcie procedur graficznych AltiVec procesora PowerPC 4- optymalizacja systemu pod procesor PPC- programowa emulacja procesora 68k (bez JIT)- pełna obsługa stosu USB, portów szeregowych i równoległych- pełna obsługa kart sieciowych i Internetu przez pakiet <i>Roadshow</i>- obsługa drukarek i kart dźwiękowych- AmigaDOS zrywa z kompatybilnością BCPL, jest w pełni 64-bitowy	
AmigaOS 4.1	2008.08.06 (Hyperion Entertainment CVBA)	(AmigaOne, Micro-A1, Sam440ep, Pegasos II) <ul style="list-style-type: none">- sprzętowy silnik kompozycji obrazu- akceleracja 3D- inteligentne stronicowanie pamięci- wprowadzenie możliwości założenia partycji wymiany- Picture Transfer Protocol (PTP) – wsparcie dla aparatów cyfrowych- wsparcie sterowników dla systemu plików JXFS- nowe i ulepszone funkcje AmigaDOS (w pełni 64-bitowy, automatyczne kasowanie i odświeżanie zawartości aktualnych zasobów dyskowych)- dodanie przeźroczystości w tym <i>AmiDock</i>'a- pełne wsparcie dla podsystemu <i>Warp3D</i>	



KATALOGI

AOS 3.1	AOS 3.9	AOS 4.0	AOS 4.1	Zawartość
C	C	C	C	zawiera komendy i programy systemowe używane w oknie konsoli tekstowej i w skryptach
Classes /DataTypes pliki.datatype /Gadgets pliki.gadgets /Images	Classes	Classes	Classes	
Devs /AHI /AudioModes /DOSDrivers /Keymaps /Monitors /Printers pliki.device	Devs	Devs	Devs	tu znajdują się sterowniki urządzeń zarówno wbudowanych w komputer jak i umieszczonych na dodatkowych kartach i przystawkach. Warto pamiętać o tym, że sterowniki do niektórych podstawowych urządzeń (np. klawiatura, mysz, zegar systemowy) są umieszczone w pamięci ROM Amigi
Expansion	Expansion			
Fonts	Fonts	Fonts	Fonts	ten katalog zawiera czcionki bitmapowe, od wersji systemu 2.0 także wektorowe
		Internet	Internet	aplikacje do obsługi sieci i Internetu – pakiet <i>Roadshow</i>
		Kickstart	Kickstart	ładownalno moduły Kickstartu
L	L	L /CharSets	L /CharSets	w tym katalogu znajdują się dodatkowe systemy plików, oraz tzw. <i>handlers</i> , czyli sterowniki urządzeń logicznych takich jak konsole, potoki i inne znajdują się pliki umożliwiające systemowi odczytywać znaki zdefiniowane w innych standardach niż US-ASCII
Libs	Libs	Libs	Libs	tu znajdują się biblioteki współdzielone, odpowiednik windowsowych *.dll albo linuxowych *.so. W bibliotekach zawiera się większa część funkcjonalności AmigaOS. Kilkanaście najważniejszych bibliotek znajduje się na stałe w pamięci ROM
Locale /Catalogs /english /polski /Countries pliki.country /Help /english /polski /languages	Locale	Locale	Locale	pliki lokalizacyjne systemu i oprogramowania (od wersji AOS 2.1)
		MUI	MUI	podsystem graficzny wraz ze swoim API
Prefs /Env-Archive	Prefs	Prefs	Prefs	tu znajdują się programy do ustawiania różnych parametrów systemu, odpowiednik windowsowego panelu sterowania (<i>AHI, Control, Font, Input, Locale, Overscan, Palette, Pointer, Printer, PrinterGfx, PrinterPS, ScreenMode, Serial, Sound, Time, WBPattern, WBStartup</i> - od AOS4.1); pamiętana ustawienia systemu (<i>ENVARC</i>)
Rexxc	Rexxc			programy do obsługi języka skryptowego <i>Rexx</i> , od 4.0 przeniesiony do C:
S	S	S /ARexx	S /ARexx	to katalog na skrypty, przede wszystkim <i>startup-sequence</i> i <i>user-startup</i> . Te dwa skrypty wykonywane są w czasie startu AmigaOS, uruchamiane są w nich wszystkie niezbędne procesy i usługi. Do pliku <i>user-startup</i> można w razie potrzeby dopisywać własne komendy
Storage /AHI /AudioModes /DOSDrivers /Keymaps /Monitors /Printers pliki.device	Storage	Storage	Storage	tu umieszczone są dodatkowe, w danej chwili niepotrzebne pliki systemowe, np. sterowniki do nieużywanych urządzeń
System	System	System	System	podstawowe programy systemu (<i>CLI, DiskCopy, Format, shell</i>)
T	T			dane tymczasowe, z których korzystają programy i aplikacje
Tools	Tools	Tools	Tools	katalog na programy niezbędne do pracy z systemem, np. do formatowania czy partycjonowania dysków, program do instalowania czcionek itp.
Trashcan	Trashcan	Trashcan	Trashcan	kosz na śmieci
Utilities	Utilities	Utilities	Utilities	drobne programy użytkowe, np. kalkulator, zegarek, przeglądarka <i>MultiView</i>
WBStartup	WBStartup	WBStartup		programy umieszczone wraz z ikonkami w tym katalogu są automatycznie uruchamiane przy starcie systemu, odpowiednik Windowsowej grupy Autostart





URZĄDZENIA WEJŚCIA/WYJŚCIA

AmigaDOS posiada własny system nazewnictwa urządzeń we/wy. Urządzenia logiczne I/O są to urządzenia, które istnieją w systemie i do których w każdej chwili możemy się odwołać wysyłając do nich dane lub dane z nich odebrać. Niekoniecznie są to urządzenia fizycznie dostępne czy widziane. Niektórym z nich możemy nadać własne nazwy. Istnieje pewna grupa urządzeń logicznych z góry przypisanych do systemu, których nazw ani oznaczeń nie można zmienić. Można także tworzyć własne urządzenia logiczne. Każde urządzenie logiczne jest wywoływane przez handler, który jest plikiem znajdującym się w katalogu L: (najczęściej) lub w ROM-ie. Korzystają z niego również wszystkie amigowskie programy.

- CON:** Konsola – jest urządzeniem domyślnym. Wszystkie dane i komunikaty wysyłane są na ekran za pośrednictwem okienka **CLI/Shell**.
- DFx:** Napęd dyskietek elastycznych, gdzie x = 0,1,2,3 oznacza kolejne zamontowane stacje dysków. Stacja **DF0:** jest napędem montowanym wewnątrz Amigi. Kolejna **DF1:** jest podłączona do złącza z tyłu komputera. Do niej zaś następną, tzn. jedna do drugiej (stacje są przelotowe).
- DHx:** Napęd stały, gdzie x to liczba naturalna. Pierwszy dysk twardy ma oznaczenie **DH0:** jest na nim zapisany m.in system operacyjny. Jest to nazwa zwyczajowa niekoniecznie stosowana.
- RAM:** Emulowany w pamięci **Ram-Dysk**, czyli "dysk w pamięci komputera". Możliwy jest w nim odczyt/zapis/usunięcie plików i katalogów oraz uruchamianie programów. Jego zawartość jednak jest tracona po resecie. Rozmiar Ram Dysku jest zawsze równy znajdującym się w nim danych i ograniczony z góry ilością wolnej pamięci RAM. Jest zakładany automatycznie przy starcie systemu.
- RAD:** Nieresetowalny **Ram-Dysk**. Ma stałą pojemność, a zawartość nie jest kasowana po resecie komputera. Ginie dopiero po zaniku zasilania lub usunięciu go komendą "**RemRAD**". Nie jest zakładany przez system. Montowany jest ręcznie komendą "**Mount RAD**". Po jego założeniu ilość pamięci RAM jest pomniejszona o jego rozmiary.
- SYS:** Dysk, z którego został załadowany system operacyjny.
- CD0:** Napęd CD-ROM.
- CC0:** Wykorzystywane do obsługi kart PCMCIA.
- PCx:** Umożliwia obsługę pecetowych dyskietek 720 KB w amigowej stacji poprzez program **CrossDos**, x = 0,1,2,3 odnosi się do kolejnych stacji w Amidze.
- SER:** Komunikacja z portem szeregowym, a ściślej z urządzeniem podłączonym do tego portu przez "serial.device"..
- PAR:** Umożliwia komunikację z portem równoległym. Najczęściej odnosi się do drukarki przez "parralel.device" z pominięciem zmian generowanych przez sterownik drukarki.
- PRT:** Wysyłanie danych na drukarkę. Odnosi się do drukarki przez "printer.device", do którego uprzednio odwołuje się sterownik drukarki zmieniając odpowiednio dane.
- NIL:** Urządzenie to pełni rolę śmietnika. Wszystkie wysłane do niego dane są tracone. Zastosowanie tego urządzenia polega na przekierowaniu do niego komunikatów, które generują pewne komendy, a nie chcemy ich (komunikatów) czytać.
- PIPE:** Kontroluje wyjście/wejście zapewniając komunikację między programami. To co pierwsze trafia do **PIPE:**, jako pierwsze z niego wychodzi (analogia do **stosu – stock**).
- QUEUE:** Urządzenie ustawiające "w kolejce" dane oczekujące na wyjście z komputera.
- *:** Aktywne okno.

Aby skierować działanie jakiejś komendy do konkretnego urządzenia piszemy na końcu wiersza "**>urządzenie**" (np. "**>NIL:**"). Na przykład użycie komendy "**Echo**" wyświetlającej podany w parametrze tekst między znakami cudzysłowiu może wyglądać tak:

- Echo "To jest Tekst" >CON:** Wyświetli ten tekst w okienku **CLI/Shell**. Możemy również wpisać:
Echo "To jest Tekst" Efekt będzie ten sam. Tekst zostanie skierowany na ekran, ponieważ konsola jest urządzeniem domyślnym. Natomiast wpisanie:
- Echo "To jest Tekst" >NIL:** Nie wyświetli żadnego tekstu, gdyż został on skierowany do śmietnika. Możemy również wysłać dane przez porty lub na drukarkę:
- Echo "To jest Tekst" >SER:**
Echo "To jest Tekst" >PAR:
Echo "To jest Tekst" >PRT:

OBSŁUGIWANE SYSTEMY PLIKÓW

Natywne:

- OFS – Old File System
- FFS – Fast File System
- FFS2 – Fast File System 2 (od AOS 4.0)
- SFS – Smart File System
- PFS – Professional File System
- JxFS

Obce:

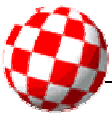
- CrossDOS (FAT12, FAT16, FAT32)
- CDFS (ISO 9660 + RockRidge + Joliet, UDF, CDDA, CDXA VideoCD, Multisession)
- HFS, HFS+

NATYWNE ATRYBUTY PLIKÓW

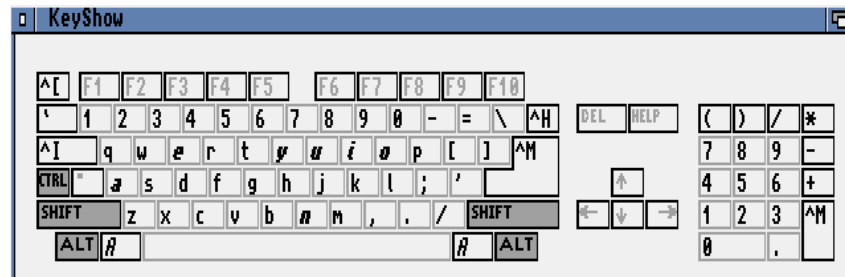
attribut (flaga)	znaczenie
d	deletable / kasowalny
e	executable / wykonywalny
w	writable / zapisywalny
r	readable / odczytywalny
a	archive / do użytku programów archiwizujących
p	pure / rezydentny (uruchamialny wiele razy z pamięci)
s	script / skrypt AmigaDOS
h	hidden / ukryty (ta flaga nie działa !!!)

**MAPA PAMIĘCI KOMPUTER AMIGA**

000000 256K bajtów chip RAM	03FFFF
040000 256K bajtów chip RAM (karta rozszerzeń Amiga 1000)	07FFFF
080000 512K bajtów rozszerzonego chip RAM (do 1 MB)	0FFFFFFF
100000 1024K bajtów rozszerzonego chip RAM (do 2 MB) (wszystkie modele z chipsetami ECS lub AGA)	1FFFFFFF
200000 Pierwsze 8 MB przestrzeni autokonfiguracji (Fast RAM)	9FFFFFFF
A00000 Zarezerwowana	BEFFFFFF
BFD000 8520-B (dostęp jedynie na adresy even-byte)	BFD000
BFE001 8520-A (dostęp jedynie na adresy odd-byte)	BFE001
C00000 Wewnętrzne karty rozszerzeń (pseudo-fast, Slow RAM w Amiga 500)	D7FFFFFF
D80000 Zarezerwowana	DBFFFFFF
DC0000 Zegar czasu rzeczywistego	DCFFFFFF
DD0000 Zarezerwowana	DFFFFFFF
DF0000 Rejestry innych chipsetów	DFFFFFFF
E00000 Zarezerwowana	E7FFFFFF
E80000 Przeźródź autokonfiguracji. Tutaj pojawia się Błat, a system przenosi go do ostatecznego adresu	E8FFFFFF
E90000 Druga przestrzeń autokonfiguracji (używane 64K w urządzeniach wejścia/wyjścia)	FFFFFFF
F00000 256K System ROM (Kickstart 2.0)	FBFFFFFF
FC0000 256K System ROM	FFFFFFF



SKRÓTY KLAWISZOWE

Menu Workbench:

R.Amiga + B	Przełącznik okno/tło <i>Blatu Workbench'a (Backdrop WB)</i>
R.Amiga + E	Wywołanie <i>CLI</i>
R.Amiga + Q	Wyjście z Workbench'a do AmigaDOS
R.Amiga + ?	<i>Help</i>

Menu Window:

R.Amiga + N	Nowy katalog (<i>drawer, szuflada</i>)
R.Amiga + K	Zamknięcie aktywnego okna
R.Amiga + A	Zaznaczenie wszystkiego
R.Amiga + .	Wyrównanie wszystkiego w bieżącym oknie (katalogu)

Menu Icons:

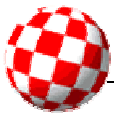
R.Amiga + O	Otwarcie zaznaczonego obiektu, ikony - <i>open</i>
R.Amiga + C	Kopiowanie zaznaczonego obiektu - <i>copy</i>
R.Amiga + R	Zmiana nazwy obiektu - <i>rename</i>
R.Amiga + I	Informacje o obiekcie
R.Amiga + S	Utrwalenie pozycji - <i>snapshot</i>
R.Amiga + U	Odbezpieczenie pozycji ikon - <i>unsnapshot</i>
R.Amiga + L	Wrzuć na Błat – <i>Leave Out</i>
R.Amiga + P	Cofnij ostatnią czynność – <i>Put Away</i>

Inne:

CTRL + Amiga + Amiga	Twardy reset
L.P.M + P.P.M przy uruchamianiu	Wejście do <i>Boot Menu</i> Amigi

Amiga z kartą turbo Blizzard:

ESC przy uruchamianiu	Wejście do <i>Boot Menu</i> karty <i>Blizzard PPC</i>
2 przy uruchamianiu	Deaktywacja karty <i>Blizzard</i> (Amiga jej nie wykryje)

Amiga OS 4.0:

L.Amiga + strzałki Num	Ruch kursorem po ekranie, którego obszar jest większy od aktualnie używanej rozdzielczości
Shift + gadżet zmiany rozmiaru	Maksymalizacja okna
R.Amiga + gadżet zamykania	Zamknięcie wszystkich okien <i>Workbench'a</i>
Ctrl + Alt	Skalowanie okna, łapiąc za dowolną krawędź ramki
Alt + klik na dowolne okienko	Sprowadzenie okna do ikony (należy zainstalować commodity <i>AmiSnap</i> z <i>OS4Depot</i>)
Shift (przytrzymanie)	Ułatwia zaznaczać/odznaczać opcje, gdy używamy blokującego się górnego menu
Shift (przytrzymanie)	Pozwala wysuwać okna poza ekran (jeśli wysuwanie ich jest domyślnie wyłączone)
Ctrl + Amiga	Możliwość przesuwania okna, łapiąc go w dowolnym miejscu (przydatne np. wtedy, gdy okno nie ma belki do przesuwania)
Ctrl + Alt	Skalowanie okna, łapiąc za dowolną krawędź ramki
R.Alt	Wyłączenie przesuwania/skalowania okienka z zawartością
Help (Scroll Lock to help w PC)	Przytrzymanie po resecie z klawiatury to wejście do <i>Early Startup Menu</i> (można tam oczywiście wejść także starym sposobem - przytrzymując dwa klawisze myszki)
Ctrl (przytrzymanie po resecie z klaw.)	Start bez sekwencji startowej
Ctrl + Alt + Alt	Twardy reset (m.in. wyrzucenie <i>Kickstartu</i> z pamięci)
Ctrl + Alt + M	<i>USB Massstorage GUI</i> (pozwala m.in. na sformatowanie nośnika pod wybranym filesystemem)
L.Amiga + A lub przytrzymanie Shift lub S.P.M	Zamiana sposobu przesuwania ekranu zdefiniowanego przez użytkownika: <i>normalny / alternatywny</i>

AmigaShell AOS 4.0 (bez KingCON):

Shift + Ctrl + klawisz kursora	Koniec/początek kolumny
Shift + klawisz kursora	Początek następnego/poprzedniego słowa w wierszu

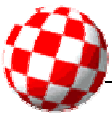
Systemowy NotePad AOS 4.0 (po wyłączeniu klawiatury numerycznej):

Insert	Ttryb wstawiania i zastępowania
Home	Pierwsza kolumna
End	Ostatnia kolumna
PageUP	Pierwszy wiersz
PageDown	Ostatni wiersz

Objaśnienia:

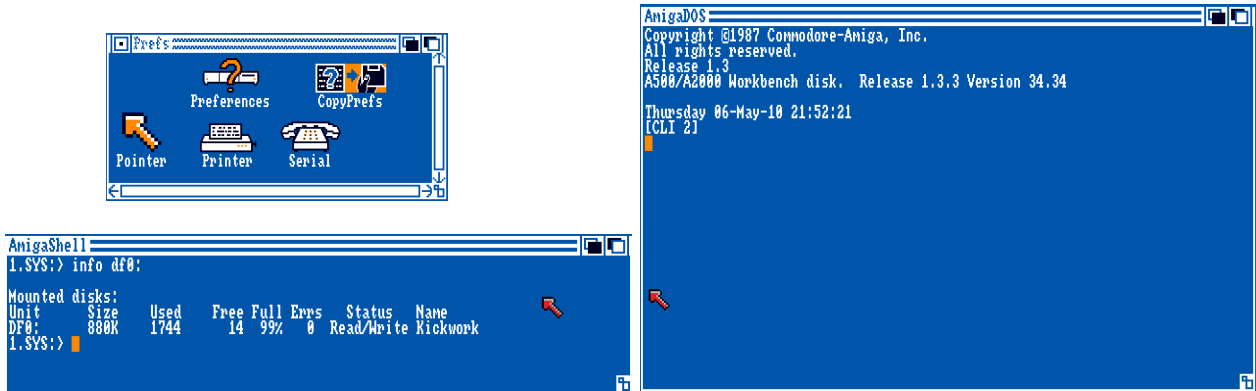
L.P.M.	Lewy przycisk myszy
S.P.M.	Środkowy przycisk myszy
P.P.M.	Prawy przycisk myszy
R.	Prawa/Prawy
L.	Lewa/Lewy

UWAGA!!! używając klawiatury komputera PC przyjmujemy, że klawisze prawa i lewa „**Amiga**“ = „**Windows**“

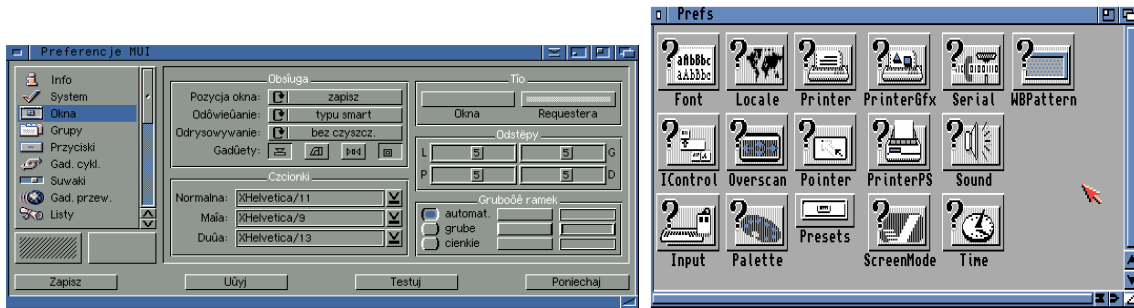


AMIGA OS CLASSIC

„AmigaOS 85 = Windows 95“

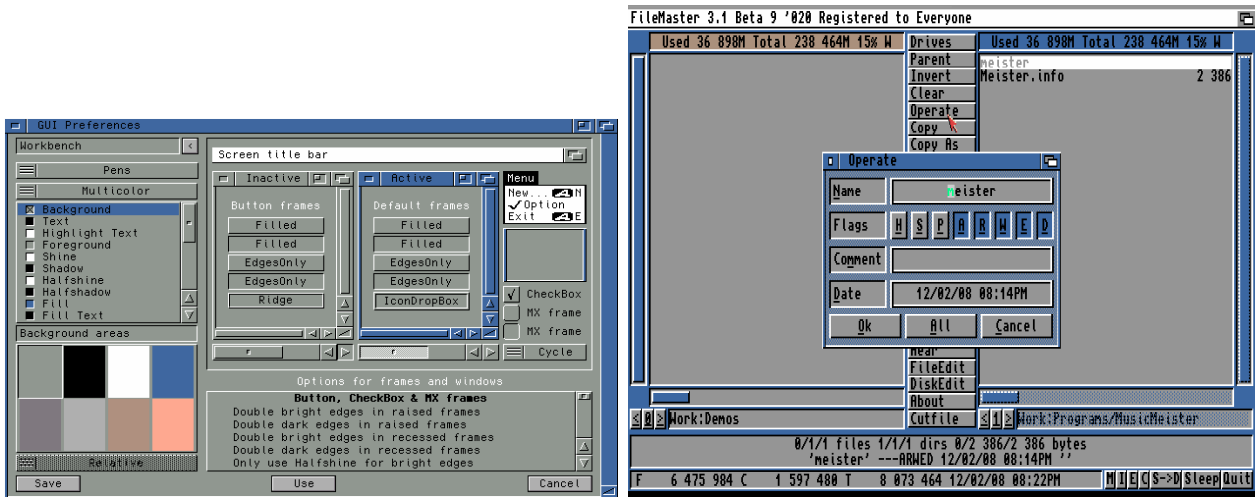


Katalog Prefs, AmigaShell i AmigaDOS w AOS 1.3



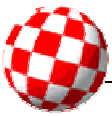
MUI 2.3 zainstalowane w AOS 3.1

Zawartość Preferencji w AOS 3.1



GUI Preferences w AOS 3.1

Panelowy menadżer plików



AMIGA OS 4.0/4.1

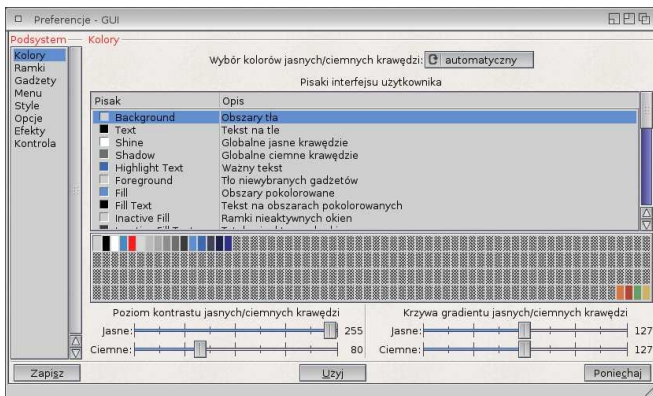
„AmigaOS 4.0 - mały krok w dziejach komputerów, wielki skok dla użytkowników Amigi”

W nowych Amigach ROM komputera zawiera oprogramowanie OpenFirmware zwany UBoot. Jego zadaniem jest inicjalizacja sprzętu i rozpoczęcie procesu uruchamiania systemu - AmigaOS lub Linuxa. UBoot nie zawiera wszystkich komponentów kickstartu (ExecSG, FFS2) lecz przechowywane są na dysku twardym. Komponenty te muszą być umieszczone w katalogu SYS:Kickstart. Zostaną załadowane do pamięci i stamtąd zostaną uruchomione. Pozostała część systemu zostanie załadowana przez standardową sekwencję Startup-Sequence. Proces uruchomienia nowej Amigi przebiega tak:

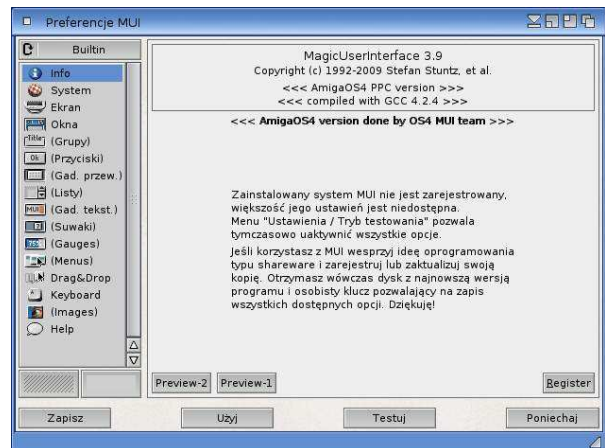
- **UBoot** sprawdza napęd, z którego ma być uruchomiony system (CDROM, twardy dysk...)
- odszukuje na nim **SLB (Second Level Booter)** zainstalowany na **RDB** dysku podczas instalacji AmigaOS 4.0/4.1
- SLB sprawdza wszystkie partycje dysku, które zostały oznaczone jako **"bootowalne"** i wybiera tę, która posiada najwyższy priorytet
- w przypadku AmigaOS, SLB odszukuje plik tekstowy **SYS:Kickstart/Kicklayout** zawierający różne konfiguracje modułów Kickstartu
- SLB zgodnie z wpisami danej konfiguracji Kicklayout ładuje moduły Kickstartu, które znajdują się w **SYS:Kickstart**
- następnie wszystkie moduły zostają uruchomione i rozpoczyna się bootowanie systemu z sekwencji startowej (w tym przypadku kontrolę przejmuje **dos.library**)
- zostają wykonane skrypty **Shell-startup**, **Startup-Sequence** i **User-startup**



W nowym AmigaOS obowiązuje zestaw znaków **ISO-8859-2**, stare katalogi lokalizacyjne w **AmigaPL** nie stanowią jednak problemu - w sposób niezauważalny dla nas, "w locie" system konwertuje te pliki do nowego standardu. Jedyny problem może stanowić dokumentacja, która pozostaje nadal w **AmigaPL**. Komendą **CharsetConvert** można skonwertować tekst z **AmigaPL** do **ISO**.



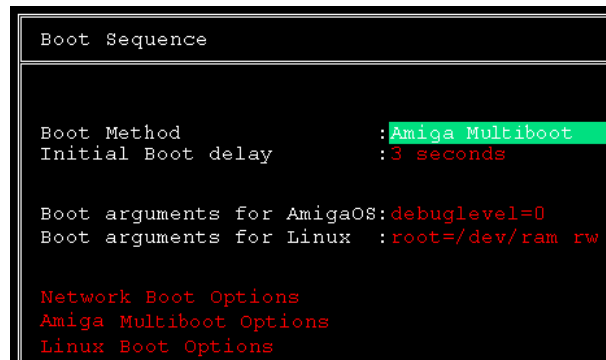
Preferencje GUI ReAction w AOS 4.1



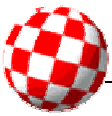
Preferencje w MUI 3.9 w AOS 4.1



Zawartość katalogu Prefs: w AOS 4.1



UBoot komputera AmigaOne



MORPH OS



MorphOS

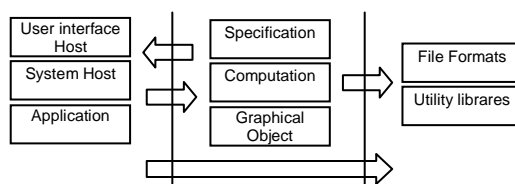


2002 grudnia	MorphOS 1.0
2003 sierpień 08	MorphOS 1.4
2004 marzec 06	MorphOS 1.4.2
2005 marzec 28	MorphOS 1.4.4
2005 kwietnia 30	MorphOS 1.4.5
2005 sierpnia 24	MorphOS 1.4.5 Classic
2008 czerwca 30	MorphOS 2.0
2008 września 06	MorphOS 2.1
2008 grudnia 20	MorphOS 2.2
2009 sierpnia 06	MorphOS 2.3
2009 październik 12	MorphOS 2.4

System operacyjny tworzony przez Genesi od 1999 r. do 2003 roku jako następcą Amiga OS (systemu operacyjnego dla komputerów Amiga). Następnie po porzuceniu przejęty przez grupę MOS Team i rozwijany do chwili obecnej. Pracuje na maszynach: Pegasos 1, Pegasos 2, Amiga 4000, Amiga 3000, Amiga 1200 z procesorem PowerPC, Efika i Apple MacMini G4.

**Cechy systemu:**

- mikrokernel **Quark**
- skalowalny
- wielozadaniowy z wyłączeniem (pre-emptive multitasking)
- lekki, elastyczny, modułarny
- w pełni zorientowany obiektowo
- duża wydajność i szybkość działania
- małe zużycie pamięci
- prostota i łatwość użytkowania
- kolorowy i efektowny interfejs użytkownika
- edytowalne panele ikon (odpowiednik AmigaDock)
- rozbudowane **hotkeys** – skróty klawiszowe
- pełna obsługa **datatypes** z możliwością rozbudowy o dodatkowe klasy
- wbudowany **Reggae** zorientowany na multimedia z możliwością rozbudowy o dodatkowe klasy
- kompatybilny na poziomie api z AmigaOS 3.1
- możliwość uruchamiania oprogramowania z MC68k przez dwa emulatorzy: emulator **Trance** z **JIT (Just In Time Compiler)** – dynamiczną rekompilacją kodu) oraz wkompiłowany w jądro systemu (pod warunkiem, że nie odwołują się do układów specjalizowanych klasycznych Amig)
- emulacja podsystemu WarpOS (WOS) i PowerUP (PUP)
- obsługa wszystkich amigowych filesystemów
- obsługa filesystemów obcych: FAT, NTFS, EXT2, EXT3, CDFS, HFS+ (do odczytu)
- api systemu oparte na **MUI (Magic User Interface)**
- graficzny interfejs użytkownika pełni środowisko **Ambient**, w pełni konfigurowalne przez użytkownika
- pełna akceleracja 3D - Rave3D, TinyGL – OpenGL, Warp3D
- pełne wsparcie dla stosu protokołów TCP/IP przez pakiet **Netstack**
- pełna obsługa stosu USB 2.0 przez pakiet **Poseidon**
- brak deficons
- brak wsparcia dla dyskietek

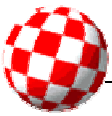


Platform level

Kernel level

Support level

Warstwy systemu MorphOS (MOS)



Ustawienia systemu przy pomocy Preferencji

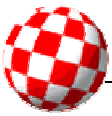


Ustawienia MUI

Fragment struktury katalogów systemowych:

C	komendy i polecenia DOS użytkownika
CLASSES	
/ MUI	dodatkowe klasy MUI użytkownika
LIBS	biblioteki użytkownika
MOSSYS	elementy natywne systemu; niezależa się jakichkolwiek modyfikacji
/ C	komendy i polecenia DOS systemu
/ LIBS	biblioteki systemu
/ S	skrypty systemu
S	skrypty użytkownika

Wprowadzenie dodatkowego katalogu **MOSSYS** zapewnia bezpieczeństwo systemu oraz konfliktom. Użytkownik ma możliwość umieszczania swoich plików w głównym katalogu, natomiast system kiedy odwołuje się do określonego elementu najpierw przeszukuje katalogi użytkownika a następnie systemowe. Dzięki temu w przypadku obecności jakiegoś pliku jednocześnie w np. **LIBS**: i **MOSSYS/LIBS**: wczytany zostanie on z tej pierwszej lokacji i nie będzie mowy o żadnych konfliktach. Użytkownik zaś eksperymentując z plikami systemu nie musi ingerować w zawartość żywotnych dla MorphOS-a katalogów.



AROS Research Operating System



Początek projektu datuje się na rok **1995**. Początkowo zmierzał do uzyskania **kompatybilności z AmigaOS 3.1**. Z czasem w wielu dziedzinach przewyższył oryginał. Początkowo rozszerzenie nazwy brzmiało "**Amiga Research Operating System**" jednak ze względu na zastrzeżenie prawne nazwy Amiga w 2007 roku zmieniono je na obecne **AROS**.

Dzięki „**bounties**” powstał m.in. stos TCP/IP (który następnie portowano w ramach analogicznego "bounty" do systemu MorphOS), stos USB, system plików SFS, natywny port GCC, sterowniki kart graficznych nVidia i ATI, przeglądarka internetowa czy wersja systemu dla architektury 64-bitowych procesorów x86.

Z powodu dużych braków w oprogramowaniu jednym z priorytetów jest uzyskanie zgodności z procesorem **Motorola MC68000**. W chwili obecnej zgodność tą uzyskuje się przy pomocy emulatora UAE. Technologia **AmiBridge** umożliwia uruchamianie programów napisanych dla komputera Amiga w postaci oddzielnych instancji UAE bezpośrednio z ich ikon. Podczas uruchamiania programu ładowane są jedynie niezbędne pliki systemowe i sam program, nie jest uruchamiany Workbench. Użytkownik wskazuje program, który chce uruchomić z poziomu AROS-a, a AmiBridge tworzy wszystkie potrzebne pliki i ikonę programu. Wymagane są pliki systemowe AmigaOS 68k i oryginalny ROM – kickstart Amigi w postaci pliku.

Cechy systemu:

- ogólnodostępny (otwarty kod źródłowy)
- przenośny
- wielozadaniowy z wyłączeniem
- obsługa języka **Rexx**
- dynamicznie rozszerzający się **RAM dysk**, który dopasowuje swoją wielkość do zawartości – do niego są ładowane pliki konfiguracyjne oraz mogą tu być kopiowane inne pliki
- technologia deficons (***.icons**) znana z AmigaOS
- obsługa TCP/IP, USB, SFS, GCC, drukarek
- dostępny na platformy **x86, PPC**
- rozwój zawdzięcza tzw. „**bounties**” – zbiórkom pieniędzy na określony projekt
- **ZUNE** - api do budowy graficznego interfejsu użytkownika programów (zamiennik Intuition czy MUI)
- **Wonderer** – graficzne środowisko GUI wzorowane na Workbench'u

Dystrybucje:

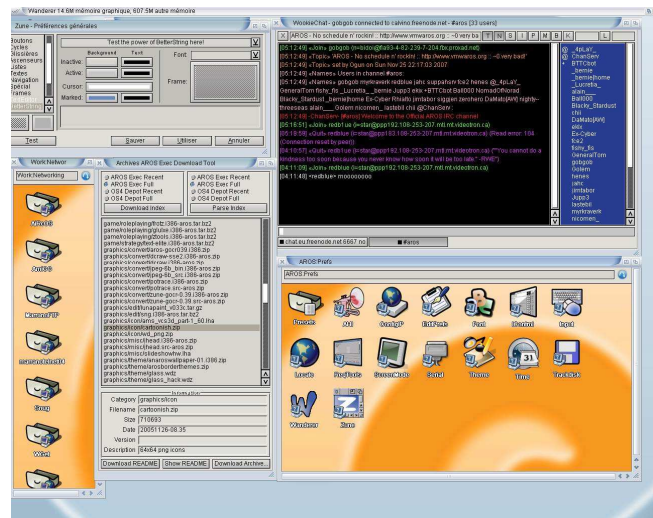
VMWaros	preinstalowane środowisko dla VMware
Icaros Desktop	w wersji LiveDVD z opcją instalacji na dysku
Icaros Desktop	preinstalowane środowisko dla VMware oraz VirtualBox
WinAROS	preinstalowany środowisko dla QEMU
ArosMAX	nierozwijana dystrybucja przeznaczona do instalacji w architekturze x86

Typy ikon:

narzędzie	program wykonywalny
projekt	plik danych wykonywalnego programu. Program, którym stworzono plik danych jest określony w pliku jego ikony. Kliknięcie na ikonie typu projekt powoduje uruchomienie przypisanego programu i załadowanie pliku danych.
katalog	
napęd	fizyczny dysk lub RAM dysk

Struktura katalogów systemowych:

C	komendy i polecenia DOS shella
Classes	katalog powiązany z urządzeniem logicznym LIBS; datatypy i rozszerzenia interfejsu użytkownika
Devs	sterowniki
Fonts	czcionki
Libs	biblioteki
Locale	tłumaczenia systemu i programów
Prefs	programy zarządzające ustawieniami systemu oraz środowisko AROS
S	skrypty
System	proste programy systemowe
Tools	proste narzędzia systemowe
Utilities	proste programy użytkowe





KOMENDY POWŁOKI CLI/SHELL

UWAGA: system AmigaOS, pomimo, że zapamiętuje duże/małe znaki nie rozróżnia ich !!! Stąd polecenia `Format`, `FORMAT`, `format` to są te same polecenia dla AmigaDOS.

Format wydawanych komend

W AmigaDOS większość poleceń do działania wymaga podania stosownych argumentów. Dlatego w większości przypadków użycie samego rozkazu prowadzi do wyświetlenia komunikatu o błędzie. Aby dowiedzieć się czego wymaga każda komenda należy wpisać po niej znak zapytania, np:

```
Copy ?
Format ?
Assign ?
```

W wyniku czego wyświetli ona swoją składnię czyli spis możliwych argumentów. Dla polecenia `Format` może to wyglądać tak:

```
Format ?
DEVICE=DRIVE/K/A, NAME/K/A, OFS/S, FFS/S, INTL=INTERNATIONAL/S, NOINTL=NOINTERNATIONAL/S,
DIRCACHE/S, NODIRCACHE/S, NOICONS/S, QUICK/S:
```

Zakończony jest on znakiem dwukropku umożliwiając wpisanie teraz tych argumentów. Jeżeli naciśniemy klawisz `[ENTER]` nie podając nic, komenda zakończy działanie z błędem. W tym przypadku wyświetli komunikat **"Requires argument missing"** - czyli brak wymaganych argumentów. Jest to równoznaczne z wpisaniem samej komendy.

W argumentach po znaku `/` (**slash**) znajdują się jednoliterowe oznaczenia, które informują jakiego typu ma być podawany argument:

<code>/A</code>	argument musi zostać podany
<code>/F</code>	argument musi zostać podany jako ostatni
<code>/K</code>	argument jest słowem kluczowym podawanym wraz z parametrem
<code>/M</code>	argument może zostać powtórzony wiele razy
<code>/N</code>	argument musi być liczbą
<code>/S</code>	opcja lub przełącznik

```
DEVICE=DRIVE/K/A, NAME/K/A
```

Argumenty te są słowami kluczowymi (w tym przypadku `DEVICE` lub `DRIVE` oraz `NAME`) z parametrami (`/K`) i muszą zostać podane (`/A`). Pozostałe argumenty to opcje (`/S`) modyfikujące działanie rozkazu, które mogą ale nie muszą być podane. Argumenty podajemy rozdzielając je pojedynczym odstępem (klawisz `[SPACEBAR]`). Końcowe wywołanie naszego polecenia `Format` może więc wyglądać tak:

```
Format DRIVE DF0: NAME Dane
```

lub tak:

```
Format DEVICE DF1: NAME "Dysk na dane" FFS NOICONS QUIET
```

Parametrem słowa kluczowego `DRIVE` (lub `DEVICE`) jest w naszym przykładzie słowo `DF0`: (`DF1:`) - czyli symbol stacji dysków. Zaś dla `NAME` etykieta `Dane` (lub `"Dysk na dane"`) - czyli nazwa dla sformatowanej dyskietki.

W drugim przykładzie włączyliśmy opcję `FFS` (Fast File System) - czyli dyskietka zostanie sformatowana w nowszym systemie plików `FFS` (880 KB pojemności). Domyślnie jest `OFS` dlatego jej nie podałem (parz pierwszy przykład).

Wzorzec

Każda komenda AmigaDOS posiada określoną składnię parametrów jakie możemy lub musimy podać, aby komenda zadziałała.

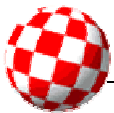
```
copy ?
```

```
FROM/M, TO/A, ALL/S, QUIET/S, BUF=BUFFER/K/N, CLONE/S, DATES/S, NOPRO/S, COM/S, NOREQ/S
```

Ten ciąg instruuje nas jakie parametry przyjmuje komenda i w jakiej formie należy je podawać (oczywiście każda komenda ma inną składnię parametrów). Za każdym parametrem (lub nazwą parametru) znajduje się poprzedzony znakiem `/` jego rodzaj. Rozróżniamy następujące rodzaje parametrów:

<code>/A</code>	jest niezbędny do działania komendy i musi zostać podany
<code>/F</code>	oznacza, że ten parametr zawsze musi być podany jako ostatni
<code>/K</code>	musi być dodatkowo określony słowem kluczowym (tekstem)
<code>/M</code>	może przyjmować więcej niż jeden element (zazwyczaj kilka nazw plików)
<code>/N</code>	musi być dodatkowo określony wartością numeryczną (liczbą)
<code>/S</code>	parametr jest przełącznikiem (switchem lub dyrektywą). Aby parametr zadziałał musi być podana jego nazwa

Rodzaje parametrów mogą być łączone, np. `/A/N` będzie oznaczać, że parametr jest niezbędny do działania i musi zostać podany oraz, że należy go określić wartością numeryczną.

**JOKERY (GLOB, GLOBBING, SYMBOLE WIELOZNACZNE)**

Jokery można inaczej określić mianem zamienników. Występują nie tylko w AmigaDOS, lecz w każdym systemie. W większości przypadków są podobne, choć zdarzają się unikalne jokery w różnych systemach. Zadaniem jokerów jest zastąpienie znaku, ciągów znaków, czy też w oknach wyboru plików wybór nazw z danym wzorcem. Występujące w AmigaDOS jokery:

? (znak pytajnika)	zastępuje jeden znak we wzorcu, np. copy krz?k.txt TO RAM : spowoduje przekopiowanie plików o podanym wzorcu, które na czwartym miejscu mają dowolną literę alfabetu (krzyk, krzak, krzek, krzck itd.). Znak ? może występować kilka razy we wzorcu.
#? (hash i pytajnik)	oznacza dowolne znaki. Jeżeli zastosujemy w kombinacji copy k#?.txt TO RAM : wówczas nastąpi przekopiowanie wszystkich plików z rozszerzeniem .txt rozpoczynających się na literę k. Jeżeli zastąpimy rozszerzenie jokerem #? lub całkowicie je pominiemy, nastąpi przekopiowanie wszystkich plików rozpoczynających się literą k z dowolnym rozszerzeniem. Można również stosować zapis w nawiasie (#?).
~ (znak tyldy)	wykorzystywany do negacji, czyli zrobienia czegoś przeciwnego niż podano we wzorcu, np. copy ~(k#?).#? TO RAM : spowoduje przekopiowanie wszystkich plików za wyjątkiem tych, zaczynających się literą k.
% (znak procenta)	znak "pusty".
(znak separacji)	można go sobie tłumaczyć jako "lub". Pozwala wstawić więcej niż jeden warunek wzorca. Przydatne w oknach wyboru plików, gdy na przykład chcemy, aby program ograniczył się tylko do plików mających w rozszerzeniu pliki graficzne, np. (#?.gif #?.jpg #?.png) ,
[] (nawiasy kwadratowe)	można powiedzieć, że pełnią funkcję narzędzia automatyzacji. W nawiasach podawane są zawsze trzy dowolne znaki [123] lub dwa rozdzielone myślnikiem [1-8]. W pierwszym przypadku wzorec obejmuje pliki, które posiadają jednoznakową nazwę (1, 2 lub 3; nie znajdzie 12, 13, 23 itd.). W drugim przypadku wzorec obejmuje pliki o jednoznakowej nazwie z przedziału od 1 do 8. Jeżeli podamy np. copy (frame[134]).jpg TO RAM : polecenie przekopiuje pliki frame1.jpg, frame3.jpg i frame4.jpg.

Praca z konsolą SHELL

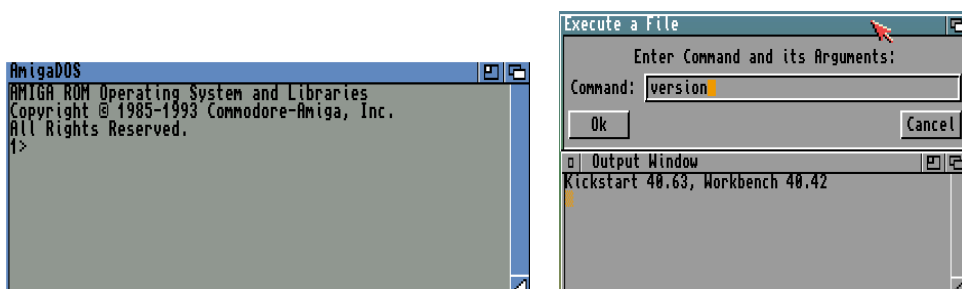
Okno *Shell* jest urządzeniem, w którym następuje komunikacja użytkownika z systemem *AmigaDOS*. Po otwarciu tego okna zostaje w pamięci utworzony bufor o wielkości 20 kB, w którym przechowywane są użyte przez użytkownika polecenia. Aby powtórzyć użytą wcześniej komendę, wystarczy nacisnąć klawisz kursora "w górę".

Parametry, z jakimi otwiera się okno *Shell* mogą być zmienione w oknie *Tool* opcji *Information* z rozwijanego menu *Workbencha*:

WINDOW=CON:x/y/wys/szer/tytuł/OPCJE

CON:	kod okna w nazewnictwie AmigaDos
x	współrzędna x górnego, lewego rogu okna Shell
y	współrzędna y górnego, lewego rogu okna
wys	wysokość okna (mierzona w punktach)
szer	szerokość okna (mierzona w punktach)
tytuł	tytuł okna – dowolny ciąg znaków o długości do 30 znaków

OPCJE	dodatkowe parametry pracy okna Shell
AUTO	okno otworzy się automatycznie, jeżeli któryś z pracujących programów będzie wymagał wprowadzenia lub wyświetlenia danych
CLOSE	okno będzie posiadać wszystkie standardowe gadżety (gadżet zamknięcia, głębokości, skalowania itp.)
BACKDROP	okno będzie otwierać się jako tło pod wszystkimi innymi otwartymi oknami
NOBORDER	okno nie będzie posiadać żadnej ramki
NODRAG	okno nie może być przesuwane
NOSIZE	okno nie będzie posiadać gadżetu skalowania wielkością okna
SIMPLE	powoduje, że po zmianie wielkości okna tekst zawarty w nim pozostanie w niezmienionej postaci - nie dopasuje się do nowych rozmiarów okna
WAIT	spowoduje, że okno będzie mogło zostać zamknięte tylko gadżetem zamknięcia okna, nie poleceniem <i>Endcli</i> lub <i>Endshell</i>





Alfabetyczny spis poleceń AmigaDOS

AddBuffers

Zwiększa bufor (cache) określonego urządzenia. Pozwala to zmniejszyć odwołania do tego napędu a tym samym nieznacznie przyspieszyć z nim pracę. Zajmuje cenną pamięć Chip, więc nie należy przesadzać z jego wielkością. W przypadku dysku twardego - działa tylko z plikiem .HDF.

AddBuffers DRIVE/A, BUFFERS/N

DRIVE	symbol urządzenia, któremu chcesz zwiększyć rozmiar bufora, np. DF0:
BUFFERS	liczba przydzielanych buforów (w sektorach po 512 bajtów)

AddBuffers DF0: 10	Dodaje 10 * 512 bajtów do bufora stacji DF0:
AddBuffers DH0: 30	Dodaje 30 * 512 bajtów do bufora twardego dysku DH0:
AddBuffers DF1: -5	Redukuje rozmiar bufora stacji DF2: o 5 * 512 bajtów.
AddBuffers DH1:	Podaje aktualny rozmiar bufora dysku DH1:

Alias

Przypisuje nazwę dla komendy. Alias'y stosuje się zazwyczaj do utworzenia skróconej formy dla złożonego polecenia.

Alias NAME, STRING/F

NAME	przypisywana nazwa
STRING	rzeczywiste wyrażenie

Alias DodajBufor "AddBuffers DF0: 10"
Ustawia alias DodajBufor na komendę "AddBuffers DF0: 10". Od tego momentu wpisanie DodajBufor będzie równoznaczne z wydaniem polecenia: AddBuffers DF0: 10

Alias FormatDF0 "Format DRIVE DF0: NAME pusta FFS QUIET"
Od tego momentu użycie FormatDF0 spowoduje wykonanie komendy Format z tymi argumentami.

Alias
Wyświetla listę wszystkich alias'ów.

Ask

Używane w skryptach (plikach wsadowych). Umożliwia zadanie pytania użytkownikowi oczekując odpowiedzi typu tak/nie (y/n). Następnie w zależności od wyniku ustawia odpowiedni warunek (WARN). Stosowana wraz z poleceniami IF, Else, EndIF do podjęcia dalszych działań określonych w pliku wsadowym. Zobacz również polecenia IF, Else i EndIF (przykłady).

Ask PROMPT/A

PROMPT	treść pytania, które zostanie zadane użytkownikowi
--------	--

```
Ask "Odpalić automatycznie Workbench'a [y/n]"
  IF WARN
    Echo "Ładuje Workbench'a i zamykam okno AmigaDOS.."
    LoadWB
    EndCLI
Else
  Echo "Pozostaję w okienku AmigaDOS.."
  Echo ""
EndIF
```

Assign

Przypisuje urządzenie logiczne konkretnemu katalogowi.

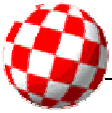
```
assign urządzenie katalog
assign gra_dysk4: WORK:gra/
```

Avail

Wyświetla ilość wolnej pamięci w bajtach.

Avail CHIP/S, FAST/S, TOTAL/S, FLUSH/S

CHIP	podaje ilość wolnej pamięci typu Chip
FAST	podaje ilość wolnej pamięci typu Fast
TOTAL	podaje ilość całkowitej wolnej pamięci



Avail FAST Wyświetla ilość wolnej pamięci Fast.
Avail CHIP TOTAL Wyświetli ilość wolnej pamięci Chip oraz wolnej całkowitej (Chip+Fast).
Avail Użycie samego rozkazu podaje rozmiar wolnej i zajętej oraz całkowitej pamięci jak i również największy, możliwy do zaalokowania blok dla wszystkich typów pamięci.

BindDrivers

Laduje sterowniki wszystkich urządzeń podłączonych do Amigi i ustala konfigurację systemu. Polecenie to umieszcza się zwykle w sekwencji startowej dysku (SYS:S/Startup-Sequence), ponieważ powinno być wykonane podczas uruchamiania systemu.

BindDrivers

Break

Ustawia kombinację klawiszy, którą można przerwać uruchomiony proces (patrz polecenie Status). Domyślna kombinacja to CTRL-C.

Break PROCESS/A/N, ALL/S, C/S, D/S, E/S, F/S

PROCESS	numer procesu, dla którego chcesz ustawić kombinację klawiszy
ALL	ustawia wszystkie możliwe kombinacje z klawiszem CTRL, czyli CTRL-C/D/E/F
C	ustawia kombinację CTRL-C
D	ustawia kombinację CTRL-D
E	ustawia kombinację CTRL-E
F	ustawia kombinację CTRL-F

Break 3 D Ustawia kombinację CTRL-D dla procesu trzeciego.

Break 1 ALL Ustawia wszystkie kombinacje dla procesu pierwszego.

CD

Zmienia bieżący katalog.

CD DIR

DIR	nowy katalog
-----	--------------

CD SYS: Przechodzi do głównego katalogu dysku, z którego został uruchomiony system.

CD SYS:Devs/Monitors Przechodzi do katalogu Devs/Monitors na dysku systemowym.

CD DF0:Gry/Arcanoid Ustawia jako bieżący katalog Gry/Arcanoid na dyskietce w stacji DF0:.

CD RAM: Przechodzi na Ram Dysk.

CD Wyświetla bieżący katalog.

ChangeTaskPri

Zmienia priorytet wykonania określonego zadania (patrz polecenie Status). W praktyce stosuje się priorytety z zakresu od -10 do +10. Zbyt niski może zablokować to zadanie. Zbyt wysoki może zablokować system.

ChangeTaskPri PRI=PRIORITY/A/N, PROCESS/K/N

PRI	nadawany priorytet z zakresu od -127 do +128
PROCESS	numer uruchomionego procesu (programu), któremu chcesz zmienić priorytet

ChangeTaskPri 5 PROCESS 2 Nada priorytet wykonania 5 procesowi drugiemu.

ChangeTaskPri 10 Nadaje wartość 10 priorytetu wszystkim uruchamianym programom.

ConClip

Aktywuje tzw. schowek, umożliwiając kopiowanie zawartości jednego okna CLI/Shell do drugiego. Polecenie to umieszcza się zwykle w sekwencji startowej na dysku (SYS:S/Startup-Sequence).

ConClip CLIPUNIT=UNIT/N, OFF/S

Copy

Tworzy kopię pliku lub katalogu o podanej nazwie.

Copy FROM/M, TO/A, ALL/S, QUIET/S, BUF=BUFFER/K/N, CLONE/S, DATES/S, NOPRO/S, COM/S, NOREQ/S

FROM	ścieżka źródłowa i nazwa pliku do skopiowania
------	---



TO	ścieżka docelowa i nazwa skopiowanego pliku
ALL	kopiuje również podkatalogi z zawartością
QUIET	nie pokazuje przebiegu operacji
CLONE	zachowuje datę i czas oryginalnego pliku

Copy DF0:plik1.txt DF0:plik1.bak	Wykona kopię pliku plik1.txt pod zmienioną nazwą plik1.bak na dyskietce w stacji DF0:.
Copy DF0: DH0:Dane	Skopiuje zawartość głównego katalogu z dyskietki znajdującej się w stacji DF0: do katalogu Dane na dysku twardym DH0: (w przypadku braku katalogu docelowego - zostanie on utworzony).
Copy DF0: DH0:Dane ALL	Skopiuje całą zawartość dyskietki w stacji DF0: do katalogu DH0:Dane.
Copy TO RAM: ALL QUIET	Skopiuje całą zawartość bieżącego katalogu na Ram Dysk nie pokazując przebiegu operacji.
Copy SYS:Dane/Grafika/hum#? DF0:Gfx/Humor	Skopiuje wszystkie pliki zaczynające się do liter "hum" z lokalizacji SYS:Dane/Grafika do katalogu DF0:Gfx/Humor.

CPU

Rozpoznaje zainstalowany w komputerze mikroprocesor i dopasowuje do niego pracę systemu.

Cpu

Date

Wyświetla lub zmienia aktualną datę i czas systemowy. W przypadku emulacji zmiana daty nie ma największego sensu. Po resecie Amigi emulator nastawi zegar Amigi zgodnie z ustawieniami BIOS'u.

Date DAY, DATE, TIME, TO=VER/K

DATE - DD-MMM-YY	format daty:	DD	dwucyfrowy numer dnia w miesiącu,
		MMM	trzyliterowy skrót nazwy miesiąca (dla angielskiej lokalizacji systemu to kolejno: Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec),
		YY	rok podawany w formacie dwucyfrowym
TIME - HH:MM:SS	format czasu:	HH	godzina,
		MM	minuta,
		SS	sekunda
TO	zapisuje datę i czas do pliku		

Date	Pokazuje aktualną datę i czas systemowy.
Date DATE 10-Feb-05	Zmienia aktualną datę na 10 luty 05 (2005).
Date TIME 12:21:34	Zmienia aktualny czas systemowy na podany.
Date DATE 09-Aug-34 TIME 05:59:01	Zmienia zarówno aktualną datę jak i czas systemowy.
Date TO DF0:data	Utworzy plik tekstowy "data" na dyskietce w stacji DF0: i do niego zapisze aktualną datę i czas systemowy.
Date DATE 09-Aug-34 TIME 05:59:01 TO RAM:aktualna-data	Utworzy plik "aktualna-data" na Ram Dysku i zapisze do niego podaną datę i czas.

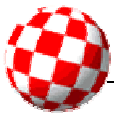
Delete

Kasuje pliki lub katalogi o podanej nazwie.

Delete FILE/M/A, ALL/S, QUIET/S, FORCE/S

FILE	nazwa pliku lub katalogu do usunięcia
ALL	usuwa całą zawartość katalogu wraz z nim
QUIET	nie pokazuje przebiegu operacji
FORCE	wymusza usunięcie również plików bez ustawionego atrybutu kasowalności (deletable)

Delete DF0:readme.txt	Usuwa plik readme.txt z katalogu głównego dyskietki w stacji DF0:
Delete Workbench:Pusty	Usuwa pusty katalog o nazwie "Pusty" z dysku Workbench'a.
Delete SYS:C ALL	Usuwa całą zawartość katalogu C wraz z nim na dysku systemowym.
Delete SYS:C ALL FORCE	To samo lecz wymusza usunięcie również plików bez atrybutu "Deletable".
Delete SYS:Dane/muzyka/#?polo QUIET	Usuwa wszystkie pliki kończące się na "polo" z katalogu SYS:Dane/Muzyka nie pokazując przebiegu operacji.



Dir

Wyświetla posortowaną zawartość katalogów.

Dir DIR, OPT/K, ALL/S, DIRS/S, FILES/S, INTER/S

DIR	katalog, którego zawartość chcesz obejrzeć
FILES	wyświetla tylko pliki
DIRS	wyświetla tylko podkatalogi
ALL	wyświetla również zawartość podkatalogów

Dir DF0: Wyświetla zawartość głównego katalogu dyskiety w stacji DF0.
Dir SYS:Devs/Monitors FILES Wyświetla tylko pliki z katalogu SYS:Devs/Monitors.
Dir SYS:Devs/Monitors DIRS Wyświetla tylko podkatalogi z katalogu SYS:Devs/Monitors.
Dir SYS:Devs/Monitors ALL Wyświetla całą zawartość katalogu SYS:Devs/Monitors z podkatalogami wraz z ich zawartością.
Dir SYS:C/F#? Wyświetla wszystkie pliki zaczynające się na literę "F" z katalogu SYS:C.
Dir Wyświetla zawartość bieżącego katalogu.

DiskChange

Informuje system o zmianie dyskiety w stacji 5.25 cala (tylko w wypadku jej dołączenia). Jest to konieczne tylko podczas pracy bezpośrednio z AmigaDOS.

DiskChange napęd:

DiskCopy

Tworzy kopię dyskiety. Polecenie czyta i zapisuje dyskiety "po sektorach" - dlatego nie wymagany jest format AmigaDOS.

DiskCopy FROM/A, TO/A, NAME/A, NOVERIFY/S, MULTI/S

FROM	dyskieta źródłowa
TO	dyskieta docelowa
NAME	ustala etykietę dyskiety docelowej (wymagany format AmigaDOS)
NOVERIFY	brak weryfikacji poprawności zapisu
MULTI	wczytuje dyskiety do pamięci umożliwiając wykonanie kilku kopii

DiskCopy DF0: DF1: Kopiuje obraz dyskiety w stacji DF0: na dyskiety w stacji DF1:.
DiskCopy DF0: DF1: NOVERIFY Kopiuje obraz dyskiety w stacji DF0: na dyskiety w stacji DF1: bez weryfikacji.
DiskCopy DF0: DF0: Wczytuje obraz dyskiety w stacji DF0: do pamięci, poczym po zmianie dyskiety, zapisuje go na nową.
DiskCopy DF0: DF1: MULTI Wczytuje obraz dyskiety w stacji DF0: do pamięci, poczym umożliwia wykonanie wielu jego kopii na dyskiety wkładane do stacji DF1:.
DiskCopy DF0: DF1: NAME KopiaDanych Kopiuje obraz dyskiety w stacji DF0: na dyskiety w stacji DF1: i ustala etykietę dyskiety docelowej na "KopiaDanych".

Echo

Wyświetla podany w parametrze tekst. Polecenie stosowane głównie w skryptach (plikach wsadowych). Zobacz także przykłady poleceń Ask, IF, Else oraz EndF.

Echo STRING/M, NOLINE/S, FIRST/K/N, LEN/K/N, TO/K

STRING	tekst do wyświetlenia znajdujący się między znakami cudzysłowiu
NOLINE	nie przechodzi do nowej linii po wyświetleniu tekstu
FIRST	numer znaku w tekście, od którego będzie wyświetlany
LEN	długość tekstu w znakach do wyświetlenia
TO	zapisuje tekst do pliku

Echo "To jest tekst!" Wyświetla ten napis i przechodzi do nowej linii.
Echo "To jest tekst!" NOLINE Wyświetla ten napis bez przejścia do nowej linii.
Echo "To jest tekst!" TO DH0:tekst.txt Tworzy plik DH0:tekst.txt kieruje do niego podany tekst.
Echo "Agnieszka Ferdka Genowefa" FIRST 11 LEN 6 Wyświetli tekst zaczynając od 11 znaku 6 kolejnych liter - czyli napis "Ferdka".

ED

Wywołanie wewnętrznego edytora tekstu.

ED nazwa

nazwa	uruchamia edytor tekstu. Wpisanie ED bez nazwy nie uruchomi edytora, należy w tedy zastosować ED cyfra
-------	--

EDIT

Wywołanie systemowego liniowego edytora tekstu.

EDIT nazwa

nazwa	uruchamia edytor tekstu. Wpisanie ED bez nazwy nie uruchomi edytora, należy w tedy zastosować ED cyfra
-------	--

Else

Stosowane w bloku warunkowym z poleceniem IF w skryptach (plikach wsadowych). Gdy warunek po IF nie jest spełniony wykonywany jest blok Else zakończony instrukcją EndIF. Zobacz także polecenia Ask, IF, EndIF (przykłady).

Else

```
IF EXISTS DH1:Dokumenty/plan.txt
    Echo "Warunek spełniony - plik plan.txt istnieje - wczytuje do edytora.."
    ED DH1:Dokumenty/plan.txt
Else
    Echo "Nie ma takiego pliku - pomijam edycję"
EndIF
Echo "Ładuję Workbench'a i zamykam okno CLI"
LoadWB
EndShell
```

EndCLI (EndShell)

Zamyka okienko AmigaDOS (kończy proces shell'a). W systemach od 2.0 w górę można stosować również EndShell, które jest alias'em na polecenie EndCLI.

EndCLI
EndShell

EndCLI	Dla systemów w wersjach 1.1/1.2/1.3.
EndShell	Dla systemów nowszych (od 2.0+).

EndIF

Kończy blok instrukcji warunkowych IF.. (Else..) EndIF w skrypcie (pliku wsadowym).

EndIF

```
IF NOT EXISTS DF0:Taurus
    Echo "Katalog nie istnieje - tworzę katalog.."
    MakeDir DF0:Taurus
EndIF
CD DF0:Taurus
```

Eval

Wykonuje proste działanie arytmetyczne i podaje wynik.

Eval VALUE1/A, OP, VALUE2/M, TO/K, LFORMAT/K

VALUE	liczby poddawane operacjom arytmetycznym
OP	operacja wykonywana na liczbach ("+" - dodawanie, "-" - odejmowanie, "*" - mnożenie, "/" - dzielenie)
TO	zapisuje wynik do pliku

Eval 2+3	Oblicza działanie "2+3" i wyświetla jego wynik - w tym przypadku 5.
Eval 2+3-1	Wyświetla wynik obliczonego działania (wynik 4).
Eval 2+3-1*2	Wyświetla wynik obliczonego działania (wynik 8).
Eval 2+3-1*2/4	Wyświetla wynik obliczonego działania (wynik 2).
Eval 2+3-1*2/4 TO RAM:wynik.txt	Oblicza wyrażenie i zapisuje wynik do pliku "wynik.txt" na Ram Dysku.



Execute

Wykonuje polecenia zawarte w pliku wsadowym (skrypcie).

Execute FILE/A

FILE skrypt (plik wsadowy) przeznaczony do wykonania

Execute DH0:S/Startup-Sequence Przetwarza ponownie sekwencję startową na dysku z Workbench'em.

Execute DF0:S/User-Startup Przetwarza rozkazy zawarte w pliku User-Startup w katalogu S dyskiety w stacji DF0:.

FailAt

Ustala granicę błędu (kod) dla wykonywanych skryptów (plików wsadowych).

FailAt RCLIM/N

RCLIM kod błędu, po przekroczeniu którego wykonywanie skryptu jest przerywane:

5	ostrzeżenie o możliwości wystąpienia błędu,
10	instrukcja nie wykonana do końca,
20	wykonanie instrukcji jest niemożliwe np. z powodu błędnych argumentów

FailAt Wyświetla aktualną wartość granicy błędu.

FailAt 5 Wykonywanie skryptów jest przerywane już przy ostrzeżeniu o możliwości wystąpienia błędu.

FailAt 10 Wykonywanie skryptów jest przerywane gdy jakaś instrukcja w nim zawarta nie wykona się do końca.

FailAt 20 Ustala kod (granicę) błędu na wartość 20. Wykonywanie skryptów jest przerywane dopiero w przypadku niemożności wykonania jakiegoś polecenia zawartego w skrypcie.

Fault

Wyświetla komunikat błędu o podanym kodzie.

Fault /N/M:

N - kod błędu, którego komunikat ma być wyświetlony

Fault 121 Wyświetla komunikat błędu o kodzie 121 - czyli "**File is not executable**".

Filenote

Wstawia komentarz do pliku o długości do 80 znaków (aby zobaczyć komentarze plików patrz polecenie List).

Filenote FILE/A, COMMENT, ALL/S, QUIET/S

FILE plik, któremu chcesz wstawić komentarz
COMMENT treść komentarza do 80 znaków (brak treści - czyści pole komentarza w pliku)
ALL wstawia komentarz do wszystkich plików w podanym katalogu
QUIET nie pokazuje przebiegu operacji

Filenote DF0:S/Startup-Sequence "To jest Sekwencja startowa"

Wstawi komentarz o treści "To jest Sekwencja startowa" do pliku Startup-Sequence w katalogu S dyskiety w stacji DF0:.

Filenote DF0:S/Startup-Sequence Usuwa komentarz z pliku DF0:S/Startup-Sequence.

Filenote DH0:C "Polecenie AmigaDOS" ALL

Wstawi komentarz o treści "Polecenie AmigaDOS" do wszystkich plików znajdujących się w katalogu C dysku twardego DH0:.

Filenote DH0:C "Polecenie AmigaDOS" ALL QUIET

Działa jak wyżej lecz nie pokazuje przebiegu operacji.

Filenote DH0:C ALL

Wyczyści pole komentarza we wszystkich plikach z katalogu C dysku DH0:.

Filenote DF0:AmigaE/Src/m#?.e "To są pliki na litetę M"

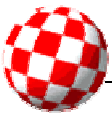
Wstawi komentarz o treści "To są pliki na litetę M" do plików zaczynających się od litery M w katalogu DF0:AmigaE/Src.

Format

Formatuje dyski lub dyskietki. Tworzy system plików OFS lub FFS zgodny z AmigaDOS na nośniku danych. Jeśli na nośniku znajdują się jakiegokolwiek dane zostaną one usunięte. W przypadku emulacji dysku twardego - działa tylko z plikiem .HDF.

Format DEVICE=DRIVE/K/A, NAME/K/A, OFS/S, FFS/S, INTL=INTERNATIONAL/S, NOINTL=NOINTERNATIONAL/S, DIRCACHE/S, NODIRCACHE/S, NOICONS/S, QUICK/S

DRIVE symbol nośnika przeznaczonego do sformatowania



NAME	nadawana nośnikowi etykieta
OFS	tworzy system plików OFS (Old File System) na nośniku
FFS	tworzy system plików FFS (Fast File System) na nośniku
NOICONS	nie zakłada katalogu Trashcan (kosz na śmieci) na nośniku
QUICK	tw. szybkie formatowanie (czyści tylko tablicę alokacji plików), działa z już sformatowanymi nośnikami. Istnieje możliwość odzyskania danych.
DIRCACHE	alokuje miejsce na bufor (cache) katalogów - szybszy będzie ich późniejszy odczyt

Format DRIVE DF0: NAME pusta OFS Sformatuje dyskietkę w stacji DF0: w systemie plików OFS, nada jej etykieta "pusta" i założy katalog "Trashcan" (kosz na śmieci).

Format DRIVE DF1: NAME "Dysk na dane #1" FFS NOICONS Utworzy system plików FFS na dyskietce w stacji DF1: nadając jej etykieta "Dysk na dane #1", nie zakłada "śmietnika".

Format DRIVE DH0: NAME Workbench FFS DIRCACHE NOICONS Sformatuje dysk twardego DH0:; nada mu etykieta "Workbench", utworzy bufor katalogów i nie założy "śmietnika".

Format DRIVE DH0: NAME Programs FFS DIRCACHE NOICONS QUICK Szybkie formatowanie dysku twardego DH0 - czyści tylko "tablicę alokacji plików". W ten sposób wszystkie dane dysku stają się niewidoczne dla systemu Amigi. Istnieje jednak możliwość ich odzyskania za pomocą programów "Fix Disk" czy "DiskSalvage2".

Get

Podaje wartość lokalnej zmiennej środowiskowej (występuje tylko w jednym procesie AmigaDOS) utworzonej wcześniej komendą Set.

Get NAME/A

NAME nazwa lokalnej zmiennej środowiskowej, której wartość chcesz obejrzeć

Get Process Wyświetli wartość zmiennej "Process".

GetEnv

Podaje wartość globalnej zmiennej środowiskowej utworzonej wcześniej komendą SetEnv. Takimi zmiennymi globalnymi (czyli dostępnymi dla wszystkich procesów) są np. tworzone automatycznie przy starcie systemu : zmienna "Workbench" oraz "Kickstart", a ich wartościami są w tym przypadku numery wersji tychże składników systemu Amigi.

GetEnv NAME/A

NAME nazwa globalnej zmiennej środowiskowej, której wartość chcesz otrzymać

GetEnv Workbench Wyświetli wartość tej zmiennej, czyli w tym przypadku numer wersji zainstalowanego Workbench-a.

GetEnv Kickstart Podobnie jak wyżej, lecz odnosi się do Kickstart-u.

IconX

Umożliwia uruchomienie pliku rozkazowego (skryptu) za pomocą ikony z poziomu Workbench-a. Powinna być wpisana na początku sekwencji rozkazów przeznaczonych do uruchamiania w ten sposób.

Iconx

IF

Rozpoczyna blok instrukcji warunkowych w skrypcie (pliku wsadowym). Jeżeli warunek po IF jest spełniony wykonywany jest ciąg poleceń zakończony instrukcją EndIF (lub Else). Zobacz również przykłady poleceń Ask, Else i EndIF.

IF NOT/S, WARN/S, ERROR/S, FAIL/S, EQ/K, GT/K, GE/K VAL/S, EXISTS/S

NOT negacja (neguje wynik porównania), np. NOT WARN, NOT EXISTS, NOT FAIL
WARN czy instrukcja ostrzegła o możliwości wystąpienia błędu
ERROR czy instrukcja nie wykonała się do końca
FAIL czy instrukcja nie wykonała się wcale
EXIST czy plik istnieje

Ask "Czy jesteś pełnoletni [y/n]"

IF WARN

Echo "Odpowiedziałeś [y]"

Else

Echo "Odpowiedziałeś [n]"

EndIF



```
Ask "Czy jesteś pewny [y/n]"
  IF NOT WARN
    Echo "Odpowiedziałeś [n]"
  Else
    Echo "Odpowiedziałeś [y]"
  EndIF

Echo "Wyłączam automatyczne przerywanie skryptów nawet w przypadku wystąpienia błędu o kodzie 20
(nieemożność wykonania polecenia)"
FailAt 21
Echo "Sam zajmę się obsługą błędów"
Lzx -a -e -r -x RAM:bla.lzx #?
  IF FAIL
    Echo "Polecenie Lzx niewykonane (złe argumenty lub brak polecenia)"
  EndIF
```

Info

Wyświetla informacje o podłączonych napędach.

Info DEVICE

DEVICE symbol napędu, o którym chcesz zacerpnąć informacji

Info DF0: Podaje szczegółowe informacje dotyczące dyskietki znajdującej się w stacji DF0. W kolejności są to pojemność, zajętość i wolny obszar w blokach, zapelnienie w procentach, liczba odczytanych błędów, status (Read/Write) oraz etykieta.

Info Wyświetli informacje o wszystkich napędach i zamontowanych urządzeniach.

Install

Instaluje, sprawdza lub usuwa bootblock na dyskietce. Polecenie stosowane po komendzie Format celem utworzenia boot'owalnej (startującej) dyskietki. Uwaga! Polecenie stosować można również w celu zniszczenia wirusa bootblock-owego - zostanie on nadpisany, a tym samym usunięty (tylko dla dyskietek używających formatu DOS).

Install DRIVE/A, NOBOOT/S, CHECK/S, FFS/S

DRIVE symbol stacji dysków, w której znajduje się dyskietka przeznaczona do operacji na bootblock-u
NOBOOT usuwa bootblock (dla dysku przeznaczonego na dane)
CHECK sprawdza istnienie bootblock-u i podaje o nim informacje
FFS instaluje bootblock na dyskietce sformatowanej w systemie plików FFS

Install DF0: Instaluje bootblock OFS na dyskietce w stacji DF0: umożliwiając jej wystartowanie na kickstart-ach 1.3 lub wyższych.

Install DF0: FFS Instaluje bootblock dla systemu plików FFS na dyskietce w stacji DF1: umożliwiając jej wystartowanie na kickstart-ach od 2.0 w górę.

Install DF0: NOBOOT Usuwa bootblock z dyskietki w stacji DF0: (nie można z niej wystartować - np. dla dysku na dane).

Install DF0: CHECK Sprawdza dyskietkę pod kątem obecności bootblock-u i wyświetla odpowiednie informacje.

Iprefs

Ustawia konfigurację Workbench'a według konfiguracji zapisanej w katalogu ENV-ARC. Konfiguracja zapisywana jest w katalogu przez programy z katalogu PREFERENCES. Tymi też programami może ona zostać zmieniona.

Iprefs

Join

Połączy se sobą kilka plików w jeden.

Join FILE/M/A, AS=TO/K/A

FILE pliki do połączenia
AS nazwa pliku powstałego z plików połączonych

Join DF0:text1.txt DF1:Text2.txt RAM:text3.txt AS DH0:Dane/Text/praca.txt

Połączy z różnych lokalizacji pliki text1.txt, text2.txt i text3.txt w jeden plik praca.txt w katalogu DH0:Dane/Text.



Lab

Ustala etykietę dla skoku wykonywanego po instrukcji SKIP.

Lab nazwa

List

Wyświetla informacje o podanym katalogu i zawartych w nim plikach.

List nazwa_katalogu

LoadWB

Rozpoczyna pracę Workbencha.

LoadWB

Lock

Software'owa metoda zabezpieczenia nośnika przed zapisem. Działa do ponownego restartu lub odbezpieczenia tą samą komendą. Stosowana zarówno dla dyskietek jak i dla dysku twardego. Uwaga! Dyskietka musi być fizycznie odbezpieczona. Dla dysku twardego działa tylko z plikiem .HDF.

Lock DRIVE/A, ON/S, OFF/S, PASSKEY

DRIVE	napęd, który chcesz zabezpieczyć przed zapisem
ON	zakłada blokadę zapisu
OFF	usuwa blokadę zapisu

Lock DF0: ON Nakłada blokadę zapisu na odbezpieczoną dyskietkę w stacji DF0:.

Lock DF0: OFF Usuwa blokadę zapisu z odbezpieczonej dyskietki w stacji DF0:.

Lock DH0: ON Nakłada blokadę zapisu na dysk twardego DH0: (tylko format .HDF).

Lock DH0: OFF Usuwa blokadę zapisu z dysku twardego DH0: (tylko format .HDF).

MakeDir

Tworzy nowy katalog w podanej lokalizacji.

MakeDir NAME/M

NAME	nazwa katalogu do utworzenia
------	------------------------------

MakeDir DF0:Programy Utworzy nowy katalog "Programy" na dyskietce w stacji DF0:.

MakeDir DH0:Franko DH0:Franko/Dane DH0:Franko/Dane/GFX

Tworzy kilka katalogów jednocześnie. W tym przypadku utworzy katalog "Franko" na dysku twardym oraz w nim podkatalogi Dane i GFX.

MakeLink

Tworzy połączenie pomiędzy plikami. Umożliwia komunikację dwóch plików np. dwóch skryptów AmigaDos.

MakeLink nazwa1 nazwa2

Mount

Informuje Amigę o przyłączeniu nowego urządzenia do systemu. Urządzenie to musi być jednak opisane w pliku MOUNTLIST znajdującym się w katalogu DEVS i mieć swój sterownik, również w tym katalogu.

Mount urządzenie

NewCLI (NewShell)

Otwiera nowe okno AmigaDOS. W systemach od 2.0 w górę można stosować również NewShell, które jest alias'em na polecenie NewCLI.

NewCLI
NewShell

NewCLI Dla systemów w wersjach 1.1/1.2/1.3.

NewShell Dla systemów nowszych (od 2.0+).



Path

Sprawdza, kasuje lub dodaje katalogi do aktualnej ścieżki dostępu do plików.

Path ADD ścieżka / RESET

ADD ścieżka	Dodaje ścieżkę do ustawień globalnych
RESET	Kasuje ścieżki

Prompt

Zmienia wskaźnik gotowości AmigaDOS.

Prompt wzór

Protect

Ustala rodzaj zabezpieczenia dla zbioru.

PROTECT nazwa_zb +/- bit

nazwa_zb	nazwa zabezpieczanego zbioru
+/-	wyłączenie lub włączenie danego bitu
bit	określenie bitu zabezpieczenia
R	bit odczytu pliku
W	bit zapisu do pliku
E	bit określający, czy plik jest plikiem wykonywalnym
D	bit określający, czy plik może zostać skasowany

Relabel

Zmienia tytuł dysku - nadany podczas formatowania.

Relabel napęd: nazwa

RemRAD

Usuwa z pamięci, założone komendą "Mount RAD:", urządzenie RAD: (nieresetowalny Ram Dysk).

RemRAD DEVICE, FORCE/S

FORCE	wymusza usunięcie RAD Dysku nawet jeśli znajdują się na nim jakieś dane
-------	---

RemRAD	Usunie z pamięci tylko pusty RAD Dysk.
RemRAD FORCE	Usunie z pamięci pusty lub zawierający dane RAD Dysk.

Resident

Odczytuje z katalogu C rozkaz AmigaDos i umieszcza go w pamięci jako polecenie rezydentne. Dzięki temu nie musi on być odczytywany z dysku.

Resident [rozkaz]

Run

Wykonuje podane polecenie jako jeden z procesów AmigaDOS.

Run polecenie

Quit

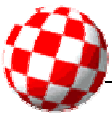
Przerywa pracę procesu generując błąd o podanym kodzie.

Quit n
n Kod błędu

Search

Poszukuje łańcucha znaków w podanym pliku lub katalogu.

Search napęd: nazwa



Set

Definiuje lokalną zmienną środowiskową (dostępną dla jednego procesu AmigaDOS) oraz przypisuje jej wartość w postaci ciągu znaków.

Set NAME, STRING/F

NAME nazwa definiowanej zmiennej
STRING przypisywana zmiennej wartość w postaci ciągu znaków

Set MyDocs "DH0:Dane/Text"
Dir \$MyDocs

Zdefiniuje nową lokalną zmienną środowiskową o nazwie "MyDocs" oraz przypisze jej wartość "DH0:Dane/Text" - ścieżka do katalogu. Rozkaz Dir wyświetli zawartość katalogu "DH0:Dane/Text".

Set Image "DH0:Dane/Obrazki/tapeta.iff"
PPShow \$Image

Zdefiniuje lokalną zmienną środowiskową "Image" oraz przypisze jej wartość "DH0:Dane/Obrazki/tapeta.iff" - ścieżka dostępu do tego pliku.

Set Wyświetli listę wszystkich zdefiniowanych lokalnych zmiennych środowiskowych.

SetClock

Ustawia lub odczytuje stan zegara czasu rzeczywistego.

SetClock load/save/reset

load ustawia czas systemowy zgodnie z zegarem czasu rzeczywistego
save ustawia zegar czasu rzeczywistego zgodnie z zegarem systemowym
reset kasuje aktualne ustawienie zegara czasu rzeczywistego

SetDate

Zmienia datę i czas utworzenia pliku.

SetDate FILE/A, WEEKDAY, DATE, TIME, ALL/S

FILE plik, któremu chcesz zmienić datę i/lub czas utworzenia
DATE - DD-MMM-YY format daty:
DD - dwucyfrowy numer dnia w miesiącu,
MMM - trzyliterowy skrót nazwy miesiąca (dla angielskiej lokalizacji systemu to kolejno: Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec),
YY - rok podawany w formacie dwucyfrowym
TIME - HH:MM:SS format czasu:
HH - godzina,
MM - minuta,
SS - sekunda

Date C:Dir Zmienia datę i czas utworzenia pliku C:Dir na aktualny.
Date C:Dir DATE 12-May-2001 Zmienia datę utworzenia pliku C:Dir na 12-may-2001 zaś czas na aktualny.
Date C:Dir TIME 23:11:00 Zmienia czas utworzenia pliku C:Dir na 23:11:00 zaś datę na aktualną.
Date C:Dir DATE 11-Apr-03 TIME 12:12:43 Zmienia datę i czas utworzenia pliku C:Dir na podane.

SetEnv

Definiuje globalną zmienną środowiskową (dostępną dla wszystkich procesów AmigaDOS-u) oraz przypisuje jej dowolną wartość w postaci ciągu znaków. Taką zmienną można wykorzystać jako parametr komendy podając znak \$ przed jej nazwą, np. PPShow \$Image.

SetEnv NAME, STRING/F

NAME nazwa definiowanej zmiennej
STRING przypisywana zmiennej wartość w postaci ciągu znaków

SetEnv AdminName "DeathMieszko"

Zdefiniuje nową globalną zmienną środowiskową o nazwie "AdminName" oraz przypisze jej wartość "DeathMieszko" - tu moja xywka.

SetEnv Image "DH0:Dane/Obrazki/tapeta.iff"

Zdefiniuje globalną zmienną środowiskową "Image" oraz przypisze jej wartość "DH0:Dane/Obrazki/tapeta.iff" - ścieżka dostępu do tego pliku.

SetEnv

Wyświetli listę wszystkich zdefiniowanych globalnych zmiennych środowiskowych.



SetFont

Ustala czcionkę używaną przez SHELL.

SETFONT czcionka wielkość [ITALIC] [BOLD] [UNDERLINE]

czcionka	nazwa czcionki
wielkość	rozmiar czcionki
[krój]	krój czcionki: <i>ITALIC</i> , BOLD , <u>UNDERLINE</u>

SetPatch

Likwiduje błędy wynikające z różnic między zainstalowanym kickstart-em a poprzednimi jego wersjami. Poszczególne wersje kickstart'ów różnią się od siebie kilkoma parametrami. W związku z czym niektóre programy napisane dla poprzednich wersji mogą nie działać prawidłowo z zainstalowanym kickstart'em. Polecenie SetPatch usuwa te różnice. Powinno być umieszczone na początku sekwencji startowej dysku systemowego. Uwaga! Dla poszczególnych wersji kickstart'u i Workbench'a dlań przeznaczonego istnieją odpowiednie wersje polecenia SetPatch.

SetPatch QUIET/S, NOCACHE/S, REVERSE/S, NOAGA/S

QUIET - nie wyświetla listy nakładanych na kickstart łań (patch-y)

SetPatch Łąca zainstalowany kickstart, czyniąc go kompatybilnym z poprzednimi jego wersjami, wyświetlając listę nakładanych łań.

SetPatch QUIET Nie wyświetla żadnych informacji o swoim działaniu.

Sort

Sortuje alfabetycznie linie pliku tekstowego. Sortuje według pierwszych liter każdej linii. Jeżeli pierwsze litery są takie same, to o kolejności decyduje druga litera i tak dalej. Linie rozpoczynające się cyframi znajdują się na początku posortowanego tekstu.

Sort nazwa1 nazwa2

nazwa1	plik do sortowania
nazwa2	plik posortowany

Stack

Wyświetla lub ustawia rozmiar stosu. Stos jest rejestrem procesora, na którym odkładane są dane wysłane przez kod programu.

Stack [rozmiar]

Status

Wyświetla informacje o uruchomionych procesach.

Status PROCESS/N, FULL/S, TCB/S, CLI=ALL/S, COM=COMMAND/K

PROCESS	numer procesu, o którym chcesz zasięgnąć informacji
FULL	wyświetla wszystkie możliwe informacje o procesie
TCB	wyświetla informacje o rozmiarze stosu przydzielonego procesowi, wielkości wektorów globalnych oraz priorytecie procesu
COM	wyświetla numer procesu uruchomionego programu podanego po tym słowie kluczowym

Status Wyświetla numery wszystkich uruchomionych procesów i ich nazwy (uruchomionych programów lub komend).

Status 1 TCB Wyświetla informacje o procesie 1 takie jak przydzielony rozmiar stosu, wielkość wektora globalnego i priorytet procesu.

Status 1 FULL Wyświetla również nazwę programu uruchomionego jako ten proces.

Status COM Filemster Podaje numer procesu uruchomionego programu Filemaster.

Status FULL Wyświetli wszystkie informacje o wszystkich uruchomionych procesach.



Type

Wyświetla zawartość podanego pliku.

Type FROM/A/M, TO/K, OPT/K, HEX/S, NUMBER/S

FROM	plik, którego zawartość chcesz obejrzeć
TO	kieruje działanie komendy do podanego urządzenia (portu) lub pliku
HEX	pokazuje zawartość szesnastkowo
NUMBER	dodatkowo numeruje linie

Type S:Startup-Sequence

Wyświetla zawartość pliku tekstowego S:Startup-Sequence.

Type S:Startup-Sequence HEX

Pokazuje zawartość sekwencji startowej szesnastkowo.

Type S:Startup-Sequence HEX NUMBER

Dodatkowo numeruje linie przy wyświetlaniu zawartości.

Type DH1:Basic/Src/prog1.bas TO RAM:prog1.bas NUMBER

Zapisuje zawartość pliku tekstowego DH1:Basic/Src/prog1.bas do pliku RAM:prog1.bas dodając numerację linii.

Type C:Assign >SER:

Wysyła zawartość binarnego pliku C:Assign poprzez port szeregowy.

Type DH0:Dane/Text/Private/g#? >PRT:

Wysyła na drukarkę zawartość plików zaczynających się na literę "g" z katalogu DH0:Dane/Text/Private.

UnAlias

Usuwa przypisaną, za pomocą komendy Alias, nazwę złożonemu poleceniu (zobacz również rozkaz Alias).

UnAlias NAME

NAME	nazwa przypisana wcześniej komendą Alias
------	--

UnAlias DodajBufor

Usuwa tego Alias'a z listy wszystkich Alias'ów.

UnAlias

Wyświetla listę wszystkich alias'ów.

Version

Wyświetla numer wersji zainstalowanego kickstart'u i Workbench'a. Polecenie stosować można również do komend systemowych, bibliotek i sterowników oraz programów (tylko dla plików wykonywalnych zawierających w kodzie ciąg znaków \$VER:).

Version NAME, VERSION/N, REVISION/N, FILE/S, FULL/S, UNIT/N, INTERNAL/S, RES/S

NAME	nazwa pliku, którego numer wersji chcesz obejrzeć
FULL	wyświetla również datę utworzenia (kompilacji) pliku

Version

Podany bez parametrów wyświetla wersję kickstart'u i Workbench'a.

Version SYS:System/Format

Podaje wersję systemowej komendy Format - dla Workbench'a 3.1 jest to wersja 40.2.

Version SYS:System/Format FULL

Podaje również datę utworzenia komendy Format - (14.06.93).

Version LIBS:amigaguide.library

Wyświetla wersję biblioteki amigaguide znajdującej się w katalogu LIBS: dysku systemowego.

Version exec.library

Podaje numer wersji biblioteki exec znajdującej się w ROM'ie (kickstart) Amigi.

Version trackdisk.device

Wyświetla wersję sterownika stacji dysków zawartego w ROM'ie Amigi.

Wait

Zatrzymuje działanie komputera na podany czas.

Wait /N, SEC=SECS/S, MIN=MINS/S, UNTIL/K

N	czas trwania przerwy
SEC	oznacza, że podawany czas jest w sekundach
MIN	oznacza, że podawany czas jest w minutach
UNTIL	zawieszają działanie komputera do podanej po tym słowie kluczowym godziny w formacie HH:MM

Wait

Wait bez parametrów zawieszają działanie komputera na 1 sekundę.

Wait 10 SEC

Zatrzymuje działanie komputera na 10 sekund.

Wait 3 MIN

Zatrzymuje działanie komputera na 3 minuty.

Wait UNTIL 13:25

Zatrzymuje działanie komputera do godziny 13:25.



Which

Wyświetla pełną ścieżkę dostępu do pliku, pod warunkiem, że została uwzględniona przy rozkazie Path (domyślnie sprawdza tylko katalog SYS:C).

Which FILE/A, NORES/S, RES/S, ALL/S

FILE	nazwa pliku, do którego ścieżkę dostępu chcesz zobaczyć
NORES	określa, że plik nie jest rezydentny (na dysku)
RES	określa, że plik jest rezydentny (w pamięci)

Which Format Podaje pełną ścieżkę dostępu do komendy Format, czyli w tym przypadku SYS:System/Format.

Which MultiView NORES Podaje ścieżkę dostępu do pliku MultiView, pod warunkiem, że nie jest on rezydentny - SYS:Utilities/MultiView.

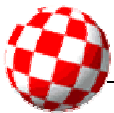
Which CD RES Wyświetli komunikat "Internal CD" ponieważ komenda CD jest plikiem rezydentnym (znajduje się w ROM'ie Amigi).

Which CD NORES Nie wyświetli żadnych informacji, gdyż nie został spełniony warunek (NORES - wymagany plik nierezydentny).

Why

Wyjaśnia, podając komunikat błędu, dlaczego ostatnio użyte polecenie nie zostało wykonane (o ile zwróciło kod błędu).

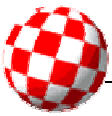
Why



Komunikaty błędów

Wszystkie ważniejsze komunikaty błędów i problemy, które mogą wystąpić podczas pracy z CLI/Shell:

- 103 - not enough memory available (Insufficient free store)**
Niewystarczająca ilość wolnej pamięci. Próba wykonania zadania wymagającego większą ilość wolnej pamięci niż jest aktualnie dostępna. Spróbuj pozamykać nieużywane okienka, programy i procesy, w tym w CLI w celu zwolnienia pamięci lub zrestartuj komputer.
- 104 - Task table full**
Tablica obsługi procesów AmigaDos jest pełna - zawiera 20 pozycji. Zakończ pracę zbędnych procesów aby zwolnić kilka pozycji w tablicy obsługi procesów
- 105 - Process table full**
Tabela procesów zapełniona. Zbyt dużo uruchomionych programów. Próbujesz uruchomić kolejny proces, którego system nie jest już w stanie zarejestrować. Pozamykaj nieużywane programy lub uruchom Amigę ponownie.
- 116 - Required argument missing**
Brak wymaganych argumentów. Próba wywołania komendy bez podania wymaganych przez nią argumentów lub nie podałeś ich wcale. Sprawdź czego wymaga komenda podając jako argument znak zapytania (np. "Assign ?").
- 117 - Value after keyword missing**
Brak parametru po słowie kluczowym. Zapomniałeś podać parametru po słowie kluczowych w argumentach komendy. Sprawdź czego wymaga komenda podając jako argument znak zapytania ("?") lub dopisz brakujący parametr po słowie kluczowym.
- 118 - Wrong number of arguments**
Niewłaściwa liczba argumentów przekazanych wywołanej komendzie. Prawdopodobnie podałeś ich za mało lub za dużo. Sprawdź czego wymaga komenda podając jako argument znak zapytania ("?").
- 120 - Argument line invalid or too long**
Zły lub za długi wiersz podanych argumentów. Podałeś zbyt długi ciąg argumentów po komendzie, w wyniku czego system nie jest w stanie ich zinterpretować. Sprawdź czego wymaga komenda podając jako argument znak zapytania ("?").
- 121 - File is not executable (File is not an object module)**
Plik nie jest wykonywalny. Podałeś nazwę pliku, który nie jest programem lub próbujesz uruchomić plik, który nie ma ustawionej odpowiedniej flagi - jest zabezpieczony przed wykonaniem. Sprawdź nazwę pliku lub ustaw odpowiednie znaczniki (executable) pliku komendą "Protect".
- 122 - Invalid resident library during load**
Uszkodzona biblioteka. System podczas ładowania bibliotek napotkał na uszkodzoną lub wystąpił błąd w obsłudze biblioteki. Być może dysk jest uszkodzony. Usuń uszkodzoną bibliotekę w katalogu Libs i zastąp poprawną z innej dyskietki (np. systemowej).
- 202 - Object is in use**
Object (plik) jest w użyciu. Prawdopodobnie próbujesz dokonać zapisu na pliku lub usunąć plik, który aktualnie jest wykorzystywany przez inne programy lub procesy. Zamknij procesy, usuń Assign lub pozamykaj otwarte okienka, a następnie usuń lub zapisz plik.
- 203 - Object already exists**
Object (plik) już istnieje. Próba utworzenia pliku lub katalogu o tej samej nazwie, który już istnieje lub próba przesunięcia ("Move") pliku w miejsce gdzie już on występuje. Użyj innej nazwy pliku lub skasuj najpierw plik "oryginalny".
- 204 - Directory not found**
Katalog nie znaleziony. System nie może odnaleźć podanego katalogu na dysku. Sprawdź poprawną nazwę katalogu lub włóż odpowiedni dysk.
- 205 - Object not found**
Object (plik) nie znaleziony. System nie może odnaleźć podanego w parametrze pliku lub obiektu na dysku. Sprawdź nazwę pliku lub włóż odpowiedni dysk.
- 206 - Invalid window description**
Złe parametry okna CLI/Shell. Podałeś złe parametry okienka AmigaDOS podczas otwierania go komendą "NewShell" lub "NewCLI". Sprawdź i popraw parametry rozkazu.



- 210 - Object name invalid Przekroczona dopuszczalna długość nazwy pliku - 31 znaków. Skróć lub zmień nazwę pliku.
- 212 - Object is not of required type Podany obiekt (plik) nie jest w wymaganym typie. Podałeś zły typ pliku przy wykonywaniu rozkazu. Na przykład użyłeś komendy "Type" (wyświetlającą zawartość pliku [tekstowego]) w stosunku do katalogu lub urządzenia. Sprawdź plik lub użyty rozkaz, czy możliwe jest wykonanie operacji jednego na drugim.
- 213 - Disk not validated Dysk nie jest poprawnie zapisany. Mapa zajętości dyskietki nie została uaktualniona. Prawdopodobnie wyjąłeś dyskietkę z napędu podczas zapisu na niej danych. Włóż odbezpieczony dysk do stacji i poczekaj aż system go "zwaliduje" (w systemach starszych niż 2.0 w katalogu L dyskietki musi się znajdować plik Disk-Validator). W przypadku niepowodzenia użyj programów typu FixDisk czy DiskSalvage2 (w ostateczności program DiskDoctor) w celu odzyskania danych z uszkodzonego nośnika.
- 214 - Disk is write-protected Dysk jest zabezpieczony przed zapisem. Próba zapisu danych na zabezpieczony nośnik. Odbezpiecz dyskietkę przed wykonaniem operacji zapisu.
- 215 - Rename across devices attempted Próba użycia polecenia "Rename" (zmieniającego nazwy plików) w kilku urządzeniach jednocześnie. Rozkaz "Rename" umożliwia zmianę nazwy pliku w obrębie jednego urządzenia. Skopiuj plik do właściwego urządzenia lub popraw argumenty polecenia "Rename".
- 216 - Directory not empty Katalog nie jest pusty. Próba usunięcia katalogu zawierającego dane. Usuń wszystkie dane z katalogu wraz z nim komendą "Delete nazwa_katalogu ALL".
- 218 - Device (or volume) is not mounted Nastąpiło odwołanie do urządzenia logicznego lub woluminu, które nie jest zamontowane. Sprawdź urządzenie (użyj komendy "Assign DEVICES"), zamontuj urządzenie komendą "Mount" lub włóż ponownie dysk do napędu.
- 220 - Comment is too long Zbyt długi komentarz do pliku. Przekroczyłeś dopuszczalną długość 80 znaków w polu FILENOTE pliku. Skróć tekst komentarza.
- 221 - Disk is full Dysk jest pełny. Wystąpił brak miejsca podczas wykonywania zapisu na tym nośniku. Usuń zbędne dane na dysku lub włóż nowy czysty dysk do napędu.
- 222 - Object is protected from deletion (File is protected from deletion) Object (plik) jest zabezpieczony przed usunięciem. Ustaw bit (deletable) pliku poleceniem "Protect".
- 223 - File is write protected (File is protected from writing) Plik jest zabezpieczony przed zapisem. Ustaw odpowiednie flagi (writable) pliku za pomocą komendy "Protect".
- 224 - File is read protected (File is protected from reading) Plik jest zabezpieczony przed odczytem. Użyj komendy Protect do ustawienia odpowiednich atrybutów (readable) pliku poleceniem Protect lub opcją **Information** w menu **Workbench**a (zależne od błędu: 222, 223, 224).
- 225 - Not a valid DOS disk (Not a DOS disk) System nie rozpoznał formatu dysku. W napędzie znajduje się dysk niesformatowany pod kontrolą AmigaDOS. Prawdopodobnie jest to dyskietka z grą używającą własnego formatu (tzw. "niedosowy" dysk). Włóż dysk do napędu używający formatu AmigaDOS.
- 226 - No disk in drive Nastąpiła próba odwołania się do napędu, w którym nie ma nośnika. Najczęściej jest to brak dyskietki w stacji. Włóż nośnik do napędu.

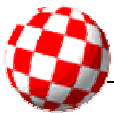


BŁĘDY GURU

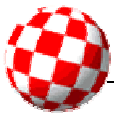
Numer	Typ	Podsystem	Informacje	
			Podstawowe	Dodatkowe
00000000	Przywracalny	CPU	Błąd ogólny	Bez wystąpienia błędu
00000001	Przywracalny	CPU	Błąd ogólny	Nieznane
00000002	Przywracalny	CPU	Błąd ogólny	Błąd sprzętowy - BUS
00000003	Przywracalny	CPU	Błąd ogólny	Nielegalny adres w pamięci
00000004	Przywracalny	CPU	Błąd ogólny	Nieznane
00000005	Przywracalny	CPU	Błąd ogólny	Dzielenie przez zero
00000006	Przywracalny	CPU	Błąd ogólny	Błąd rozkazu - CHK
00000007	Przywracalny	CPU	Błąd ogólny	Rozkaz - TRAPV
00000008	Przywracalny	CPU	Błąd ogólny	Przywilej naruszenia
00000009	Przywracalny	CPU	Błąd ogólny	Znak
00000010	Przywracalny	CPU	Błąd ogólny	Nieznane
...017				
00000018	Przywracalny	CPU	Błąd ogólny	Sztuczne przerwanie
00000019	Przywracalny	CPU	Błąd ogólny	Auto-wektorowa płaszczyzna 1 błędu przerwania
00000020	Przywracalny	CPU	Błąd ogólny	Nieznane
...029				
00000030	Przywracalny	CPU	Błąd ogólny	FPCP Rozgałęzienie za nie uporządkowanym stanem
00000031	Przywracalny	CPU	Błąd ogólny	FPCP Niedokładność wyniku
00000032	Przywracalny	CPU	Błąd ogólny	FPCP Dzielenie przez zero
00000033	Przywracalny	CPU	Błąd ogólny	FPCP Dolny bieg...(np. rzeki)
00000034	Przywracalny	CPU	Błąd ogólny	FPCP Błąd argumentu
00000035	Przywracalny	CPU	Błąd ogólny	FPCP Najście
00000036	Przywracalny	CPU	Błąd ogólny	FPCP Sygnał NAN
00000037	Przywracalny	CPU	Błąd ogólny	FPCP Nieskończony sygnał
00000038	Przywracalny	CPU	Błąd ogólny	PMMU Wstrzymanie
00000039	Przywracalny	CPU	Błąd ogólny	PMMU Nielegalne ustawienie
00000040	Przywracalny	CPU	Błąd ogólny	Nieznane
0000000A	Przywracalny	CPU	Błąd ogólny	Linia 1010 Emulator
0000000B	Przywracalny	CPU	Błąd ogólny	Linia 1111 Emulator
0000000C	Przywracalny	CPU	Błąd ogólny	Nieznane
0000000D	Przywracalny	CPU	Błąd ogólny	Koprocesor - protokół naruszenia
0000000E	Przywracalny	CPU	Błąd ogólny	Stos - błąd formatu
0000000F	Przywracalny	CPU	Błąd ogólny	Nieznane
0000001A	Przywracalny	CPU	Błąd ogólny	Auto-wektorowa płaszczyzna 2 błędu przerwania
0000001B	Przywracalny	CPU	Błąd ogólny	Auto-wektorowa płaszczyzna 3 błędu przerwania
0000001C	Przywracalny	CPU	Błąd ogólny	Auto-wektorowa płaszczyzna 4 błędu przerwania
0000001D	Przywracalny	CPU	Błąd ogólny	Auto-wektorowa płaszczyzna 5 błędu przerwania
0000001E	Przywracalny	CPU	Błąd ogólny	Auto-wektorowa płaszczyzna 6 błędu przerwania
0000001F	Przywracalny	CPU	Błąd ogólny	Auto-wektorowa płaszczyzna 7 błędu przerwania
002C93C0	Przywracalny	CPU	Nieznane	Nieznane
01000000	Przywracalny	exec.library	Błąd ogólny	Problem z exec.library
01000001	Przywracalny	exec.library	Błąd ogólny	68000 Wyjatek-wektor-suma kontrola
01000002	Przywracalny	exec.library	Błąd ogólny	Execbase - błąd sumy kontrolnej
01000003	Przywracalny	exec.library	Błąd ogólny	Library - błąd sumy kontrolnej
01000004	Przywracalny	exec.library	Błąd ogólny	Nieznane
01000005	Przywracalny	exec.library	Błąd ogólny	FreeMem: niszczące zapełnienie pamięci
01000006	Przywracalny	exec.library	Błąd ogólny	Brak pamięci na przerwanie
01000007	Przywracalny	exec.library	Błąd ogólny	InitStruct () jako APTR-źródło
01000008	Przywracalny	exec.library	Błąd ogólny	Jeden semafor jest w nielegalnym położeniu
01000009	Przywracalny	exec.library	Błąd ogólny	Zwolnienie wolnej pamięci
01000010	Przywracalny	exec.library	Błąd ogólny	Próba użycia zwiększonego MSG-semafora
0100000A	Przywracalny	exec.library	Błąd ogólny	Nielegalny 68K-zastrzeżone użycie
0100000B	Przywracalny	exec.library	Błąd ogólny	Zastosowanie jednego aktywnego IORequest
0100000C	Przywracalny	exec.library	Błąd ogólny	Sprawdzenie w pamięci listy negacji
0100000D	Przywracalny	exec.library	Błąd ogólny	IO do zamkniętego IORequest
0100000E	Przywracalny	exec.library	Błąd ogólny	Zamrożenie stosu do chwili powstania
0100000F	Przywracalny	exec.library	Błąd ogólny	Memory-Header nie znaleziony
10000000	Przywracalny	audio.device	Błąd ogólny	Problem z audio.device
11000000	Przywracalny	console.device	Błąd ogólny	Problem z console.device
12000000	Przywracalny	gameport.device	Błąd ogólny	Problem z gameport.device
13000000	Przywracalny	keyboard.device	Błąd ogólny	Problem z keyboard.device
14000000	Przywracalny	trackdisk.device	Błąd ogólny	Problem z trackdisk.device
15000000	Przywracalny	timer.device	Błąd ogólny	Problem z timer.device
22000000	Przywracalny	misc.resource	Błąd ogólny	Problem z misc.resource
22010000	Przywracalny	misc.resource	Brak pamięci	Nieznane
22020000	Przywracalny	misc.resource	Nie ustalono utworzonej biblioteki	Nieznane
22030000	Przywracalny	misc.resource	Nie ustalono otwartej biblioteki	Nieznane
22040000	Przywracalny	misc.resource	Nie ustalono otwartego urządzenia	Nieznane
22050000	Przywracalny	misc.resource	Nie ustalono otwartego zasobu (Resource)	Nieznane
22060000	Przywracalny	misc.resource	Błąd I/O	Nieznane



22070000	Przywracalny	misc.resource	Brak sygnału	Nieznane
22080000	Przywracalny	misc.resource	Falszywy parametr	Nieznane
22090000	Przywracalny	misc.resource	Ne ustalono zamknięcia biblioteki	Nieznane
80000000	Całkowicie nieczynny	CPU	Błąd ogólny	Bez wystąpienia błędu
80000001	Całkowicie nieczynny	CPU	Błąd ogólny	Nieznane
80000002	Całkowicie nieczynny	CPU	Błąd ogólny	Sprzętowe (wejście do pamięci)-BUS błąd
80000003	Całkowicie nieczynny	CPU	Błąd ogólny	Nielegalny adres pamięci
80000004	Całkowicie nieczynny	CPU	Błąd ogólny	Nielegalny rozkaz
80000005	Całkowicie nieczynny	CPU	Błąd ogólny	Dzielenie przez zero
80000006	Całkowicie nieczynny	CPU	Błąd ogólny	CHK-błąd rozkazu
80000007	Całkowicie nieczynny	CPU	Błąd ogólny	TRAPV-rozkaz
80000008	Całkowicie nieczynny	CPU	Błąd ogólny	Przywilej naruszenia
80000009	Całkowicie nieczynny	CPU	Błąd ogólny	Znak
80000010	Całkowicie nieczynny	CPU	Błąd ogólny	Nieznane
...017				
80000018	Całkowicie nieczynny	CPU	Błąd ogólny	Nieprawidłowe przerwanie
80000019	Całkowicie nieczynny	CPU	Błąd ogólny	Auto-wektorowa płaszczyzna 1 błędu przerwania
80000020	Całkowicie nieczynny	CPU	Błąd ogólny	Nieznane
...029				
80000030	Całkowicie nieczynny	CPU	Błąd ogólny	FPCP Rozgałęzienie za nie uporządkowanym stanem
80000031	Całkowicie nieczynny	CPU	Błąd ogólny	FPCP Niedokładny wynik
80000032	Całkowicie nieczynny	CPU	Błąd ogólny	FPCP Dzielenie przez zero
80000033	Całkowicie nieczynny	CPU	Błąd ogólny	FPCP Dolny bieg... (np.rzeki)
80000034	Całkowicie nieczynny	CPU	Błąd ogólny	FPCP Błąd argumentu
80000035	Całkowicie nieczynny	CPU	Błąd ogólny	FPCP Najście
80000036	Całkowicie nieczynny	CPU	Błąd ogólny	FPCP Sygnał NAN
80000037	Całkowicie nieczynny	CPU	Błąd ogólny	FPCP Nieskończony sygnał
80000038	Całkowicie nieczynny	CPU	Błąd ogólny	PMMU Wstrzymanie
80000039	Całkowicie nieczynny	CPU	Błąd ogólny	PMMU Nielegalne ustawienie
80000040	Całkowicie nieczynny	CPU	Błąd ogólny	Nieznane
8000000A	Całkowicie nieczynny	CPU	Błąd ogólny	Linia 1010 Emulator
8000000B	Całkowicie nieczynny	CPU	Błąd ogólny	Linia 1111 Emulator
8000000C	Całkowicie nieczynny	CPU	Błąd ogólny	Nieznane
8000000D	Całkowicie nieczynny	CPU	Błąd ogólny	Koprocesor - protokół naruszenia
8000000E	Całkowicie nieczynny	CPU	Błąd ogólny	Stos - błąd formatu
8000000F	Całkowicie nieczynny	CPU	Błąd ogólny	Nieznane
8000001A	Całkowicie nieczynny	CPU	Błąd ogólny	Auto-wektorowa płaszczyzna 2 błędu przerwania
8000001B	Całkowicie nieczynny	CPU	Błąd ogólny	Auto-wektorowa płaszczyzna 3 błędu przerwania
8000001C	Całkowicie nieczynny	CPU	Błąd ogólny	Auto-wektorowa płaszczyzna 4 błędu przerwania
8000001D	Całkowicie nieczynny	CPU	Błąd ogólny	Auto-wektorowa płaszczyzna 5 błędu przerwania
8000001E	Całkowicie nieczynny	CPU	Błąd ogólny	Auto-wektorowa płaszczyzna 6 błędu przerwania
8000001F	Całkowicie nieczynny	CPU	Błąd ogólny	Auto-wektorowa płaszczyzna 7 błędu przerwania
80F8155C	Całkowicie nieczynny	CPU	Nieznane	Nieznane
81000000	Całkowicie nieczynny	exec.library	Błąd ogólny	Problem z exec.library
81000001	Całkowicie nieczynny	exec.library	Błąd ogólny	68000 Wyjatek-wektor-suma kontrola
81000002	Całkowicie nieczynny	exec.library	Błąd ogólny	Execbase - błąd sumy kontrolnej
81000003	Całkowicie nieczynny	exec.library	Błąd ogólny	Library - błąd sumy kontrolnej
81000004	Całkowicie nieczynny	exec.library	Błąd ogólny	Nieznane
81000005	Całkowicie nieczynny	exec.library	Błąd ogólny	FreeMem: niszczące zapełnienie pamięci
81000006	Całkowicie nieczynny	exec.library	Błąd ogólny	Brak pamięci na przerwanie
81000007	Całkowicie nieczynny	exec.library	Błąd ogólny	InitStruct () jako APTR-źródło
81000008	Całkowicie nieczynny	exec.library	Błąd ogólny	Jeden semafor jest w nielegalnym położeniu
81000009	Całkowicie nieczynny	exec.library	Błąd ogólny	Zwolnienie wolnej pamięci
81000010	Całkowicie nieczynny	exec.library	Błąd ogólny	Próba użycia zwiększonego MSG-semafora
8100000A	Całkowicie nieczynny	exec.library	Błąd ogólny	Nielegalny 68K-zastrzeżone użycie
8100000B	Całkowicie nieczynny	exec.library	Błąd ogólny	Zastosowanie jednego aktywnego IORquest
8100000C	Całkowicie nieczynny	exec.library	Błąd ogólny	Sprawdzenie w pamięci listy negacji
8100000D	Całkowicie nieczynny	exec.library	Błąd ogólny	IO do zamkniętego IORquest
8100000E	Całkowicie nieczynny	exec.library	Błąd ogólny	Zamrożenie stosu do chwili powstania
8100000F	Całkowicie nieczynny	exec.library	Błąd ogólny	Memory-Header nie znaleziony
82000000	Całkowicie nieczynny	graphics.library	Błąd ogólny	Problem z graphics.library
83000000	Całkowicie nieczynny	layers.library	Błąd ogólny	Problem z layers.library
84000000	Całkowicie nieczynny	intuition.library	Błąd ogólny	Problem z intuition.library
84000001	Całkowicie nieczynny	intuition.library	Błąd ogólny	Nieznany wzór Gadget
85000000	Całkowicie nieczynny	math#?.library	Błąd ogólny	Problem z math.library
8580DEAD	Całkowicie nieczynny	math#?.library	Błąd ogólny	Nieznane
8581DEAD	Całkowicie nieczynny	math#?.library	Brak pamięci	Nieznane
8582DEAD	Całkowicie nieczynny	math#?.library	Nie ustalono utworzonej biblioteki	Nieznane
8583DEAD	Całkowicie nieczynny	math#?.library	Nie ustalono otwartej biblioteki	Nieznane
8584DEAD	Całkowicie nieczynny	math#?.library	Nie ustalono otwartego urządzenia	Nieznane
8585DEAD	Całkowicie nieczynny	math#?.library	Nie ustalono otwartego zasobu (Resource)	Nieznane
8586DEAD	Całkowicie nieczynny	math#?.library	Błąd I/O	Nieznane
8587DEAD	Całkowicie nieczynny	math#?.library	Brak sygnału	Nieznane
8588DEAD	Całkowicie nieczynny	math#?.library	Falszywy parametr	Nieznane



8589DEAD	Całkowicie nieczynny	math#.library	Ne ustalono zamknięcia biblioteki	Nieznane
86000000	Całkowicie nieczynny	Nieznane	Błąd ogólny	Nieznane
87000000	Całkowicie nieczynny	dos.library	Błąd ogólny	Problem z dos.library
87000001	Całkowicie nieczynny	dos.library	Błąd ogólny	Nieznane
87000002	Całkowicie nieczynny	dos.library	Błąd ogólny	EndTask-negacja
87000003	Całkowicie nieczynny	dos.library	Błąd ogólny	Qpkt-błąd
87000004	Całkowicie nieczynny	dos.library	Błąd ogólny	Niespodziewane otrzymanie pakietu
87000005	Całkowicie nieczynny	dos.library	Błąd ogólny	FreeVec-negacja
87000006	Całkowicie nieczynny	dos.library	Błąd ogólny	Dysk-Sekwencja blokowa-błąd
87000007	Całkowicie nieczynny	dos.library	Błąd ogólny	Zniszczenie Bitmapy
87000008	Całkowicie nieczynny	dos.library	Błąd ogólny	Klawisz jest już wolny
87000009	Całkowicie nieczynny	dos.library	Błąd ogólny	Falszywa suma kontrolna
8700000A	Całkowicie nieczynny	dos.library	Błąd ogólny	Błąd Dysku
8700000B	Całkowicie nieczynny	dos.library	Błąd ogólny	Klawisz poza zakresem
8700000C	Całkowicie nieczynny	dos.library	Błąd ogólny	Wadliwe pokrycie
8700000D	Całkowicie nieczynny	dos.library	Błąd ogólny	Nieważny pakiet-Init dla CLI/Shell
8700000E	Całkowicie nieczynny	dos.library	Błąd ogólny	Filehandle, raz bardzo zamknięty
8700000F	Całkowicie nieczynny	dos.library	Błąd ogólny	Nieznane
88000000	Całkowicie nieczynny	ramlib	Błąd ogólny	Problem z ramlib
88000001	Całkowicie nieczynny	ramlib	Błąd ogólny	Niedozwolone pokrycia bibliotek
89000000	Całkowicie nieczynny	icon.library	Błąd ogólny	Problem z icon.library
9FFFFFF9	Całkowicie nieczynny	Nieznany	Nieznane	Nieznane
B1010009	Całkowicie nieczynny	Workbench	Brak pamięci	Uszkodzony GadTools
B3000000	Całkowicie nieczynny	gadtools.library	Błąd ogólny	Problem z gadtools.library
B3010000	Całkowicie nieczynny	gadtools.library	Brak pamięci	Nieznane
B3020000	Całkowicie nieczynny	gadtools.library	Nie ustalono utworzonej biblioteki	Nieznane
B3030000	Całkowicie nieczynny	gadtools.library	Nie ustalono otwartej biblioteki	Nieznane
B3040000	Całkowicie nieczynny	gadtools.library	Nie ustalono otwartego urządzenia	Nieznane
B3050000	Całkowicie nieczynny	gadtools.library	Nie ustalono otwartego zasobu (Resource)	Nieznane
B3060000	Całkowicie nieczynny	gadtools.library	Błąd I/O	Nieznane
B3070000	Całkowicie nieczynny	gadtools.library	Brak sygnału	Nieznane
B3080000	Całkowicie nieczynny	gadtools.library	Falszywy parametr	Nieznane
B3090000	Całkowicie nieczynny	gadtools.library	Ne ustalono zamknięcia biblioteki	Nieznane
FFC0FFC0	Całkowicie nieczynny	Nieznany	Nieznany	Nieznane
FFC1FFC1	Całkowicie nieczynny	Nieznany	Nieznany	Nieznane
FFC2FFC2	Całkowicie nieczynny	Nieznany	Nieznany	Nieznane
FFC3FFC3	Całkowicie nieczynny	Nieznany	Nieznany	Nieznane
FFC4FFC4	Całkowicie nieczynny	Nieznany	Nieznany	Nieznane



BŁĘDY STARTU I AUTODIAGNOSTYKI

Kolor obrazu na po włączeniu zasilania widoczny na monitorze:

- **CIEMNOSZARY** procesor jest sprawny. Oznacza to, że najpoważniejszy układ naszego komputera działa bez zarzutu.
- **JASNOSZARY** poprawna wartość sumy kontrolnej pamięci ROM. Kickstart zgłasza się poprawnie.
- **BIAŁY** wystarczająca do uruchomienia komputera ilość pamięci RAM. Oznacza to, że mamy co najmniej 256 KB Chip RAM oraz, jeśli mamy rozszerzenie pamięci, co najmniej 256 KB Fast RAM. Czasami jednak może określać uszkodzenie inne – chip, pamięć.
- **CZERWONY** błąd sumy kontrolnej ROM lub błąd sprzętowy. Oznacza to, że uszkodzony został Kickstart albo któreś z podłączonych urządzeń (kart).
- **ZIELONY** uszkodzenie Chip RAM.
- **NIEBIESKI** uległ uszkodzeniu jeden z układów specjalizowanych Amigi.
- **ŻÓŁTY** sygnalizacja błędu w uruchamianym programie przed instalacją obsługi błędów.

Jeśli zamiga dioda CapsLock:

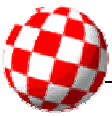
- 1 raz: błąd sumy kontrolnej ROM;
- 2 razy: uszkodzenie jednej z kości pamięci RAM;
- 3 razy: wewnętrzny timer uległ uszkodzeniu (nie chodzi bynajmniej o zegar z karty rozszerzenia);
- 4 razy: zwarcie w klawiaturze -- natychmiast wyłączyć komputer!

BŁĘDY GURU MEDIATION

Software Failure. Press left mouse button to continue.
Guru Meditation #00000000.4E454D4F

#000000x.yyyyyyyy błąd procesora
#aabbcccc.dddddddd błąd systemu operacyjnego

Kolory: **ZIELONY** błąd ignorowalny
ŻÓŁTY błąd odzyskiwalny – *Recoverable Alert*
CZERWONY błąd ostateczny – *dead-end*

**PROGRAMY**LHA

LHA jest stosunkowo starym i trochę już przestarzałym formatem kompresji danych. Kompresja LHA jest tym dla Amigi czym dla PC jest format ZIP (chodzi oczywiście o popularność, kompresja LHA jest jednak wydajniejsza do ZIP).

LHA [-options] command archive[.lha/lzh] file(s) [DestDir:]

[-options]	opcje:	
	-z/0/2/3	astosowany algorytm kompresji, gdzie: z - brak kompresji (store), 0 - słaba kompresja (algorytm LhArc V1.x), 2 - domyślna metoda (algorytm -lh5-), 3 - najlepsza kompresja (algorytm -lh6-). Naturalnie im lepszy algorytm zastosujemy tym dłużej będzie trwała operacja archiwizacji danych za to objętość archiwum będzie mniejsza.
	-a	zachowuje atrybuty archiwizowanych plików
	-e	archiwizuje również puste podkatalogi
	-r	archiwizuje również podkatalogi z plikami
	-x	zapamiętuje pełne ścieżki
command	komendy:	
	a	archiwizacja (dodaje pliki do archiwum)
	m	archiwizacja (przenosi pliki do archiwum)
	u	archiwizacja przez update (aktualizuje pliki w archiwum)
	e	wydobywa pliki z archiwum
	x	wydobywa pliki z archiwum z zachowaniem pełnych ścieżek
	v	listuje (wyświetla zawartość) archiwum
archive[.lha/lzh]	nazwa archiwum	
file(s)	pliki przeznaczone do zaarchiwizowania	
[DestDir:]	katalog na wypakowywane z archiwum pliki	

LHA -e archive[.lha/lzh] destdir:	Rozpakowanie dowolnego archiwum w formacie LHA/LZH
LHA -e filemaster.lha RAM:	Wypakuje wszystkie pliki z archiwum "filemaster.lha" do Ram Dysku.
LHA -e disk1.lha DH0:Temp/	Wypakuje pliki z archiwum "disk1.lha" do katalogu Temp na dysku twardym DH0:. Uwaga! Jeśli podajemy ścieżkę docelową jako katalog a nie symbol urządzenia logicznego stosujemy znak / (slash) po nazwie katalogu tak jak w powyższym przykładzie.
LHA a archive[.lha/lzh] file(s)	Tworzenie archiwum LHA.
LHA a DF0:tools.lha SYS:Tools/#?	Program utworzy archiwum "Tools.lha" na dyskietce w stacji DF0: z wszystkich plików znajdujących się w głównym podkatalogu "Tools" na dysku systemowym.
LHA -3 -a -e -r a DF0:tools.lha SYS:Tools/#?	Kompresja z opcją "-3" zmienia zastosowany algorytm kompresji na najlepszy -lh6- (domyślnie jest 2 [-lh5-]). Kompresja z podkatalogami włącznie z pustymi oraz zachowanie atrybutów plików.
LHA -3 -a -e -r -x a DH0:Dyski/Dysk1.lha DF0:#?	Zaarchiwizowanie całej zawartości dyskietki w stacji DF0: z zachowaniem katalogów i atrybutów plików do archiwum Dysk1.lha, które zostanie utworzone w katalogu "DH0:Dyski" (podany katalog musi istnieć!)
LHA x DH0:Dyski/Dysk1.lha DF0:	Wypakowanie archiwum na czystą dyskietkę w stacji DF0:
LHA -v archive[.lha/lzh]	Listowanie archiwum LHA o pełnej ścieżce dostępu.
LHA -l archive[.lha/lzh] (-l to litera (el [od list]))	Listowanie archiwum LHA o pełnej ścieżce dostępu.

DMS

Wydany przez bliżej nieznaną mi firmę ParCon Software DMS (Device-Masher) należy do grupy archiwizerów całodyskowych. Programy tego typu odczytują i zapisują dane "po ścieżkach" bez względu na format nośnika. Oznacza to, że potrafią one również archiwizować zawartość dyskietek niesformatowanych pod kontrolą AmigaDOS (tzw. "niedosowych"). Sposób ten umożliwia tworzenie archiwów również z gier/demo'sów, które to najczęściej korzystają z własnych formatów ("niedosowych").

DMS command file[.dms] [option] [option] ..

command	typowe komendy:	
	Read	odczyt dyskietki i kompresja do archiwum DMS
	Write	dekompresja archiwum DMS i zapis na dyskietkę
	Repack	przepakuje (dekompresja i ponowna kompresja) istniejące archiwum DMS (możemy zmienić metodę kompresji)
	View	podaje szczegółowe informacje o dowolnym archiwum DMS
	Test	testuje poprawność archiwum DMS
file[.dms]	nazwa archiwum DMS	
[options]	typowe opcje:	



dyskietki	[FROM Dev:]	symbol stacji dysków, w której zostanie wykonany odczyt
dyskietkę	[TO Dev:]	symbol stacji dysków, w której zostanie wykonany zapis na
	[TO File]	nazwa dla nowego archiwum DMS (tylko, gdy przepakowujemy)
	[CMODE BEST/NONE/HEAVY1/HEAVY2]	zastosowany algorytm kompresji, gdzie: BEST (domyślnie) - najlepsza kompresja, NONE - brak kompresji, HEAVY1 - średnia kompresja, HEAVY2 - to samo co BEST
	[LOW lowtrack]	numer ścieżki na dyskietce, od której zacznie się odczyt/zapis danych
	[HIGH hightrack]	numer ścieżki na dyskietce, na której zakończy się odczyt/zapis danych
	[NOZERO]	ignoruje mapę dyskietki (nie pomija pustych ścieżek)
	[NOVERIFY]	brak weryfikacji poprawności zapisu na dyskietkę
	[HD]	umożliwia wykonanie archiwizacji/dearchiwizacji na dyskietkach HD (1760 KB w napędach wysokiej gęstości)

Aby zaarchiwizować dowolną dyskietkę:

```
DMS Read [Dysk:ścieżka/]nazwa_pliku[.dms] [FROM Dev:]
```

DMS Read RAM:Disk1.dms Zawartość dyskietki w stacji DF0 zostanie zapisana w pliku "Disk1.dms" na Ram Dysku. Domyślnie program sprawdza właśnie stację DF0. Dlatego, jeśli odczyt dyskietki będzie dokonywany z wewnętrznego napędu dyskietek, nie jest wymagane podawanie jego symbolu.

DMS Read RAM:Disk1.dms FROM DF1: Zaarchiwizuje dyskietkę w stacji DF1 na Ram Dysku w pliku "Disk1.dms". Jeśli odczytywana ma być dyskietka znajdująca się w którejś z pozostałych stacji należy podać jej symbol po słowie "FROM" (jak w tym przykładzie).

DMS Read RAM:Disk1.dms FROM DF1: CMODE NONE Domyślnie program kompresuje wczytywane ścieżki z dyskietki metodą "BEST" - czyli najmocniejszym algorytmem. Jeżeli jednak nie chcemy pakować odczytywanych danych (plik DMS będzie miał objętość równą objętości dyskietki, czyli 880 KB dla napędów DD) zmieniamy zastosowany algorytm na "NONE":

```
DMS Read RAM:Disk1a.dms FROM DF0: LOW 0 HIGH 39
```

```
DMS Read RAM:Disk1b.dms FROM DF0: LOW 40 HIGH 79
```

Użycie pierwszej składni polecenia odczyta ścieżki 0-39 z dyskietki i zapisze do pliku "RAM:Disk1a.dms", druga składnia zrobi podobnie z pozostałymi 40 ścieżkami (40-79) i zapisze tą "drugą połowę" dyskietki do pliku "RAM:Disk1b.dms".

Aby wypakować dowolne archiwum DMS na czystą dyskietkę:

```
DMS Write [Dysk:ścieżka/]nazwa_pliku[.dms] [TO Dev:]
```

[Dysk:ścieżka/]nazwa_pliku[.dms] Dokładna lokalizacja archiwum DMS
[TO Dev:] Dymbol stacji dysków, w której znajduje się czysta dyskietka.

DMS Write RAM:Disk1.dms Wypakuje zawartość archiwum "RAM:Disk1.dms" na czystą dyskietkę w stacji DF0. Jak pisałem wyżej, domyślnym napędem jest wewnętrzna stacja Amigi, dlatego jej symbol nie musi zostać podany.

DMS Write RAM:Disk1.dms TO DF2: Wypakuje zawartość tego samego archiwum DMS na dyskietkę w stacji DF2.

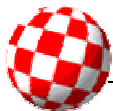
```
DMS Write DH0:Disk1a.dms TO DF1: LOW 0 HIGH 39
```

```
DMS Write DH0:Disk1b.dms TO DF1: LOW 40 HIGH 79
```

Jeśli mamy dwa poprzednio utworzone archiwa DMS z zawartości jednej dyskietki, znajdujące się teraz dla odmiany na dysku twardym DH0 i chcemy zrzucić ich zawartość na dyskietkę w stacji DF1,

DMS Write RAM:Disk1.dms TO DF0: NOVERIFY
Brak weryfikacji dyskietki

DMS Write DH2:Archives/Disks/HD/disk1.dms TO DF3: HD Wypakowanie uprzednio utworzonej z dyskietki HD archiwum DMS na czystą dyskietkę HD w stacji HD (u nas DF3).

**SŁOWNIK AMIGOWCA**

AGA – Advanced Graphics Architecture, zaawansowana architektura graficzna – to ulepszony układ grafiki stosowany w trzeciej generacji komputerów Amiga, który składał się z chipsetów:

- **Alice** – nowszy odpowiednik układu *Agnus*
- **Lisa** – nowszy odpowiednik układu *Denise*
- **Paula** – układ dźwiękowy

Dodatkowo do architektury AGA zalicza się układy:

- **Super Gary** – kontroler stacji dyskietek i dysku twardego
- **Super Amber** – sprzętowy flicker-fixer (podwajanie częstotliwości odświeżania)
- **Super Ramsey** – kontroler dla pamięci SIMM (tylko Amiga 4000)
- **Akiko** – chip usprawniający konwersję chunky-planar (tylko Amiga CD32)

Najbardziej zaawansowany system koprocesorów (układów specjalizowanych) zwany też "AA-ChipSet" (Advanced Amiga) wbudowany na płycie popularnej Amigi 1200 i w dużej A4000 oraz konsoli CD32. Umożliwia korzystanie trybów graficznych:

Nazwa trybu	Rozdzielczość		Częstotliwość		Obsługiwane palety
	min	max	pozioma	pionowa	
PAL	320 x 256	1280 x 512	50 Hz	15,60 kHz	od 2 do 256 kolorów jednocześnie z palety 24-bitowej (16 777 216 kolorów)
NTSC	320 x 200	1280 x 400	60 Hz	15,72 kHz	
Euro36	320 x 200	1280 x 400	73 Hz	15,76 kHz	
A2024	-----	1024 x 1024	15 Hz	15,72 kHz	
DbiPAL	320 x 512	640 x 1024	50 Hz	29,45 kHz	
DbiNTSC	320 x 400	640 x 800	59 Hz	29,20 kHz	
Multiscan	640 x 480	640 x 960	60 Hz	31,44 kHz	
Super72	400 x 300	800 x 600	72 Hz	24,62 kHz	
Euro72	640 x 400	640 x 800	70 Hz	31,43 kHz	262 144 kolory jednocześnie z palety 24-bitowej

AHI – (Audio Hardware Interface) interfejs dźwięku umożliwiający obsługę dowolnej karty dźwiękowej w systemach amigowych, o ile posiadamy do niej stosowne sterowniki.

Ambient – powłoka graficzna systemu MorphOS oparta na MUI.

AmigaDOS – powłoka tekstowa systemu AmigaOS oparta na systemie Tripos firmy MetaComCo z początku lat 80 XX wieku.

Arexx – Amiga Rexx – implementacja języka skryptowego Rexx w systemie AmigaOS.

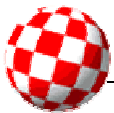
Bit Plane – w dosłownym tłumaczeniu "płaszczyzna bitów". W Amidze obraz wyświetlany na ekranie reprezentowany jest w pamięci w postaci płaszczyzn bitowych. Amiga obsługuje ich maksymalnie sześć (kości OCS/ECS) lub osiem (kości AGA). Zatem o liczbie normalnie dostępnych kolorów decyduje ilość użytych płaszczyzn bitowych (dany tryb graficzny). Oznacza to, że jeżeli ustawiony jest tryb 2-kolorowy - do rysowania obrazu wykorzystywany jest tylko jeden bitplan (bit zapalony - kolor punktu, bit zgaszony - kolor tła). W trybie 4-kolorowym - używane są dwie płaszczyzny, przy 8-kolorach - trzy, itd. Płaszczyzny są nałożone na siebie i dają pewną kombinację bitów, która stanowi numer rejestru koloru użytego przy rysowaniu danego punktu na ekranie.

Blitter – procesor specjalizowany, zajmuje się kopiowaniem obszarów pamięci wraz z możliwością wykonania operacji logicznych na danych oraz kreśleniem linii. Układ ten jest wyposażony w specjalną funkcję wypełniającą kontury. Jego operacje charakteryzuje bardzo duża szybkość - w ciągu sekundy jest on w stanie przesłać do 2 MB danych.

BootBlock – pierwsze dwa sektory startowe dyskietki (1024 bajty). Po jej włożeniu do stacji są odczytywane przez Amigę. Aby dyskietka wystartowała, w sektorach tych musi się znajdować specjalny program inicjalizujący. Odpowiedzialny m.in. za ukazanie się okienka "AmigaDos" (dyskietki "dosowe") lub ładowanie gry (dyskietki "niedosowe" - najczęściej). Dyskietka bez zapisanego bootblock'u jest nieuruchamialna (np. dysk na dane). Z kolei uszkodzenie tych sektorów prowadzi zwykle do niemożności dalszego jej odczytu (dotyczy dyskietek nie używających formatu DOS). Program wykonywany na bootblock'u nie jest kontrolowany przez system. Dlatego może być również szkodliwy (virus bootblock'owy). Na Amigę powstało/powstaje najwięcej właśnie tych virusów. Mogą one również zablokować dalszy odczyt danych z dyskietki lub cały system.

Chip RAM – zwana też pamięcią audio/video. Może być używana przez wszystkie procesory (graficzny, muzyczny i główny) i z tego powodu aplikacje uruchamiane przy użyciu tej pamięci działają wolniej. Amiga może mieć co najwyżej 2 MB tego typu pamięci.

Tu muszą być zlokalizowane dane ekranu, dźwięku, bufor stacji dysków, program dla Copper'a (tzw. Copper-List'a) oraz dane dla Blitter'a. Pierwsze Amigi 1000 wyposażone były tylko w 256 KB Chip-RAM, co pozwalało jedynie na uruchomienie systemu operacyjnego (kickstart'u wczytywanego jeszcze wtedy z dyskietki). Później zaczęto montować w nich 512 KB, gdy zaś pojawiła się Amiga 500, od samego początku montowano w niej standardowo 512 KB Chip-RAM. W nowszych Amigach (500+, 600) wyposażonych w nową wersję układu Agnus (Super Fat Agnus) zamontowano 1 MB z możliwością rozbudowy do 2 MB. Amiga 3000, 1200 i 4000 od początku posiadała 2 MB Chip-RAM.



CIA – tak zwane "porty". Amiga posiada dwa układy CIA-A i CIA-B. Zajmują się one obsługą złącza komputera. Używając pamięci od \$A00000 do \$BFFFFF, przechowują w tym obszarze m.in. rejestry stacji dysków i złącza drukarki. Porty CIA znane są zresztą ze swych destruktywnych skłonności do psucia się bez przyczyny. Często trudno w ogóle stwierdzić uszkodzenie.

Class – obiekty zorientowane na wykonywanie określonej czynności, np widget opisuje zdarzenie interfejsu. Wprowadzenie klas poprowadziło do powstania MUI (Magic User Interface) i ClassACT, które następnie przekształciło się w ReAction.

ClassACT – nowe API wprowadzone do systemu AmigaOS 3.5 początkowo dodatek do Intuition. Bardzo rozbudowane w systemie 3.9, ostatecznie zastąpiło Intuition w systemie 4.0 gdzie przybrało nazwę ReAction.

Commodities – mechanizm małych programów, wykonujących pojedyncze czynności w celu usprawnienia pracy z systemem. MUI wykorzystuje ten mechanizm bez względu na rodzaj programu i jego zastosowanie. ReAction decyduje o tym, czy program ma być "commodities" oddaje w ręce programisty i dzieje się tak bez względu na wersję systemu AmigaOS.

Copper – koprocesor, który tak jak procesor główny, posiada własne instrukcje. Zna on tylko trzy rozkazy: MOVE, WAIT oraz SKIP - są one jednak uniwersalne. Pisze się z nich specjalny program dla copper'a, tzw. "Copper-List'ę". W ten sposób, za pomocą tego układu, możemy zmieniać zawartość rejestrów układów specjalizowanych Amigi - w zależności od położenia wiązki elektronów na ekranie. Rozkaz MOVE przesyła słowo do określonego rejestru sprzętowego (adresy \$DFFxxx). Rozkaz WAIT oczekuje, aż wiązka elektronów tworząca ekran osiągnie zadaną pozycję, natomiast SKIP powoduje warunkowe pominięcie instrukcji. Brak tej kości uniemożliwiłoby powstanie około 90% wszystkich "dem" dla Amigi.

Datatypes – typy danych – pliki procedur odpowiedzialnych za obsługę różnych formatów multimediów. Pomysł datatypes zadebiutował w AmigaOS 3.0 sprawiając, że aplikacje zdolne są bez większych problemów wirtualnie odczytać każdy rodzaj pliku zawierającego dane (obrazki, dźwięki, filmy, multimedia, itd.), dla którego istnieje odpowiednia klasa datatype. W szczególnych przypadkach pomaga to zaoszczędzić programistom czasu i trudu podczas pracy nad własnymi procedurami dekodującymi, jak również sprawia, że aplikacje mogą wczytywać dowolne rodzaje danych bez potrzeby modyfikacji samej aplikacji.

DirectoryOpus Magellan – początkowo menadżer plików, następnie przekształcił się w nakładkę – zamiennik Workbench'a

Denise – nazwano tak układ specjalizowany Amigi, odpowiedzialny za generowanie obrazu – czyli układ wizyjny (kość graficzna). Posiada własne rejestry, z których odczytuje dane grafiki i sprite'ów poczym zamienia je na odpowiednią reprezentację sygnałów RGB.

Device – **Driver** – program do obsługi fizycznych urządzeń, tzw. sterownik (program sterujący działaniem urządzenia). Część z nich znajduje się w ROM-ie Amigi (np: trackdisk.device - do obsługi stacji dysków, ramdrive.device - obsługujący Ram Dysk), część zaś w katalogu DEVS na dysku systemowym (np: serial.device, parallel.device). Podobnie jak w przypadku bibliotek systemowych rozwiązanie to umożliwia pisanie własnych sterowników urządzeń zewnętrznych i umieszczanie ich we wspomnianym katalogu.

Dragowanie ekranu – unikalny system przełączania się pomiędzy programami działającymi w pełnych **ekranach (screenach)**, polegający na przesunięciu górnej belki w dół ekranu

ECS – **Enhanced Chip Set** (ulepszony chipset) – nowy skład koprocesorów zastosowany w Amigach II generacji, zawierający:

- **Paula** – stara wersja
- **Super Agnus [Super Fat Agnus 8732B]** – obsługa do 2 MB CHIP RAM
- **Super Denise [87373]** – ulepszona wersja.

ECS znajduje się w modelu Amigi 500+, Amidze 600 oraz A3000. W stosunku do kości OCS nowy chipset zwiększył funkcjonalność starych trybów graficznych oraz wprowadził nowe rozdzielczości:

[**Productivity**] 640x480 – 4 kolory z palety 4096

[**SuperHires**] 1280x256 i 1280x200 przy dwóch lub czterech kolorach z palety 4096.

EHB – **Extra Half Bright** – specjalny tryb graficzny umożliwiający uzyskanie równocześnie 64 kolorów na ekranie. Na jeden punkt przypada 6 bitów (płaszczyzn) określających numer użytego koloru. Jednak Amiga ma tylko 32 rejestry kolorów (modele wyposażone w kości ECS). Pięć bitów więc określa numer rejestru koloru, zaś szósty, gdy jest włączony, jasność punktu zmniejszana jest dwukrotnie. W ten sposób, oprócz normalnie dostępnych 32 kolorów mamy dodatkowo 32. W Amigach wyposażonych w kości AGA wszystkie 64 kolory są niezależne.

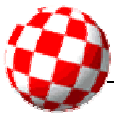
Exec – zorientowane obiektowo wielozadaniowe jądro typu mikrokernel systemu AmigaOS.

Fast RAM – do tej pamięci dostęp ma jedynie procesor główny, toteż jest ona najszybsza. W mniejszych Amigach jest 16-bitowa, w lepszych 32-bitowa. W najmniejszej Amidze (Amidze 600) można jej zainstalować co najwyżej 32 MB, w Amidze 2000 i 3000 ponad 1 GB przez złącze Zorro II/III, a w Amidze 4000 nawet 1,25 GB. W nierozbudowanej Amidze pamięć ta leży od adresu \$200000 i może mieć do 8MB (do adresu \$9FFFFFF).

Obszar ten zajmuje rozszerzenie pamięci podłączane do gniazda w lewym boku Amigi. Fast (szybki), dlatego, że pamięć ta ma własne kanały DMA i procesor ma do niej całkowicie swobodny dostęp, kości specjalizowane (dźwięku, obrazu oraz Blitter) nie mają do niej w żaden sposób dostępu, gdyż dobierają się do Chip-RAM przez zupełnie inne kanały.

Flicker-fixer – urządzenie likwidujące uciążliwe dla człowieka migotanie obrazu, spowodowane zastosowaniem przeplotu (*interlace*) pół wizyjnych, przy wyświetlaniu sygnału dostosowanego do wyświetlania na telewizorze (PAL, NTSC) na ekranie VGA lub innym, oferującym wyższe częstotliwości odświeżania niż telewizor.

FPU – jest to koprocesor matematyczny, czyli jednostka zmiennoprzecinkowa. Ten dodatkowy układ wspomaga procesor główny w operacjach na liczbach zmiennopozycyjnych. Jest obsługiwany dopiero przez 32-bitowe procesory od 68020 w górę. Istnieją dwa typy jednostek FPU - 68881 i 68882. Przy czym ten drugi przy tej samej częstotliwości taktowania jest znacząco szybszy. Koprocesory



możemy spotkać w obudowach PLCC lub PGA. Te pierwsze mają wyprowadzone sygnały na obwodzie obudowy (jak w procesorze 68020). Natomiast w tych drugich ułożone są prostopadłe względem płaszczyzny obudowy (przypominają "jeża"). Również te pierwsze mogą chodzić z maksymalną częstotliwością 40 MHz, zaś w PGA do nieco ponad 60 MHz. Ich praca (szybkość) może być asynchroniczna (zależna od dodatkowego oscylatora) lub synchroniczna (zależna od procesora). FPU pozwala znacznie przyspieszyć skomplikowane obliczenia. Dotyczy to głównie programów do grafiki 3D, matematycznych, kompresorów i samych gier 3D. Jednak samo zainstalowanie układu nie spowoduje zwiększenia szybkości pracy komputera. Tzn. muszą istnieć specjalne wersje programów oraz gier, potrafiące obsługiwać to urządzenie. W związku z czym bardzo często zdarza się, instalując jakiś program, iż mamy do wyboru wersję dla 68000 bez koprocessora albo 68020 z koprocessorem.

Guru Mediation – medytacja, alert systemu dotyczący błędu w jego działaniu. Nazwa ma związek z powstawaniem systemu w latach 1984/85, kiedy konstruktorzy i programiści w krótkich przerwach podczas pracy oddawali się medytacjom na specjalnej macie.

HAM – Hold And Modify – tryb graficzny, który udostępni 4096 kolorów naraz na ekranie. Przy czym na jeden punkt przypada tylko 6 płaszczyzn (bitów). Osiągnięto to w ten sposób, że 2 bity określają funkcję 4 pozostałych. Czyli bezpośrednio adresowanie 1 z 16 kolorów lub zmianę wartości jednej z trzech składowych RGB na określoną przez te dwa bity. W tym drugim przypadku kolor punktu jest wyznaczany na podstawie koloru punktu poprzedniego.

Handler – sterowniki urządzeń logicznych takich jak konsole, potoki i inne.

Ikony – icons – piktogramy – system plików graficznych odpowiadający za wyświetlanie plików w środowisku graficznym.

Intuition – to oryginalny amigowy interfejs użytkownika oraz system okien i menu. Poprawione mechanizmy rysowania okien, możliwość generowania cieni pod oknami, zredukowane zapotrzebowanie na pamięć graficzną, lepsza obsługa tematów graficznych. Zastosowane w systemach do wersji 3.9.

JIT – (Just In Time Compiler – rekompilacja kodu w czasie rzeczywistym) – jest to konwersja kodu procesora MC68k Amigi na kod procesora komputera hosta i zapamiętanie go, dzięki czemu jeśli będzie użyty w przyszłości jeszcze raz, nie trzeba będzie go tłumaczyć ponownie tylko użyje się już ten przetłumaczony. To ma bardzo duży wpływ na wydajność emulacji.

Kickstart – ROM – nazwano tak zawartość pamięci ROM Amigi, przyczym chodzi tu o całość, a nie o poszczególne części systemu operacyjnego. W Amidze 500 zajmuje on 256 KB (Amiga 500+, 600, 1200, 3000 i 4000 - 512 KB). Obszar Kickstart'u (systemu operacyjnego) został umieszczony pod adresem \$FC0000 i ciągnie się aż do samego końca przestrzeni adresowej procesora MC 68000, czyli do adresu \$FFFFFF. Od adresu \$F80000 (do \$FBFFFF) znajduje się kopia ROM, przez nikogo (a już w zupełności przez Amigę) nie wykorzystywana. Z obszarem Kickstart wiąże się pewna ciekawostka: otóż zawsze po inicjalizacji (Reset) procesor pobiera sobie adres pierwszego rozkazu z adresu 4, jednakże zaraz po włączeniu zasilania nic tam nie ma (albo są jakieś śmieci). W tej sytuacji procesor natychmiast by się zablokował (razem z systemem). Problem ten rozwiązano w ten sposób, że w momencie inicjacji pod adresem \$000000 rozplanowywana jest pamięć ROM i procesor może sobie spokojnie pobrać potrzebny mu adres.

Część Kickstartu została zapisana na stałe w kościach pamięci ROM, a część jest ładowana z dyskietki lub dysku twardego komputera Amiga. W ROM'ie znajduje się jądro systemu oraz biblioteki potrzebne do wystartowania system, sterowniki SCSI, część Intuition oraz rdzeń AmigaDOS. W systemach AmigaOS 4.0 i nowszych całość Kickstartu ładowana jest z dysku twardego.


Library – wszystkie procedury systemu operacyjnego Amigi umieszczone są w tzw. bibliotekach. Część z nich znajduje się w ROM'ie (np: dos.library, exec.library), zaś pozostałe w katalogu LIBS (np: asl.library, diskfont.library) na dysku bądź dyskietce. System ten zapewnia przejrzystość oraz możliwość pisania własnych bibliotek (umieszczanych w katalogu LIBS). Raz załadowana biblioteka przechowywana jest w pamięci a uruchomione programy mogą korzystać z jej procedur w tym samym czasie. Co nieraz znacznie ułatwia życie programistom.

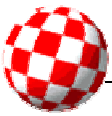
Locale – pliki z tłumaczeniami oprogramowania na języki narodowe. Dzięki możliwości zmiany domyślnych locali można w bardzo łatwy i szybki sposób zmienić obsługiwany język.

MMU – Memory Management Unit – jednostka zarządzająca pamięcią. Pozwala, poprzez krótki programik (np. VMM), założyć tzw. pamięć wirtualną RAM na twardym dysku. Przydatne, gdy posiadamy niewystarczającą ilość fizycznej pamięci (np. do obróbki zeskanowanych zdjęć). Posiadając odpowiedni procesor i dysk twarde, w łatwy sposób "stworzymy" sobie dodatkową pamięć. Wadą tego rozwiązania jest szybkość działania (pamięci wirtualnej), zależna od prędkości twardego dysku.

MUI – Magic User Interface – to zestaw współdzielonych bibliotek do tworzenia graficznego interfejsu użytkownika (GUI). Jedną z zalet MUI jest konfigurowalność, możliwość ustawień wyglądu programów przez użytkownika. Pomimo iż pakiet ten nigdy nie był dodawany do systemów AmigaOS 3.x (w AmigaOS 3.5 i 3.9 zamiast MUI wybrano nieco ulepszony ClassAct przemianowany na ReAction) był i jest popularny wśród programistów i użytkowników przez co stał się nieformalnym standardem cenionym sobie za swoje zalety, możliwości i łatwość programowania.

Do wersji 3.8 pakiet był rozwijany tylko dla AmigaOS 3.x. Wyższe (natywne) wersje rozwijane dla procesorów PowerPC (3.9 i 4.0) są jednym z podstawowych składników systemu MorphOS, a wersja 3.9 jest dodawana w formie dodatku do AmigaOS 4.0/4.1. MUI jest pakietem shareware dla AmigaOS 3.x i AmigaOS 4.0. MorphOS zawiera pełną wersję pakietu. Autorem MUI jest Stefan Stuntz.

Paleta kolorów MUI									
									
Colour									
Red	149	000	255	59	123	175	170	255	
Green	149	000	255	103	123	175	144	169	
Blue	149	000	255	162	123	175	124	151	



OCS – Original Chip Set – najstarszy (dla Amig I generacji) zestaw kości specjalizowanych, w skład którego wchodzi układy:

- **Agnus** (*Address Generator*) 8370 (**Fat Agnus** 8372A, **Super Agnus** 8372B) – serce OCS, układ odpowiadający za multimedia:
 - pełnił funkcję kontrolera DMA dla całej architektury (25 kanałów DMA),
 - zajmował się obsługą pamięci *Chip RAM* (8370 do 512kB, 8372A do 1MB, 8372B do 2MB),
 - zawierał w sobie koprocesory graficzne Blitter i Copper,
- **Paula** 8364 – niezmienny od 1984 chip odpowiadający za fenomenalne w swym czasie możliwości dźwiękowe Amigi:
 - cztery 8-bitowe przetworniki cyfrowo-analogowe (po 2 na każdy kanał stereo), poprzez obniżenie głośności w jednym z 8-bitowych przetworników na kanał można było zasymulować dodatkowe 6 młodszych bitów – co dawało w sumie 14 bitowy dźwięk stereo,
 - format próbki: PCM 8-bit, ze znakiem, kod uzupełnień do dwóch,
 - różna częstotliwość próbkowania w zależności od trybu graficznego (grafika i dźwięk korzystały ze wspólnej pamięci),
 - dynamika dźwięku – 48 dB,
 - kontroler obsługi joysticka i myszki,
 - filtr dolnoprzepustowy – działający globalnie na wszystkie kanały,
 - wspomagał układ *Gary* w obsłudze stacji dyskietek,
- **Denise** (*Display Enable*) 8362 – kontroler obrazu
 - zajmował się wyświetlaniem obrazu,
 - potrafił zsynchronizować własny sygnał video z zewnętrznym źródłem sygnału,
 - zajmował się obsługą duszków (sprites),

Do architektury **OCS** zalicza się także:

- **CIA** (*Complex Interface Adapters*)
 - kontroler portów wejścia/wyjścia,
 - kontroler klawiatury
- **Gary** 5719
 - kontroler szyny,
 - kontroler stacji dyskietek,
 - samodiagnostyka
- **Chip RAM** – pamięć, do której dostęp miały wszystkie układy (DMA)
- **Fast RAM** – pamięć, do której dostęp miał tylko procesor (pamięć instalowana opcjonalnie jako karta rozszerzeń)
- **Slow RAM** – (tylko w OCS) pamięć, do której dostęp miał tylko procesor, ale korzystał z niej przez wspólną z chipsetem szynę, co obniżało jej wydajność.

OCS korzysta z palety 4096 barw i udostępnia cztery rozdzielczości: (dla wersji PAL):

[**Lores**] 320x256 i 320x512 przy liczbie normalnie dostępnych kolorów od 2 do 32

[**Hires**] 640x256 i 640x512 przy 2-16 barwach

[**EHB**] [**Extra Half Bright**] tryb specjalny dla niskiej rozdzielczości poziomej [**Lores**] wyświetla 64 kolorów z palety 4096

[**HAM**] [**Hold And Modify**] tryb specjalny dla niskiej rozdzielczości poziomej [**Lores**] wyświetla wszystkie 4096 kolory.

Petunia – systemowy emulator procesora MC 68k z JIT w AmigaOS 4.0 i nowszych.

PowerUP – rozszerzenie bibliotek *exec.library* o obsługę procesorów PowerPC. Nie jest kompatybilne z systemem WarpOS.

Preferencje (Preferences) – katalog z aplikacjami do konfiguracji całego systemu operacyjnego.

Ram Dysk – wirtualny napęd tworzony podczas uruchamiania systemu operacyjnego AmigaOS. Jego wielkość jest ograniczona ilością wolnej pamięci RAM.

RDB – (Rigid Disk Block) – metoda organizacji dysku twardego i partycji w komputerach Amiga. W przeciwieństwie do swojego odpowiednika w komputerach PC – MBR, RDB bezpośrednio nie zawiera metadanych dla każdej partycji. Zamiast tego wskazuje na związane z listą bloków partycje. Dopiero dana partycja zawiera rzeczywiste dane partycji: ich początek, koniec, system plików, priorytet rozruchu, typ pamięci bufora. Ponieważ nie ma ograniczenia liczby w bloku partycji, nie ma potrzeby odróżniania podstawowych i rozszerzonych ich typów, stąd wszystkie partycje mają ten sam status ważności dla różnych architektów. RDB nie musi się znajdować w pierwszym sektorze dysku twardego, stąd też może współistnieć z MBR.

ReAction – nowe API obsługujące interakcje użytkownika w środowisku graficznym Workbench. ReAction ewoluowało z ClassACT wprowadzonego w AmigaOS 3.5/3.9 do jedyne API w systemach AmigaOS 4.0 i nowszych.

Reggae – nowe podejście do przetwarzania strumieni mediów w systemie MorphOS. Jest to modułowy system pobierania małych fragmentów plików multimedialnych i ich przetwarzania w czasie rzeczywistym (tzw rurociąg). Zaletą tego rozwiązania jest bardzo małe zużycie pamięci komputera. System w pełni wspiera także datatypy.

Rexx – (REstructured eXtended eXecutor) proceduralny język skryptowy do wzajemnej komunikacji programów działających w danym systemie operacyjnym.

ScalIOS – graficzny interfejs użytkownika stworzony dla systemu AmigaOS 3.0 – 3.9 jako zamiennik Workbench.

Screen - ekrany - to koncepcja wirtualnych pulpity lub obszarów roboczych, generowanych dynamicznie przez aplikacje w razie potrzeby. Każdy ekran może mieć inną rozdzielczość i głębię kolorów. Gadżet w prawym górnym rogu ekranu, pozwala na przełączanie się pomiędzy screenami poprzez przeciąganie w górę i w dół przez paski tytułów. Na starszych Amigach funkcje te były obsługiwane przez niestandardowe chipsety zaprojektowane specjalnie dla tego systemu; od AmigaOS 4 nowe techniki przyjęcia i ekranów są przeciągane w dowolnym kierunku także przez technologię **przeciągnij i upuść – drag and drop**.



Slow RAM – przez system widziana jako Fast RAM, w rzeczywistości działa jak Chip-RAM. Łączy wady obu poprzednich typów jest więc najgorsza przy czym nie ma zalet ani Chip ani Fast RAMu. Może mieć do 1,8 MB (od adresu \$C00000 do \$DBFFFF). Jest tania w instalacji (w przeciwieństwie do Fast'u), wkładana od spodu komputera (pod klapkę), jednak korzysta z tych samych kanałów co Chip-RAM, przez co jest spowalniana przez kości specjalizowane, choć nie mają one do niej dostępu.

Symbole wieloznaczne - jockers, glob, globbing – dopasowywanie wzorców wyszukiwania, są to np znaki „*“, „?“, „~“

Szuflada – Drawer – katalog w środowisku Workbench komputera Amiga.

Trance – systemowy emulator procesora MC 68k z JIT w MorphOS.

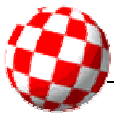
UAE – (Ultimate Amiga Emulator) – najpopularniejsza wersja emulatora komputera Amiga z procesorami serii MC68k obsługująca JIT. Dostępna dla różnych architektur (x86, PPC) i systemów operacyjnych (Windows, Linux, MacOS Classic i „X“, AmigaOS)

WarpOS – rozszerzenie bibliotek exec.library o obsługę procesorów PowerPC. Nie jest kompatybilne z systemem PowerUP.

WOM – Write Once Memory – pamięć jednokrotnego zapisu. Nazwano tak dodatkowy obszar w modelu Amigi 1000, którego pojemność wynosi 256 KB. Amiga 1000 nie posiadała jeszcze pamięci stałej ROM z zapisanym kickstart'em (systemem operacyjnym). Dlatego należało go wczytywać z dyskietki po włączeniu komputera właśnie do pamięci WOM. Operację tę przeprowadzało się tylko raz (aż do zaniku zasilania). Poczym niemożliwe było wpisanie do tej pamięci jakichkolwiek danych.

Wonderer – graficzny interfejs użytkownika stworzony dla systemu AROS.

Workbench (Warsztat) – natywny graficzny interfejs użytkownika i środowisko stworzone dla systemu AmigaOS. Wydany w roku 1985 wraz z AmigaOS 1.0. Bardzo często myli się go z AmigaOS. Workbench nie jest systemem operacyjnym a jedynie powłoką graficzną, wykorzystującą metody generowania elementów GUI – Intuition, ReAction lub MUI.



LITERATURA

http://www.amiga_emulation_in_pc.republika.pl/home.html
<http://exec.pl>
<http://en.wikipedia.org>
<http://www.gregdonner.org/workbench/>
<http://pl.wikipedia.org>
<http://ppa.pl>
<http://ulisses32.sitesled.com/amiga/>

NA WESOŁO

ODA DO AMIGI

Dwadzieścia lat niedługo upłynie od chwili,
Gdyśmy nową jakością zobaczyli.
Od Amigi 1000 to wszystko się zaczęło,
Co mysz i GUI miała, właśnie stąd się to wzięło.

Z początku skromna, pokazała swe oblicze
Swoimi możliwościami, których nie zlicze:
HAMowe obrazki (doprawdy kolorowe),
W NTSC Lo-res były resztki na głowę.

Przepiękny z jej portów stereo dźwięk wciąż miałeś,
Gdy słuchałeś wzmacniacza, to zawsze wiedziałeś,
Ze sample to były lub dźwięk generowany,
Nawet mowę słyszałeś - sukces murowany.

I był też procesk, "sześćdziesiąt osiem ka" zwany
7 MHz, często był doturbiany.
Pamięci 1MB każdemu starczało,
Gdy z flopa przez dwuwiscik kompa się bootowało

Scalaki w Amigach robiły trudne taski,
A kobiecie imiona zdołały ich maski:
Agnus, Paula i Denise robiły co mogły,
By godziny szczęścia/gry nigdy nas nie zmogły.

Używały też portów, niezły to numerk:
Dwukierunkowy PAR, czy też joystick do gierk.
A "grzyby" mogły tylko patrzeć w jedną stronę,
W te albo we wte, nie RAZEM, bo to złożone.

Gdy było jasne, że potęgę mają w rękach,
Wyszła z Workbenchem 1.3 nowa "pięćsetka".
Mniejsza i lepsza, z odpowiednim ROMem - RAMem,
Kupowana na gwiazdkę przez tatę i mamę.

Przez tak wiele lat używali ją ludziki,
Aż ząb czasu nadgryzł pod naporem techniki
A500, 600 i 1200 w rodzinie
Razem z towerkami, jej sukcesu przyczynie.

By znowu dzieło powrotu dokonać, nową Amigę nam zbudowali,
Wyposażając w nowsze układy z PC, AI ją teraz nazwali.
Ale napisałem to na swojej A1200 korzystając z YAMa,
Co było dla mnie zdecydowanie najlepszym "eksportem" od Wujka Sama

20 lat minęło od pierwszej A1000
MULTIMEDIALNEJ - to trzeba uczciwie przysiąc.
Reszta wie - to nie oni znaleźli te słowa,
Bo kto tego nie wie - niechaj pod stół się schowa.

Kończę tą odę dla pramatki wszystkich mediów,
Wiedźcie, że nigdy nie byłoby MULTIMEDIÓW,
Na darmo ciężka praca inżynierów cała,
Gdyby Amiga 1000 nigdy nie powstała...

Przekład M. Sokołowski

Śmieszne wierszyki

Amigowcze, drogie dziecko,
czy Ty masz pod czaszką równie?
Zrozum, że daleko przecie
nie ujedziesz na tym głównie

Stoi na biurku komputer Amiga.
Cichutko pracuje, pot z niej nie spływa.
Niczego się nie boi, wielka i potężna.
Kości AGA, Procesor szybciutki,
tak szybki jak w muzyce dwie nutki.
Ładna, zgrabna... Tak jak dziewczyna.
Kto to? To musi być Amiga...

Humor amigowy

Amigowiec został porażony przez swoją ukochaną amigę na śmierć i
poszedł do nieba. Staje przed Św.Piotrem a Piotr się go pyta:
- Chciałbyś Amigowcu pójść do nieba czy do piekła?
- Chciałbym się rozejrzeć - odpowiada nieśmiało amigowiec.
- Patrz tak wygląda niebo - mówi św.Piotr - pokazując dookoła.
- Niby chłodno tu - inteligentnie zauważa Amigowiec.
- Teraz chodź pokażę ci piekło - mówi św.Piotr
Zjeżdżają do piekła patrzą a tam gołe panienki siedzą na monitorach,
super myszki. - Dookoła leżą amigi a nawet najnowszy model "Amiga
CD Power Belzebub" z procesorem matolarola 40000000000, która ma
100 TMipsów.

Wracają na górę - czyli do nieba - i św.Piotr się pyta:
- No to jak - niebo czy piekło?
- Nie obraż się św. Piotr - ale ja wołam do piekła - mówi amigowiec.
Dobra pojechał na dół, wychodzi z windy a tu go dwa Diabły za bary i
do beczki ze smotą i na części. Przestraszony Amigowiec - krzyczy co się
stało z tamtym piekłem które widział poprzednio:
Na to jeden z diabłów odpowiada:
- To było DEMO baranie, DEMO...

A na Amidze nie ma gdzie piwa postawić, za to jest szybsza od PC-ta jak
spada na podłogę.

Przychodzi baba do weterynarza z Amigą na sznurku i mówi:
- Ja do uśpienia....

Przychodzi baba do lekarza:
Lekarz: - Co pani dolega?
Baba: - A miga mi coś tam.

Przychodzi Jasiu do szkoły i płacze..Pani mu na to Jasiu co jest?
- Rodzice kupili mi Amigę...Co ja im zrobiłem??

Wchodzi facet do komputerowego, patrzy, a tam ...miga.

- Kiedy Amiga jest szybsza od peceta?
- Gdy dobrze obciążona kamieniami spada z mostu do rzeki.

Comodore z klawiszami w kształcie rąbów, używane tylko przez
jaskiniowych głabów