

AMIGA ZYNY

MAGAZYN UŻYTKOWNIKÓW KOMPUTERÓW AMIGA

NR 22



Zdalne sterowanie » Własny styl Workbencha » Bootblock inaczej niż zwykle

Gry dla czterech graczy » ARexx praktycznie » Nowe narzędzia sieciowe » Rozszerzenia systemu 1.3

22

JADŁOSPIS

INTEGRACJA

Jeszcze więcej mocy
Wywiad: Grupa Retro Bones
Kartka z kalendarza

SPRZĘT

Amiga 500: uratowana
Karta GVP A530
Automatyczny rozruch na Raspberry Pi

WORKBENCH 1.3

Excellence
Ciekawe dodatki do Workbencha
Ikony 8-kolorowe
Czym jest Workbench 1.4?
Układanka w Amiga Basic

AMIGA OS

Polskie locale na dyskietce
Multi File System
Własny styl Workbencha (część 1.)
Installer: aktywność użytkownika
Tworzymy Amiga Guide
Najprostsze zabezpieczenie dysku
Przypisania raz jeszcze
Stare metody kompresji
Okna w Blitz Basicu
Bootblock inaczej niż zwykle
Użycie ARexxa w programie Music-X
Zdalne sterowanie
Nowe narzędzia sieciowe

ROZRYWKA

Gry dla czterech graczy
A-Train w detalach
Future Bike

 **AMIGA**.net.pl

AMIGAZYN

22/2020

Redakcja:
Adam Zalepa

Autorzy tekstów:
Rafał Chyła
Witek Klajn
Krzysztof Kliś
Marcin Libicki
Krzysztof Radzikowski
Kamil Stokowski
Robert Szacki
Mariusz Wasilewski

Wydawca:
A2 Renata Gralak, Łódź

Korekta:
Adrian Węgorek

Grafika:
Marzena Bukowska,
Amiga Hardware Database,
Freepik, Marcin Libicki

Jeszcze więcej mocy

ADAM ZALEPA

Macie przed sobą już 22 numer naszego magazynu, wydany w trudnym okresie dla nas wszystkich. Ograniczenia wynikające z zagrożenia koronawirusem powodują, że ukończenie i zebranie wszystkich artykułów stało się nie lada wyzwaniem. Musieliśmy zrobić kilka zmian w stosunku do pierwotnych planów, a ja musiałem zająć się składem w większym stopniu niż do tej pory. Mam nadzieję, że wyjdzie nam to tylko na dobre.

Tym razem poświęcamy więcej miejsca najprostszemu Amigom wyposażonym jeszcze w Kickstart 1.3. Wiele osób powracających do naszego komputera pamięta właśnie tę wersję systemu i jest to ich punkt wyjścia do dalszej zabawy w retro. Uświadomiła mi to rozmowa z długo niewidzianym znajomym, a jednocześnie zobaczyłem prawdziwy "friendship" w naszym środowisku. Otóż, wspomniany kolega wiele lat temu podarował swoją Amigę 500 innej osobie - zupełnie za darmo. Przekonany, że nigdy nie wróci do Amigi, która wtedy nie była prawie nic warta, był zadowolony, że ktoś zajmie się jego komputerem. Po latach okazało się, że chętnie przypomni sobie stare czasy i zagra w ulubione gry jak Moonstone czy Rampart. Odzyskał pięćsetkę od tej samej osoby, której wcześniej dał Amigę - również za darmo. Co prawda nie jest to ten sam egzemplarz, ale sprawny komputer, na którym może bawić się tak jak kiedyś. Oba zdarzenia dzieli ponad 10 lat.

Tak powinni zachowywać się wobec siebie Amigowcy. Wsparcie, przyjaźń, koleżeństwo, ale i zdrowa rywalizacja między sobą. Takie było środowisko w mojej rodzinnej Łodzi wiele lat temu i cieszę się, że wiele z tych ideałów nie

przepadło. Niestety często nie widać tego w sieci, gdzie fora internetowe rozgrzewają się do czerwoności od dyskusji o niczym i tak zwanego "hejtu". Nie lubię tego słowa, ale czasami rzeczywiście najlepiej określa nastawienie niektórych internautów. Na szczęście jest to mniejszość i nadal Amigowcy mogą liczyć na pomoc kolegów po drugiej stronie kabla.

Od pewnego czasu powstaje nowa gra pod tytułem Verge World. Grupa RetroBones jest coraz bardziej doceniana i publikuje coraz śmieiej postępy swojej pracy, między innymi na Facebooku. Przeczytajcie koniecznie wywiad z członkami grupy, którzy opowiadają o tym, dlaczego gra powstaje i jaki chcą osiągnąć efekt końcowy. Z uwagi na ograniczoną ilość miejsca jest to skrócona część wywiadu, jaki przeprowadziliśmy z RetroBones. Szerszy tekst z większą ilością grafiki ukaże się w następnym numerze magazynu RetroKomp.

Zwróćcie też uwagę na artykuł o automatyce domowej, bo dawno nie mieliśmy tekstu opisującego tematykę tak odległą o wszechobecnych gier i dem scenowych. Ja również uważam, że Amiga powinna kojarzyć się przede wszystkim z jej szeroką bazą oprogramowania użytkowego. Dla mnie - tak samo w przeszłości, jak i dzisiaj - jest to kwintesencja platformy stworzonej przez Jaya Minera. Oczywiście, kijem Wisły nie zawrócimy i rozrywka zawsze będzie bardziej popularna niż ciekawe narzędzia dla Workbench. Jednak może przynajmniej niektóre osoby pamiętające "stare konsole", jak często Amiga czy C64 jest określana na aukcjach, zobaczą, że jest w nich coś więcej.

WYWIAD: Grupa Retro Bones, autorzy nowej gry Verge World

Grupa o nazwie Retro Bones pojawiła się niespodziewanie, a na pewno nieoczekiwany był jej start z nowatorską grą Verge World. Teoretycznie mieliśmy kilka gier realizujących efekt Voxel, ale można je policzyć na palcach jednej ręki. Ponadto, pozycje takie jak na przykład Shadow of the Third Moon były mało znane i wymagały rozbudowanej Amigi. Spytailiśmy członków grupy o plany dotyczące ich nowej gry.

Czym właściwie jest VergeWorld?

Kwahu (producer): VergeWorld: Icarus jest utrzymaną w stylistyce retro sci-fi grą typu shoot'em up/racer, w której wcielamy się w tajemniczego pilota statku powietrznego Icarus, znanego pod imieniem Nix. Poznajemy losy bohatera, staramy się o wsparcie ze strony wybranej przez nas frakcji, handlujemy, ulepszymy sprzęt i strzelamy do każdego, kto stanie na naszej drodze. Przedstawiony w grze świat to zdewastowana, pustynna planeta, na której fizyka nie działa zgodnie ze znanymi nam zasadami. W tych okolicznościach największa ludzka misja kolonizacyjna stara się przeżyć w miejscu, które z konieczności stało się ich nowym domem. To tak w wielkim skrócie.

Kiedy zrodził się pomysł stworzenia VergeWorld?

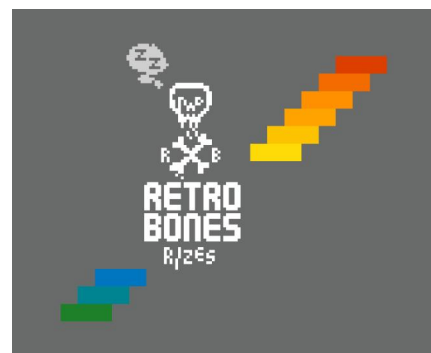
Kwahu: To był 2018 rok. Postanowiłem, po raz kolejny, zainstalować WinUAE i zagrać w gry, które pamiętałem z dzieciństwa.

Ponieważ miałem problemy z emulacją zainteresowałem się tematem sprzętowym i końcem 2018 nabyłem MIST FPGA. Bawiąc się sprzętem zagrałem w kilka gier i zapragnąłem spełnić młodzieńcze marzenia o stworzeniu własnej gry na Amigę. Postanowiłem, że chcę, aby było to coś nowatorskiego na tej platformie. Rozpoczęła się przygoda. Pierwszym krokiem było wykonanie szybkiego prototypu gry przygodowej w Unity 3D, która miała się opierać o renderowanie panoram 360 w czasie rzeczywistym, co miało dać relatywną swobodę poruszania w świecie 3D. Drugim krokiem było stworzenie prototypu gry voxelowej opartej o założenie, że nie będzie się dało skręcać i dzięki temu obliczenia będą mniej obciążać pocziwą Amigę 500. Mając taki prototyp rozpocząłem przygodę z programowaniem na prawdziwym sprzęcie. Odpowiednie środowisko udało się mi uruchomić dopiero z pomocą kolegów na wakacyjnym AmiParty 2019. To tam powstał pierwszy działający kod voxelu na Amidze 1200, który miał

zaledwie 4-5 klatek na sekundę, ale był kamieniem milowym, który spowodował intensywne prace w okresie wrzesień-październik 2019. Stworzone w tym czasie demo zaprezentowałem kolegom z Bad Bones.

Dawid (story designer): Moje pierwsze zetknięcie z projektem miało miejsce w październiku 2019, kiedy Kwahu przedstawił mi wczesną wersję silnika, przypominającą pojawiające się na ekranie zielone bloki, mniej więcej w jednakowym kolorze. Kiedy powiedział, że chce z tego zrobić grę początkowo wziąłem to za żart. Potem zostałem głównym projektantem świata gry i żarty się skończyły. Biorąc pod uwagę ilość czasu która upłynęła to patrząc na uzyskane obecnie efekty nie podlega dyskusji, że nastąpiły kolosalne postępy!

Szudi (msx&sound design): Do tworzenia projektu VergeWorld dostałem osobiste zaproszenie od Kwaha jako osoby odpowiedzialnej za warstwę muzyczną, wiedząc iż chyba aktualnie jako jeden z niewielu po świecie chodzących mam już siwe włosy i jakieś jeszcze blade pojęcie na temat robienia muzyki na maszynie 16 bit. Do projektu przystałem od razu. Pomimo iż technologia poszła do przodu człowiek dawno opuścił zapis trackowy i sampling 8 bit 22



KHZ na rzecz bardziej aktualnego softu z zapiętym tysiącem VST. Tak więc trzeba było odgrzebać stare wypociny i zainstalować wiekowy soft w celu tzw. "powrotu do korzeni". Tak więc po 15 latach zabawa w trackery zaczyna się na nowo.

Jakie były główne założenia podczas inicjowania projektu?

Kwahu: Pierwszorzędnym założeniem było stworzenie gry na Amigę, której jeszcze amigowcy świat nie widział. Śmiały plan, a mierząc zbyt wysoko łatwo się potknąć. Z tego powodu postanowiliśmy oprzeć całość projektu na solidnych podstawach technologicznych. Voxele potrafią wyglądać obtędnie!

Dawid: Zaczynaliśmy od projektu gry na Amigę, z różnymi trybami zależnymi od sprzętu, na którym ją odpalimy. Wiedzieliśmy, że metoda generowania poziomów za pomocą voxelów jest zarówno atrakcyjna wizualnie, jak również niezbyt szeroko stosowana - dla wielu osób może być nowością, chociaż stosowana była już od wielu, wielu lat. Ponadto, projekt miał być tak zaplanowany, by finalna gra zmieściła się na jednej dyskietce - chcieliśmy wszystko stworzyć tak, jak robiło się to dawniej, odtworzyć to doświadczenie.

Postanowiliśmy wykorzystać wszelkie sztuczki technologiczne, na które się natknęliśmy przez lata - nieocenioną okazała się Amigowa scena demo. Ludzie którzy dalej odkrywają sposoby na zwiększenie wydajności i atrakcyjności przekazu na "gołych" Amigach potrafią dosłownie tworzyć cuda. W toku pracy szybko okazało się jednak, że



implementacja ich rozwiązań niekoniecznie jest prosta! Bardzo ważnym założeniem było stworzenie świata, który można by rozwijać w ewentualnych kontynuacjach gry - koniecznym więc było spoglądanie na projekt w bardzo szerokiej perspektywie.

Ben Ghazi (lead programmer): W momencie dołączenia do projektu moim głównym założeniem było w miarę szybkie stworzenie grywalnego tech demo na dwie platformy - Amigę i Atari ST. Z tyłu głowy miałem jednak wytyczne długofalowe - napisanie silnika gry w taki sposób, by łatwo można było ją portować na inne platformy, ponieważ z czasem docierało do nas, że maszyn opartych o procesor 68k o parametrach zdolnych

pociągnąć VergeWorld jest dużo więcej niż dwie wcześniej wspomniane.

Szudi (msx&sound design):

Motywacja do powrotu do starego trybu tworzenia muzy - jak sprzed 20 laty. I znów mam brązowe włosy (i w czasie golenia nie naciągam skóry na gębę)!

Czym inspirowaliście się tworząc VergeWorld?

Kwahu: Z technicznego punktu widzenia moim punktem zaczepienia była amigowa demo scena. Kiedy zobaczyłem w jaki sposób ludzie generują krajobrazy za pomocą voxelów wiedziałem, że chcę to przekuć w grę.



Dawid: Jako główny projektant świata pamiętam, kiedy dostałem od Kwaha wytyczne - chciałby grę typu space opera, mówiącą o przygodach kogoś w rodzaju przemytnika, latającego po planecie pokrytej kanionami. Oczywiście inspiracją byłby Han Solo i Gwiezdne Wojny jako całość, jednak zgodziliśmy się, że zbyt duża liczba podobieństw wypada bardzo niekorzystnie w bezpośrednim porównaniu.

Mając do dyspozycji dwie pozycje z gatunku space opera jaki jest powód, dla którego gracz nie miałby pozostać przy znanych mu postaciach i lokacjach? Z tego powodu zmniejszyliśmy rozmach, zwracając się po trochu w stronę zarówno klasyków, jak i nowoczesnych produkcji.

Z mojej strony czerpałem garściami z takich pozycji jak Mandalorian, Diuna, dopiero co debiutującej gry The Outer Worlds oraz książkowej gry fabularnej Gasnące Słońca. Wszystko to po to, aby uzyskać pewien dualizm - z jednej strony gracz otrzymuje wiele elementów, które są mu znane i łatwo się z nimi identyfikować (nie czuje się zagubiony), a z drugiej przedstawiamy mu świeże spojrzenie na retro sci-fi.

Kuba (art director): Od strony graficznej inspiracji było bardzo wiele, ale przede wszystkim to te zaczerpnięte z gier. Dune II: Battle for Arrakis, Star Wars: X-Wing i Another World wywarły swego czasu na mnie największe wrażenie i pierwsze przyszły na myśl, kiedy Kwahu zaproponował współpracę nad projektem. Na te wspomnienia nałożyły się bardziej aktualne fascynacje - począwszy od prac

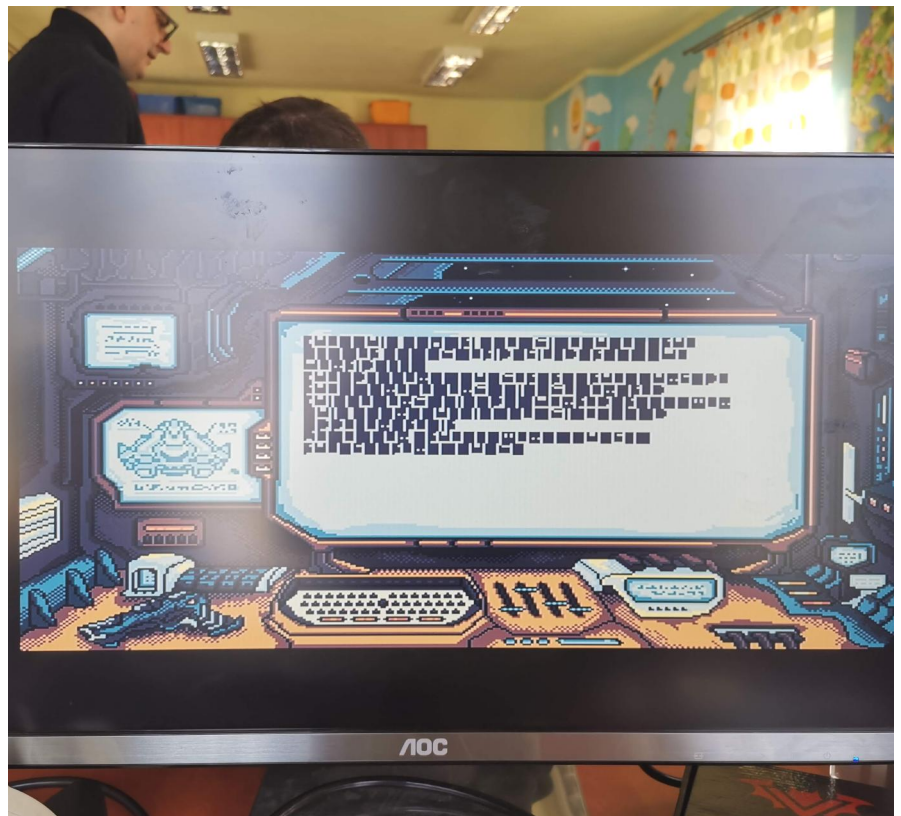
Gustavo Viselner i gier jak Superbrothers: Sword & Sorcery, po czysto artystyczne jak malarstwo Edwarda Hoppera czy Davida Hockneya.

Ben Ghazi: Od strony developmentu do moich inspiracji zaliczyć można w pierwszej kolejności projekt Handmade Hero - jego twórca Casey Muratori (swoją drogą, wielki fan Amigi) najlepiej w Internetach uczy jak pisać silniki! Podglądanie jego pracy dało mi bardzo dużo wskazówek i motywacji do działania.

Elf (graphic designer): Jeżeli o mnie chodzi, to inspirowałem się atmosferą filmu Dune Davida Lyncha, sztuką retro sci-fi z lat 70 i 80 oraz serią Fallout. Pragnąłem oddać klimat odrobinę mistycznego Mad Maxa w kosmosie. Od strony estetycznej za wzór stawiam gry typu Another World czy Beneath a Steel Sky. Dużą częścią inspiracji,

choć nie stricte graficznej, jest cała scena muzyczna New Retro Wave, odtwarzająca klimat sci-fi wyobrażany w latach 80 ubiegłego wieku. Niesamowicie klimatyczne "syntezatorowe" kompozycje mocno działają na wyobraźnię. Mam przede wszystkim na myśli takich artystów jak Dynatron, Perturbator, Neon Nox, Mega Drive czy Lazerhawk.

Szudi (msx&sound design): Aby podejść od strony muzycznej stwierdziłem iż trzeba by było zaaplikować sobie coś nieco postapokaliptycznego (korona) wirusa. Na początek połączyłem parę książek - trylogię Silos i zbiór opowiadań "Inne światy". Po zobaczeniu pierwszych screenów stwierdziłem że to raczej nie to... że można by było bardziej oprzeć się o klasycznie klimatyczny światów takich jak Perihelion (uwielbiam klimat z tej gry aczkolwiek gameplay mnie mocno drażni) i



wygrzebałem na strychu tryptyk Wiecznej wojny. Moje Dzieciaki zaczęły swojego starego postrzegać lekko no, jak to ująć - coś się dziwnego z ich starym dzieje! . No to zacząłem coś wypinkiwać spod klawiatury na kompie przez co w domostwie zaczęły nasilać wśród współdomowników nagłe bóle głowy,

W końcu team stwierdził iż pasowałoby tu coś mocniejszego - w stylu Perturbatora, Dynatrona, Turbo Knight'a ... no to do znudzenia rodzina musiała męczyć się z Retro synth wave z przeplatanką Spacesynth (wsio tylko nie Laserdance), gamingowych EDM ów Fat Rata :D i ambientowo - chilloutowych Sundial Aeon (demoscena w końcu).

W końcu stwierdziłem iż nie ma co się opierać innym trendom tylko iść na żywioł, bo z zapożyczanie w trackerach i tak nie wychodzi :) A co wyjdzie to ocenią potencjalni testerzy i gracze. Pierwszy mod - rzut na taśmę ale się ponoć spodobał. Mnie nie.

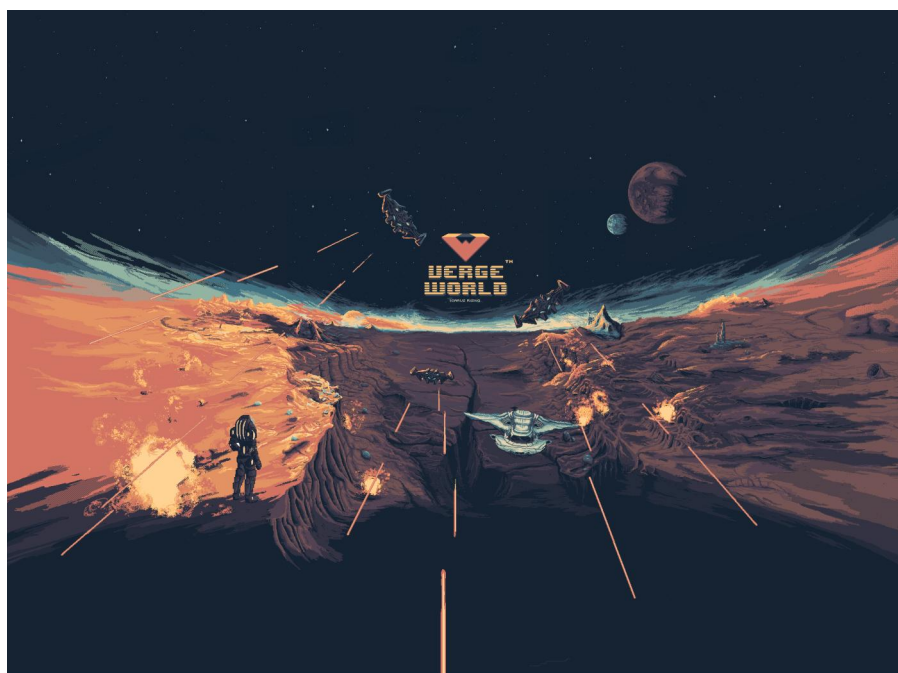
Z jakimi wyzwaniami zetknęliście się podczas rozwijania projektu?

Kwahu: Najtrudniej było wystartować. Dopiero na AmiParty w 2019 roku, z pomocą doświadczonych osób, udało się uruchomić kompletny toolchain, a zaraz potem - pierwszy voxelspace. Później wszystko poszło już z górki! Ale wiadomo, jak to w górach - z jednej góry na drugą.

Dawid: Zdecydowanie największym wyzwaniem dla mnie podczas projektowania świata i gameplayu okazują się ograniczenia dostępnego miejsca na nośniku.

Kuba: Podstawową kwestią do ustalenia była paleta kolorów. Zresztą dalej jest - nie uważam, że obecne rozwiązania są finalne. Sporo próbujemy i zmieniamy, aby uzyskać najlepszy efekt wizualny. Założeniem jest utrzymanie 16 kolorów na przestrzeni całego projektu, począwszy od samej gry na materiałach promocyjnych skończywszy. Z jednej strony jest to duże ograniczenie, jednakże z drugiej - to ograniczenie wzmacnia kreatywność rozwiązań.

Ben Ghazi: No cóż - z doskoku programować wersję tech demo gry na platformę, z którą nie miałem wcześniej styczności - to chyba



wystarczające wyzwanie. Nauka jest dla mnie wpisana w całość tego projektu, ale to bezcenne doświadczenie!

Szudi: Nowe przyzwyczajenia zdominowały zbytnio nasze umysły i umiejętności. Aktualnie w dobie programosów typu DAW (ja używam Cubase + pakiet Absolute) możliwości są praktycznie nieograniczone, a jak czegoś ci brakuje to praktycznie wszystko znajdujesz na necie. Tutaj sprawa się nieco komplikuje.

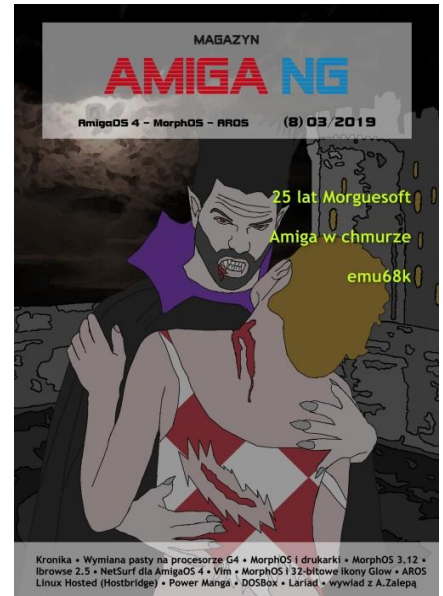
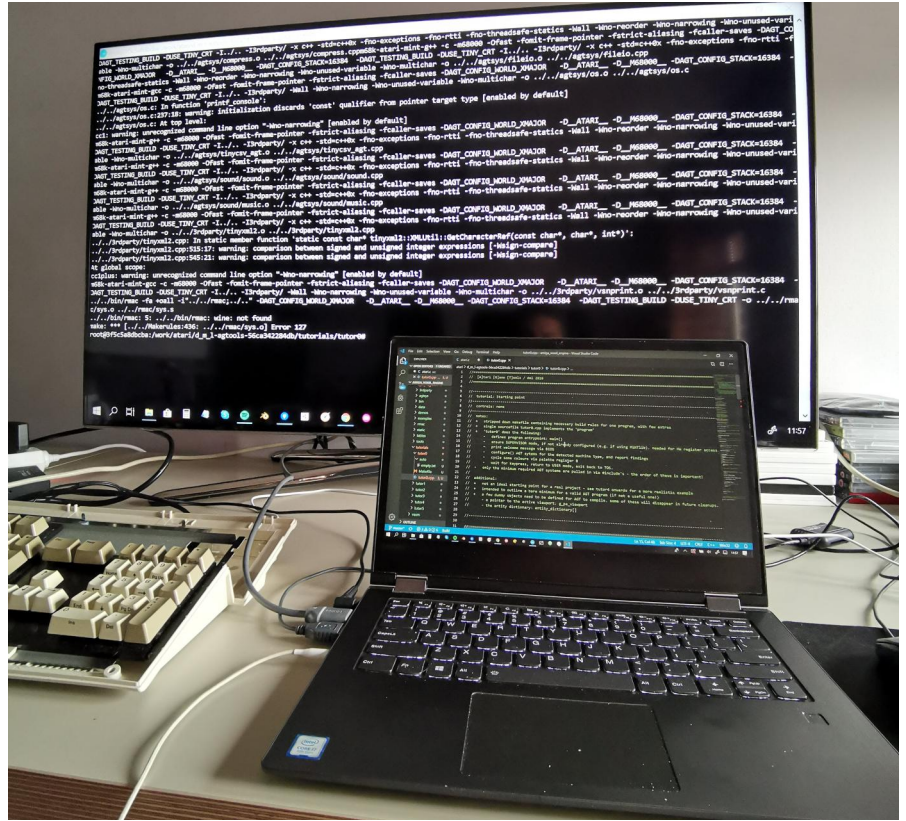
Przeskakujemy z nieograniczonej ilości ścieżek z zaimplemen-



towanym świetnym syntezatorem (nie będę reklamował), przepuszczonych przez pierdyldard kompresorów, reverbów, saturatorów w suchy, goły 8 bitowy, 4 ścieżkowy zapis, w którym możesz ... człowieku ... ująć ... tylko 3 oktawy..... eh. Na bonus dodam iż w Atari będzie synteza z trzema kanałami.

Elf: Z jednej strony zderzenie z ograniczeniami graficznymi Amigi i Atari mogą być wyzwaniem. Z drugiej - pixel art nie jest niczym nowym w dzisiejszych czasach - tak naprawdę ograniczenie palety do 16 kolorów jest zarówno przekleństwem, jak i błogosławieństwem. Daje bardzo specyficzny styl pracy, który osobiście bardzo mi odpowiada.

Dziękujemy za rozmowę.



Chcesz zagrać w demonstracyjną wersję gry VergeWorld? Kup magazyn RetroKomp 15 - dołączamy do niego 2 dyskiety z grą w wersji na Amigę oraz Atari ST.

Sprawdź też nasze pozostałe wydawnictwa. Piszemy już w trzech językach - polskim, angielskim i niemieckim. W przygotowaniu kolejne produkty!

AMIGA.net.pl
ŚWIAT KOMPUTERÓW RETRO

Kartka z kalendarza

W latach '80-tych i '90-tych w Polsce wiele osób konkurowało się sobą nie tylko, w tym, co potrafią robić na swoim komputerze, lecz jaki komputer w ogóle udało mi się zdobyć. Oczywiście Amigowcy dyskutowali nie tylko z pecetowcami, lecz początkowo przede wszystkim z użytkownikami Atari ST. Mówiło się dużo o Atari, bo w naszym kraju rozbudowana była sieć dystrybucji tego komputera - jak żadnego innego sprzętu domowego. Jednak na świecie komentarze były zgoła inne. Jakże? Przekonajmy się.

W kwietniu 1984 roku, w ciekawym amerykańskim czasopiśmie Creative Computing ukazał się artykuł Johna Andersona pod tytułem "Amiga Lorraine: nareszcie Atari nowej generacji". Dotyczył premiery Amigi, która nie była widziana jako zupełnie nowa platforma. Pewnie wszystko przez skojarzenie z Jayem Minerem. Oddajmy głos autorowi oryginalnego artykułu:

Kiedy zimowe targi CES w 1984 roku dobiegły końca, ludzie zaczęli pytać mnie: Jaki był hit tej imprezy? Trudno powiedzieć. Ogólnie rzecz biorąc, pokaz był krótki, przynajmniej w dziedzinie mikrokomputerów. Commodore zaprezentowało swoją nową maszynę, 264. Atari i Apple jak zwykle pokazywały swoje produkty, ale tym razem miały niewiele do powiedzenia. Nowego oprogramowania było pod dostatkiem, ale za wyjątkiem Relaxu z Synapse, nic tak naprawdę nie powaliło mnie z nóg.

Jeśli chodzi o "przebój" programu według mnie, to musi to być moje pierwsze spojrzenie na superkomputer od Amigi o nazwie Lorraine. Na stoisku nie było co prawda nigdzie śladu tej maszyny. Jednak, kiedy dostałem

zaproszenie, by wziąć udział w tajnej prezentacji, muszę przyznać, że szczęka mi opadła. Jeśli chodzi o mnie, to demo na sprzęcie Lorraine było wystarczającym powodem, by wybrać się do Las Vegas.

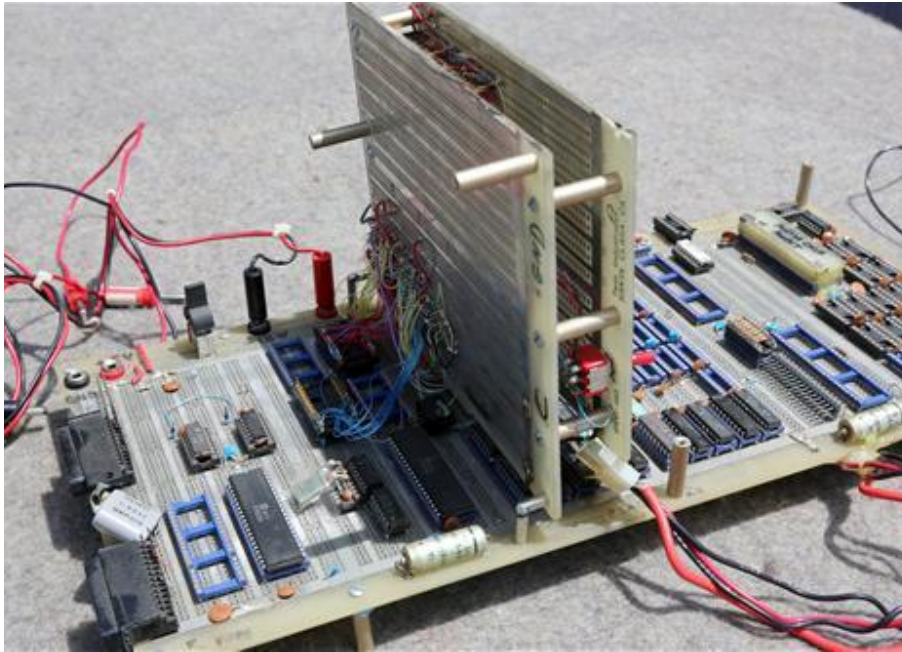
(...)

Około pół roku temu, Dave Morse - prezes Amigi, odwiedził nas w laboratorium, aby opisać nam Lorraine. Byliśmy podekscytowani, ale słyszeliśmy już wcześniej bardzo różne informacje. Życząc mu jak najlepiej, przedstawiliśmy własne sugestie i powiedzieliśmy mu, aby pozostał w kontakcie.

Cóż, tajny pokaz w centrum West Hall spełnił z nawiązką swoje obietnice. To, co zobaczyliśmy na Lorraine zrobiło wielkie wrażenie. Chłopcy z działu oprogramowania, łatwo rozpoznawalni po swoich zużytych Adidasach i radosnych uśmiechach, spędzili nieco ponad tydzień na opracowywaniu demo



Amiga w 1985 roku była prezentowana przez poważnych panów w garniturach.



Prototyp Amigi - komputer Lorraine. Ten sprzęt nie wcale nie wyglądał tak, jak wiele osób sobie to wyobraża. W porównaniu z nim dzisiejsze płytki FPGA to wzór elegancji.

na nowym komputerze. Było to utrudnione przez nieukończony system wejścia/wyjścia. Jednak udało im się napisać serię których programów, które pokazały moc Lorraine. To po prostu niesamowita grafika i dźwięk.

Obiecałem Dave'owi, że na razie nie powiem zbyt wiele o Lorraine. Wystarczy powiedzieć, że ma ona najbardziej zdumiewającą grafikę i dźwięk, jakie kiedykolwiek były oferowane na rynku konsumenckim. Jakiego rodzaju podstawę techniczną ma Amiga? Jest to zestaw układów scalonych VLSI zaprojektowany przez Jaya Minora - człowieka, który zaprojektował zestaw superukładów z oryginalnych maszyn Atari.

Mimo że Lorraine będzie posiadała wyjście NTSC, aby uzyskać najlepsze rezultaty, potrzebny będzie monitor RGB. Lorraine jest w stanie wyświetlać wielobarwne

animowane obrazy w czasie rzeczywistym na równi z animowanymi kreskówkami. Jej czterokanałowe, stereofoniczne wyjście dźwiękowe jest zaprojektowane tak, aby mogło być podłączone bezpośrednio do Waszego dotychczasowego sprzętu audio i może generować dźwięk porównywalny do dedykowanego syntezatora, jak również wiele dodatkowych efektów dźwiękowych i syntezowaną mowę.

Lorraine jest mała, elegancka i posiada klawiaturę podobną do tej z TI Professional (komputery firmy Texas Instruments - przyp. Red.) - jednym słowem, świetną. Oprócz gniazda na układy ROM, będzie zawierała napęd dyskowy o dużej gęstości 5,25" i będzie miała 128 KB pamięci RAM. I jest szybka - naprawdę szybka. To właśnie sprawia, że animacja oraz sprite'y generowane przez Lorraine są absolutnie najpotężniejsze, jaką

kiedykolwiek widziałem na domowym komputerze.

Chociaż pan Morse nie wyjaśnił, na jakim procesorze opiera się Lorraine, to na podstawie demo można się domyślić, że jest to przynajmniej układ 68000. Lorraine może obsługiwać wszystkie 16-bitowe systemy operacyjne i będzie kosztować mniej niż w pełni skonfigurowany IBM PCjr. A czy jest z nim kompatybilna?

Niestety nie, choć uważam, że jest to bardzo ważne. Według mnie, podstawowa konfiguracja Lorraine powinna być zgodna z PCjr. (...) W obecnej postaci, kompatybilność z IBM będzie wymagała dodatkowego procesora z układem ROM. Jeśli firma Amiga jest przewidująca, wbudują ten układ w produkcyjnym modelu Lorraine. Wtedy będą mieli zachowaną kompatybilność nie wymagającą żadnej konfiguracji.

Niezależnie od tego, że standard IBM jest mierny, z pewnością stanie się standardem domowym. Jestem sceptycznie nastawiony do komputerów o "specjalnych możliwościach", bez względu na to, jak imponujące mogą one być. Widzieliśmy, jak wydajne maszyny (choćby Atari), pomimo ich imponujących możliwości, odchodzą w zapomnienie. Amigo, proszę, nie dołączaj do tych przykrych szeregów, które zmarnowały technologiczną przewagę przez marketingowe bzdury.

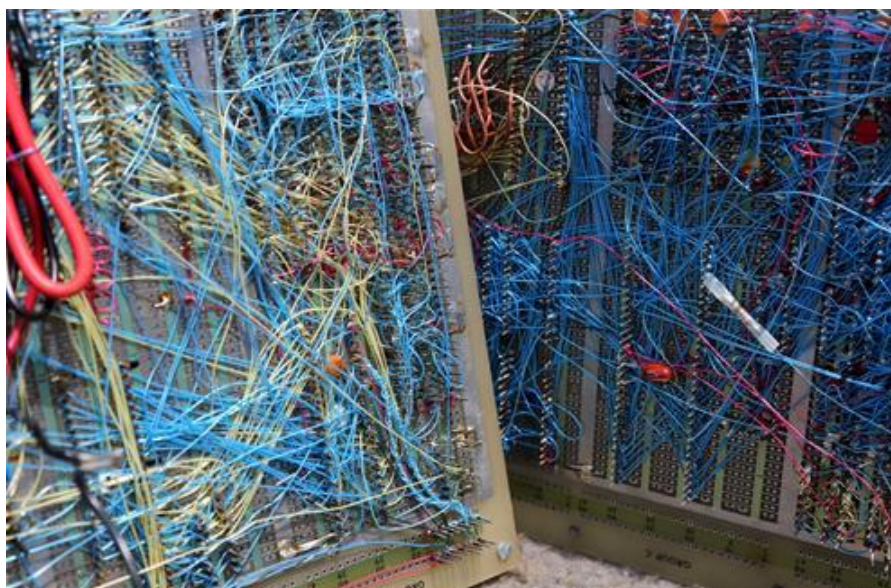
Amiga jest podobna do Atari również pod innym względem - musi przewyższyć swoją obecną reputację, zanim będzie mogła być traktowana poważnie. Firma musi

sobie sama powiedzieć: Ok, chłopaki. Zostawmy Joyboard za nami i bądźmy wiarygodni. Lorraine może być maszyną, na którą czekały setki tysięcy entuzjastów grafiki i dźwięku od czasu premiery Atari. Dzięki kompatybilności z PC Jr i odpowiedniemu marketingowi, Amiga Lorraine może okazać się wielkim sukcesem już w połowie lat '80-tych. Trzymajmy kciuki! (...)

Tu kończy się mój długi cytat. Przyznacie, że w tym tekście jest wiele odniesień i opinii niezbyt popularnych w naszym kraju. Amiga jest widziana nie jako oddzielna platforma, lecz raczej jako niesamowity sprzęt, który powinien zdobyć rynek



Nie ma się co dziwić, że tak spięte płyty elektroniczne nie działały zbyt dobrze, a twórcy bali się o awaryjność. Na szczęście wszystko się udało i niedługo później świat mógł poznać Amigę.



zarezerwowany przez PC Jr, czyli peceta w domowym wydaniu. Jak wiele z tych słów sprawdziło się wiele lat później. Pecety uzyskały możliwości Amigi i stały się najpopularniejsze - tak jak przewidywał autor artykułu. Niestety (lub stety) nie stało się to poprzez zgodność ze standardem peceta, ale za to mieliśmy piękne złote lata Amigi.

Ten tekst pokazuje też, jakie nastawienie do Lorraine miał rynek amerykański. Oczekiwano dołączenie do grupy wielkich tego świata i rozwinięcie peceta tak, aby jeden komputer mógł służyć do wszystkiego - pracy i zabawy. Ambicje projektantów Amigi były dużo szersze i może dlatego, gdy nasz komputer potrzebował pomocy w bankrutującej firmie Commodore, nikt nie wyciągnął

przyjaznej ręki. Chcieliście, to macie - zdają się mówić cienie przeszłości. Tak, czy inaczej - zdajmy sobie sprawę z faktu, że Amiga była skazana na integrację z wiodącym systemem lub na zapomnienie. Tym bardziej możemy cieszyć się z tego wspaniałego okresu, gdy nasz komputer rozwijał się dynamicznie i był jednym z najpopularniejszych sprzętem na rynku, przynajmniej w Polsce. Wygląda na to, że nie było innego wyjścia, Lorraine musiała podzielić los innych komputerów domowych. Szkoda mi tylko zespołu Jaya Minera, który włożył tak wiele wysiłku w stworzenie nowego standardu, zarówno sprzętowego, jak i programowego. Nowy właściciel Amigi, firma Commodore, nie podążyła za ambicjami twórców Lorraine. Dzisiaj możemy to ocenić bardziej obiektywnie i oddać hołd tym ludziom - przecież wielu z nich żyje. Jak to zrobić? Po prostu - używajmy naszych Przyjaciółek, bawmy się nimi, rozbudowujmy, grajmy w gry i piszmy nowe programy. Oby jak najdłużej.

Amiga 500: uratowana

Wielu z nas odnawia i naprawia Amigi, które wcześniej znalazły się na śmietniku. Nie inaczej było w moim wypadku. Trafiła do mnie pięćsetka, którą ktoś wcześniej spisał na straty. Porysowana, pomalowana flamastrami z niekompletną obudową, lekko uszkodzoną klawiaturą oraz strasznym stanie w środku. Gdy zobaczyłem ekran płyty głównej, byłem przekonany, że komputer nie będzie działał. Wręcz bałem się go włączyć - tyle było w środku brudu, kurzu i rdzy. Na szczęście ta historia kończy się happy-endem.

Obudowa Amigi nie była nawet w złym stanie, ale została pomalowana pisakami, które wydawały się trudne do usunięcia. Właściwie, nie bardzo rozumiem, dlaczego ktoś zadał sobie tyle trudu, bo zamalował cały napis AMIGA. Nic nie pomagało, tylko mechaniczne usunięcie szczoteczką z detergentem. Trwało to chwilę, a na koniec okazało się, że miejsce to nie jest zażółcone. Ktoś musiał to więc zrobić dawno temu, gdy Amiga była jeszcze w miarę nowa.

Osobiście bardzo mnie ciekawią takie historie, bo nasza Przyjaciółka miała zawsze dla mnie duszę, niezależnie od modelu i konfiguracji. Odnawiałem już przynajmniej kilkanaście komputerów (kolejne czekają w kolejce) i zawsze jestem zadowolony nie tylko z efektu, ale też z faktu, że mogę później oddać Amigę komuś, kto ją wykorzysta i będzie o nią dbał. No dobrze, nie zawsze oddaję komputer za darmo i często też sprzedaję, ale zawsze

robię to za niską cenę. Na przykład, moją poprzednią A500 z pamięcią 1 MB Chip, Kickstartem 2.0, zasilaczem, myszką i pudłem dyskietek sprzedałem dobrze znanemu koledze za 200 zł.

Wydaje mi się, że lepiej naprawić Amigę i sprzedać ją za mniejsze pieniądze - aby odzyskać włożony w to czas i trochę kosztów, niż trzymać niesprawną. Nigdy nie rozumiałem takiego sposobu "kolekcjonowania" sprzętu. Wróćmy jednak do mojej pięćsetki, bo też włożyłem wiele wysiłku w to, aby mogła cieszyć kolejnego (tym razem nowego) Amigowca.

Zacząłem od rzeczy najważniejszej, czyli wyjęciu płyty głównej z ekranu. Następnie lekko ją wyczyściłem, aby zorientować się, czy nie widać większych uszkodzeń. Nie jestem specjalistą w zakresie elektroniki, ale porównałem wygląd ze zdjęciami dostępnymi w sieci i wydało mi się, że "na oko" nic złego nie widać.

Wyjąłem wszystkie układy z podstawek, delikatnie przeczyściłem i włożyłem na nowo. Przy okazji stwierdziłem, że komputer ma Kickstart 1.2, więc moja teoria o dawnym pochodzeniu Amigi okazała się prawdziwa.

Po tych wstępnych czynnościach postanowiłem zaryzykować i spróbować uruchomić samą płytę główną - bez dodatków typu rozszerzenie pamięci, a nawet bez stacji dyskietek. Wiem, że wiele osób nie poleca uruchamiania starego sprzętu bez wymiany kondensatorów, ale nie jestem aż takim purystą sprzętowym. Szybko okazało się, że komputer uruchamia się, chociaż zajmuje mu to trochę więcej czasu niż się spodziewałem. Było to jednak tylko moje wrażenie, bo po podłączeniu stacji dyskietek wszystko wróciło do normy.

Oczywiście, po takim długim czasie, gdy Amiga leżała nieużywana, oryginalna stacja okazała się nie do końca sprawna. Generalnie odczytywała dyskiетки, ale pojawiały się często błędy, silnik przestawał się jakby kręcić, a na dodatek napęd robił bardzo dużo hałasu. Po podłączeniu sprawnej stacji wszystko było w porządku. Pozostał jednak problem wymiany stacji. Tak się złożyło, że posiadam kilka nowych (chyba można to nazwać NOS-em) pecetowych stacji firmy Alps. Trzeba było działać dalej.

Aby mieć pewność, że napędy są sprawne zakupiłem kiedyś zewnętrzny interfejs do podłączenia stacji dyskietek na jednej z popularnych aukcji w sieci. Nie wygląda on super elegancko, ale spełnia swoje zadanie i ma jedną wielką zaletę. Posiada dwa przetworniki, za pomocą których

można podłączyć stację amigową lub pecetową - bez żadnych przeróbek. Nie wiem, czy działa to dobrze dla wszystkich modeli stacji, ale interfejs sprawdzałem na kilku napędach Samsunga oraz Alpsa i wszystko było w porządku.

Płytkę interfejsu jest podłączana w zwykły sposób, do portu Disk Drive w Amidze. Niestety ma on też jedną wadę. Mianowicie, posiada przełączniki typu DIP pozwalające ustawiać numer zewnętrznej stacji. Amiga może ją widzieć jako DF1, ale również DF2 lub DF3. Przełączniki w mojej wersji "były, ale się zmyły", bo po którymś podłączeniu płytki bez zabezpieczenia przestały dobrze kontaktować. W związku z tym stacja raz była widziana w systemie, a raz były problemy z odczytywaniem danych. Dlatego pozbyłem się przełączników i przylutowałem jedną ze "zworek" tak, aby była zawsze w pozycji DF1. I tak niczego więcej od tego interfejsu nie oczekuję, a w ten sposób działa idealnie.

Wracając do samej Amigi: chciałem ją dość dokładnie przetestować, więc podłączyłem emulator Gotek i uruchomiłem Workbench 1.3, nieśmiertelne narzędzia typu X-Copy oraz kilka przykładowych gier. O ile system zadział od razu, to gry miały problem z działaniem na pamięci 512 KB, ale to normalne zachowanie. Udało mi się jednak uruchomić okno CLI i sprawdzić klawiaturę. Niestety, nie obyło się bez kolejnych problemów.

Niektóre klawisze nie chciały działać. Zauważyłem też, że kilka z nich ma uszkodzenia, jakby zostały



Fragment obudowy po wstępnym czyszczeniu. Widać, że zamalowane fragmenty nie stały się żółte na skutek upływu czasu.

dotknięte czymś na gorąco. Jasne, że wygląd klawisza nie ma nic wspólnego z działaniem membrany i tak dalej, ale jednak widać było, że poprzedni właściciele nie przejmowali się klawiszami. Pewnie, jak większość pięćsetkowiczów, widzieli Amigę głównie w roli konsoli do gier i klawiatura nie była im poprzednia.

Na szczęście uszkodzenia obejmowały tylko 3-4 klawisze, ale jednak musiałem coś z tym zrobić. Na początek postanowiłem rozłożyć klawiaturę i zobaczyć, czy zwykłe czyszczenie nie zatałwi sprawy. Muszę przyznać, że była to najtrudniejsza operacja, a na pewno najbardziej żmudna. Po ciężkich bojach udało mi się jednak wszystko wyczyścić i złożyć z powrotem. Okazało się, że teraz wszystkie klawisze działają. Hura!

Warto tutaj zwrócić uwagę, że do tej pory nie musiałem praktycznie nic naprawiać. Wystarczyło czyszczenie, poprawienie styków, kontaktów i ponowne delikatnie skręcenie wszystkiego. Dochodzę do wniosku, że pięćsetka jest prawie jak ruski czołg - "gniotsa nie łamiotsa". To oczywiście żart, ale z mojego doświadczenia jest to sprzęt bardziej żywotny od kolejnej generacji Amigi, czyli A600 i A1200. Dodatkowo łatwiej go naprawić, nawet metodami prób i błędów w domu, prawie bez użycia narzędzi.

Mając już działającą Amigę stwierdziłem, że zamontuję w środku Goteka. Nie chciałem jednak ciąć obudowy w miejscu oryginalnej stacji, bo przecież mogła się przydać do montażu napędu przerobionego z peceta.



Jeden z uszkodzonych klawiszy oraz dolna część obudowy. Widać, że klawiatura lub obudowa była wymieniana, tak więc Amiga nie jest fabrycznym zestawem.

Postawiłem sobie za zadanie montaż obu rzeczy wewnątrz obudowy tak, aby można było z nich korzystać jednocześnie - ze stacji DF0 i DF1.

Właściwie, czemu nie zamontować Goteka w środku wycinając tylko trzy małe otwory? Będą mało widoczne, nie zepsują estetyki obudowy i będzie można je umieścić w najbardziej dla mnie wygodnym miejscu, czyli po lewej stronie - "naprzeciwko" oryginalnej stacji dyskietek. Jak pomyślałem, tak zrobiłem.

Nie jest to jednak tak zupełnie bezproblemowe i to z kilku powodów. Po pierwsze, przyciski Goteka są dość krótkie i po włożeniu do obudowy, nie wystają na tyle, aby można było z nich wygodnie

korzystać. Wystarczy przykleić jakikolwiek plastikowy czy gumowy element i sprawa jest załatwiona.. Druga sprawa to wyświetlacz pokazujący aktualny numer obrazu dyskietki. Ja z niego zrezygnowałem, bo nie chciałem niszczyć bardziej obudowy, ale czasami nie mam pewności, czy przełączyłem już slot na określony numer, czy jeszcze nie. Na szczęście używam Goteka głównie do nagrywania dyskietek i gram raczej z "normalnych" nośników, więc nie jest to duży problem.

Ktoś, kto chce używać tylko Goteka powinien zastanowić się nad przedłużeniem oryginalnych przewodów, aby wyświetlacz mógł znaleźć się gdzieś obok. Trzeba też jakoś umocować płytke Goteka w obudowie. Najlepiej wyjąć ją z

oryginalnej obudowy, która nie będzie przydatna - wręcz przeszkadza z uwagi na nietypowe wymiary i "podcięcie" w górnej części.

Aby wszystko się dobrzetrzymało po prostu wkręciłem dwie małe śrubki w górną część obudowy, wykorzystałem przy tym otwory na płytce Goteka. Trzeba tylko pamiętać, aby śrubki były krótkie, bo inaczej będzie je widać na zewnątrz, a to jest już bardzo nieestetyczne. Plastik obudowy nie jest też bardzo słaby i punkty do wkręcenia najlepiej przygotować na gorąco. Użyłem do tego małego szpikulca, którym naruszyłem delikatnie plastik, a resztę dopasowałem jeszcze mniejszym pilnikiem. Wygląda to dość elegancko, a od zewnątrz przeróbka jest niewidoczna. I właśnie o to chodziło.

Oczywiście zarówno Goteka, jak i zwykłą stację dyskietek podłączamy taśmami sygnałowymi i przewodami zasilającymi. Wystarczy użyć takich samych jakie normalnie mamy w Amidze. Jedna taśma musi być dłuższa, a przewód zasilający osobiście wolałem rozgałęzić, niż korzystać z gniazda na zewnętrznym interfejsie. Jest to wygodniejsze i dzięki temu na zewnątrz obudowy mamy tylko fragment taśmy sygnałowej, czyli jest elegancko.

Aby było jeszcze lepiej, chcę kupić wtyk D-Sub 25-pinowy z przeznaczeniem na taśmę i "nabić" go z jednej strony tak, aby na zewnątrz widoczna była tylko mała wtyczka. Jednak wymaga to drugiego kabla, który podłączymy do gniazda interfejsu, a płytke zamocujemy w środku, podobnie jak samego Goteka. W tej chwili

trudno kupić wtyk D-Sub żeński, dlatego wykorzystam do tego celu starego "śledzia" od płyty pecetowej typu AT. W takim sprzęcie porty takie jak Serial i Parallel były mocowane za pomocą taśm sygnałowych biegnących od portu na płycie głównej do tylnej części obudowy. Można to więc łatwo wykorzystać. Jest to mój plan na przyszłość.

To, co jest ważne przy montażu stacji dyskietek to jej wysokość. Mój napęd Alps jest na tyle wysoki, że po zamocowaniu za pomocą oryginalnych tulejek na odpowiednią wysokość, nie mogłem zamknąć obudowy. Wystarczyło jednak zdjąć górną część metalowej obudowy stacji, aby wszystko zaczęło pasować. Nie trzeba tu się specjalnie przejmować o kurz zbierający się w stacji, bo ogranicza do z kolei górną część obudowy samej Amigi.

Trzeba również dostosować przycisk stacji, bo oryginalny jest zbyt mocno przesunięty w lewo - patrząc z boku

pięćsetki. Można to zrobić ucinając go mniej więcej w 1/3 długości i przyklejając bardziej z boku. Gdy zrobimy to dokładnie, z zewnątrz nie będzie widać przeróbki, a całość dobrze się trzyma. Można też inaczej - połączyć oba elementy na gorąco. Wygląda to trochę mniej elegancko, więc wypada taki przycisk pomalować, ale spełnia swoje zadania. Jest to mocne

połączenie, tylko znowu - liczy się precyzja, bo są to jednak bardzo małe części.

To właściwie wystarczy, aby nasza Amiga wczytywała obrazy dyskietek z Goteka, a jednocześnie pozwalała na nagrywanie dyskietek na stacji DF1. Ja jednak lubię mieć swoje ulubione gry na fizycznych dyskietkach, więc traktuję Goteka



Kolory wyświetlacza emulatora Gotek oraz Power LED pasują do siebie. Wycięty otwór wymaga jeszcze obróbki, aby zlikwidować nierówności.

jako kopię swojej kolekcji. Dlatego dodatkowo zamontowałem w obudowie boot selector, kupiony zresztą na kolejnej internetowej aukcji. Instalacja jest bardzo łatwa, po prostu wyjmujemy układ CIA B i wkładamy go w podstawkę, która wyposażona jest w przełącznik. Jedna pozycja ustawia jako pierwszą stację Goteka, druga - normalną stacją dyskietek.

Jest to znana sprawa ze starych czasów, ale dawniej miałem stację 5,25 cala, dodatkowo boot selector był jedynym przełącznikiem w obudowie. Teraz mam jeszcze dwa dodatkowe, które doprowadzone są



do Kickstartu typu "4-w-1". Mogę dzięki temu uruchamiać Amigę w klasycznym trybie Workbench 1.3, systemu 2.0 (jak Amiga 500+) lub 3.0/3.1. Czy to ważne? Niekiedy tak, bo niektóre dyskietki nie wczytają się na Kickstart 1.3, który nie obsługuje formatu Fast File System (FFS).

Druga sprawa to programy, które rzadko wymagają systemu 3.1, ale często zadowolają się wersją nie starszą niż 2.0. Poza tym, Kickstart 2.0 pozwala uruchomić Workbench 2.1, który posiada wiele rozszerzeń zbliżających go do błatu znanego z Amigi 1200. Jako nowe elementy mamy tu, między innymi, format AmigaGuide i system polskiej lokalizacji, który jest wspierany przez słynny pakiet WFMH Locale PL. Z Amigi 500 robimy wtedy model Plus lub wręcz "prawie" Amigę 600.

To wszystko jest oczywistą sprawą dla Amigowców rozbudowujących swoje pięćsetki w latach '90-tych. Ja miałem wtedy tylko Commodore 64, a Amiga była u mnie przelotnie. Z tego powodu, wiele rzeczy jest dla

mnie nowych lub znanych tylko ze słyszenia. Na przykład, moja Amiga ma teraz 512 KB pamięci Chip i tyle samo pamięci Fast i chciałbym zamontować jeszcze jeden przetwornik, dzięki któremu całość będzie mogła być widziana jako 1 MB Chip RAM. Po co? Wiele gier nie chce działać bez 1 MB pamięci graficznej. Moja płyta to Rev. 6A, dlatego - z tego co się orientuję - muszę po prostu przelutować dwie zworki, ale o tym opowiem Wam następnym razem.

Myślałem także o wybieleniu obudowy, ale nie polecam tego robić. Wiele osób już przekonało się, że żółty kolor po jakimś czasie wraca, a plastik robi się bardziej kruchy. Lepiej pomalować obudowę na ulubiony kolor, choć wymaga to wprawy, aby ładnie wyglądało. Sam zostawiam obudowę w oryginalnym stanie, jedynie chcę ją jeszcze lepiej przeczyścić i być może lekko zeszlifować uszkodzone miejsca.

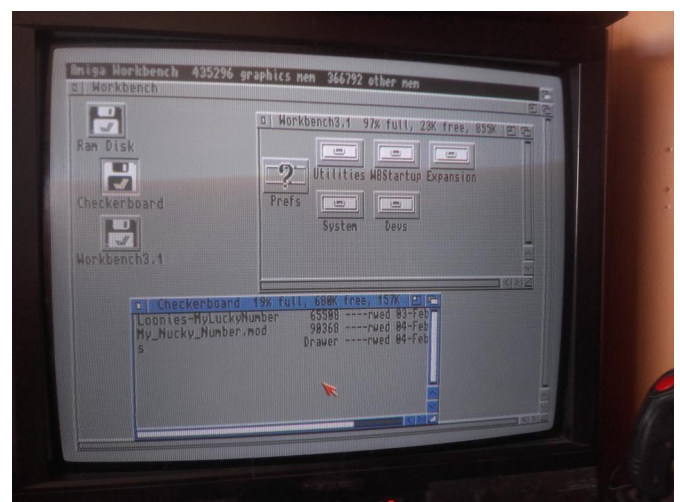
Jeszcze jedna uwaga: jeśli chcecie wykonać podobne przeróbki,

uważajcie przy skręcaniu obudowy. Najlepiej pozostawić środkowe śrubki niedokręcone, albo wręcz ich nie wkręcać. Mam tu ma myśli taśmę sygnałową, która wychodzi z tylnej części obudowy i lepiej, aby nie była zbyt mocno przyciśnięta. Co prawda, nic nie powinno się stać, w końcu obudowa Amigi jest plastikowa, ale lepiej uważać, niż później żałować.

Przedstawiłem swoją historię ulubionej pięćsetki, jakich w sieci można znaleźć na pęczki. Każdy Amigowiec dopieszcza swój komputer i uważa go za najlepszy. Pewnie tutaj właśnie mieści się dusza Amigi, o której tak wiele się mówi. Moim drugim ulubionym sprzętem jest Commodore 64, którego staram się wykorzystać z połączeniu z A500. Tutaj też można osiągnąć ciekawe rezultaty. Można nawet użyć emulatora C64 po to, aby odczytywać dane z prawdziwej stacji dyskietek Commodore 1541 na Amidze. Postaram się o tym napisać niedługo w nowym artykule. Amiga Rulez!



Zewnętrzny interfejs stacji dyskietek w porcie Disk Drive. Stacja dyskietek jest zamontowana wewnątrz Amigi.



Obie stacje dyskietek działające na blacie Workbench. Można je używać zarówno w programach zgodnych z systemem, jak i Not-DOS.

Karta GVP A530

Właściwie słowo "karta" nie do końca tu pasuje, bo jest to rozszerzenie podłączane do portu Expansion Amigi. Wizualnie jest to więc całe urządzenie wyposażone w dużą obudowę pasującą stylistycznie do pięćsetki. A co znajdziemy w środku? Oprócz samej płytki z procesorem i pamięcią, mamy tu kontroler SCSI-2 z możliwością montażu dysku twardego wewnątrz obudowy.

Dzisiaj, jeśli komuś uda się zdobyć A530, zazwyczaj jest to albo czysty przypadek, albo trzeba na nią wydać spore pieniądze. Instalacja nie jest trudna, bo w sieci można znaleźć obrazy oryginalnych dyskietek, które były dołączane do karty. Na przykład tutaj:

<http://amiga.resource.cx/exp/gvp530>

Urządzenie zostało wyprodukowane w 1992 roku i producent jak najbardziej chciał uprościć obsługę. Na twardym dysku zainstalowano oprogramowanie wykrywające Kickstart w Amidze - wtedy były to wersje 1.3 lub 2.0. W zależności od wersji, przy pierwszym uruchomieniu użytkownik mógł skorzystać z programu instalującego Workbench w odpowiedniej wersji. Z tego, co pisali użytkownicy produktów firmy GVP zza oceanu, działało to bardzo dobrze. Szkoda, że tak proste, ale i ciekawe rozwiązania nie były później zbyt często wykorzystywane.

Nawiasem mówiąc, dawniej większość firm amigowych sprzedawała A530 z dyskami Maxtor 120 MB, które mówiąc delikatnie, nie należały do najcichszych. Nawet późniejsze większe dyski tej firmy, choć oceniane były pozytywnie, to

Produkty amerykańskiej firmy Great Valley Products są szeroko znane w świecie Amigi. W latach '90-tych wypowiadano się o nich prawie wyłącznie w samych superlatywach. Chciałbym opisać swoje doświadczenia z kartą turbo do Amigi 500 o oznaczeniu A530. Jak łatwo można się domyślić, jest to karta z procesorem 68030 o taktowaniu 40 MHz.

powodowały wibracje i mocno hałasowały. Karta pozwala na mapowanie Kickstartu w pamięci, nawet jeśli zamontowana jest ekonomiczna wersja procesora, czyli 68EC030.

Firma GVP sprzedawała kartę z pamięcią 4 MB, ale można było na niej zamontować maksymalnie 2 moduły SIMM po 8 MB każdy. Dostępne były też specjalne SIMM-y GVP o objętościach 1 i 4 MB, ale dzisiaj są praktycznie nie do zdobycia. Wielkości pamięci nie mogą być mieszane, co jest sporym ograniczeniem. Możliwy jest także montaż koprocesora 68882, który działa synchronicznie w stosunku do procesora, czyli ma zegar 40 MHz.

Amiga z kartą A530 jest prawie 10 razy szybsza od zwykłej pięćsetki, natomiast przy mapowanym Kickstartie wydajność wzrasta dodatkowo o ok. 10 procent. Takie

wyniki pokazuje program SysInfo, który jest średnio wiarygodny jeśli chodzi o prezentację konkretnej szybkości sprzętu. Wydaje mi się jednak, że dobrze mierzy różnice szybkości, o ile jest uruchamiany w tej samej wersji.

Dysk twardy działa z prędkością ok. 1 MB na sekundę, co dzisiaj nie jest żadnym wyjątkowym wynikiem, ale na Amidze 500 jest to wystarczające - szczególnie jeśli korzystamy z tak małej objętości dysku.





Szybkość pracy karty GVP 530 jest wystarczająca do wielu zastosowań, nie tylko rozrywkowych.

Na przedniej ścianie karty znajduje się przełącznik Turbo, dzięki któremu można włączać i wyłączać akcelerator w celu zapewnienia pełnej kompatybilności ze starymi grami.

Niestety, nie można wyłączyć procesora 68030 bez utraty pamięci RAM i dysku twardego. Z tyłu obudowy znajduje się natomiast w pełni przelotowe 25-pinowe gniazdo SCSI. Urządzenie jest zasilane zewnętrznie, dzięki temu nie przeciążymy naszej poczciwej A500.

Czas na małe podsumowanie. Podoba mi się



tutaj szybkość, przelotowy port SCSI, opcja wyłączenia karty, jak również łatwość instalacji - o ile będzie korzystać z oryginalnej dyskietki. Można zainstalować dodatkowo koprocesor oraz powiększyć pamięć po prostu wkładając inne moduły SIMM. Wiadomo, że dzisiaj nie

jest to takie proste, bo tych pamięci dawno nie ma w sprzedaży, ale wszystkie karty turbo do Amigi z epoki lat '90-tych mają SIMM-y. Trudno pod tym względem czynić zarzuty producentowi.

Jest jednak kilka innych rzeczy, które mi się specjalnie nie spodobały. Produkty firmy GVP zawsze miały wysokie ceny, dodatkowo specjalne SIMM-y GVP były ok. 2-krotnie droższe od zwykłych. Dzisiaj nie ma to znaczenia, bo ceny wyznaczają "handlarze" internetowi, ale warto powiedzieć więcej o obsłudze pamięci. Mianowicie, w niektórych kartach GVP zwykłe pamięci

Karta GVP 530 podłączona do Amigi 500 po zdjęciu obudowy. Widoczny dysk twardy SCSI oraz układy pamięci.

w ogóle nie działały i trzeba było koniecznie kupić od razu odpowiednią konfigurację, praktycznie bez możliwości rozszerzenia. Tutaj można włożyć oba rodzaje pamięci, co jest pozytywne, bo daje użytkownikowi swobodę działania.

Tak wygląda w pełni wyposażona karta, z zamontowaną kartą sprzętowego emulatora komputera PC 286.

Automatyczny rozruch na Raspberry Pi

Karta ma też pewną nieprzyjemną cechę. Gdy mamy tylko 1 MB pamięci RAM, występują problemy z transmisją danych na dysku. Pojawiają się po prostu uszkodzenia plików. W sieci znalazłem informacje, że podobne rzeczy dzieją się podczas przesyłania danych poprzez port szeregowy lub równoległy.

Dlatego korzystając z A530 dobrze sprawdzać od czasu do czasu sumy kontrolne plików, przynajmniej na początku po instalacji. Nie spotkałem się natomiast z opiniami o problemach na komputerach z pamięcią 4 MB lub większą. Być może producent stwierdził, że jest to niewielki problem i nie ma sensu pracować nad jego wyeliminowaniem, jednak nie jest to eleganckie zachowanie.

Kiedyś wielu Amigowców zastanawiało się, czy zmieniać A500 na A1200, czy też rozbudowywać pięćsetkę o kartę podobną jak GVP A530. Odpowiedź nigdy nie była jednoznaczna, bo zbyt wiele zależy od indywidualnych oczekiwań.

Jednak jedno trzeba przyznać: z pewnością procesor 68030 z zegarem 40 MHz to zdecydowanie więcej niż 68020 14 MHz. Sytuacja wygląda inaczej, jeśli przyjrzymy się szybkości operacji graficznych, ale to już nie jest wina jakiegokolwiek producenta kart turbo.

Amibian to bardzo przyjemny emulator Amigi, który daje spore możliwości konfiguracji, mimo uproszczonego menu, jakie widać na ekranie. Od pewnego czasu próbowałem go ustawić tak, aby automatycznie wczytywała się jedna z moich konfiguracji, mianowicie Amiga 1200 z kartą graficzną. Program uparcie jednak wyświetlał menu główne. Już myślałem, że zainstalowałem jakąś "dziwną" wersję Amibiana, a prawda okazała się dużo bardziej prozaiczna. Wszystko przez moją niewiedzę.

Sądziłem, że wystarczy użyć opcji "Show GUI on startup", a dokładniej wyłączyć ją. Niestety to nic nie zmieniło, bo opcje konfiguracji ciągle pojawiały się na początku. Przez jakiś czas nie interesowałem się tym tematem, ale w końcu postanowiłem rozwiązać problem. Okazuje się, że trzeba zrobić dodatkowo kilka innych, ale bardzo prostych, czynności.

Pierwsza sprawa to wyjście z emulatora za pomocą opcji "Quit", aby wrócić do konfiguracji w terminalu. Następnie wpisujemy komendę:

uaeconf0

Spowoduje ona, że automatycznie będzie uruchamiana konfiguracja "zerowa", czyli "config0", którą widać pod zakładką "Configurations". Można też wpisać inny numer na końcu, wszystko zależy od tego, ile ustawień mamy zapisanych na

karcie pamięci. Domyślna konfiguracja to zawsze "zero". Co ciekawe, komenda ta nie jest nigdzie podkreślona w opcjach, gdy wejdziemy do terminala. Jednak można uzyskać dostęp do krótkiego opisu wybierając "7. Emulators setup", a później "emulatorconfig". Szkoda, że trzeba się trochę namęczyć, aby znaleźć tak przydatną funkcję, ale na szczęście robimy to raz i później nasza ulubiona konfiguracja Amigi wczytuje się automatycznie.

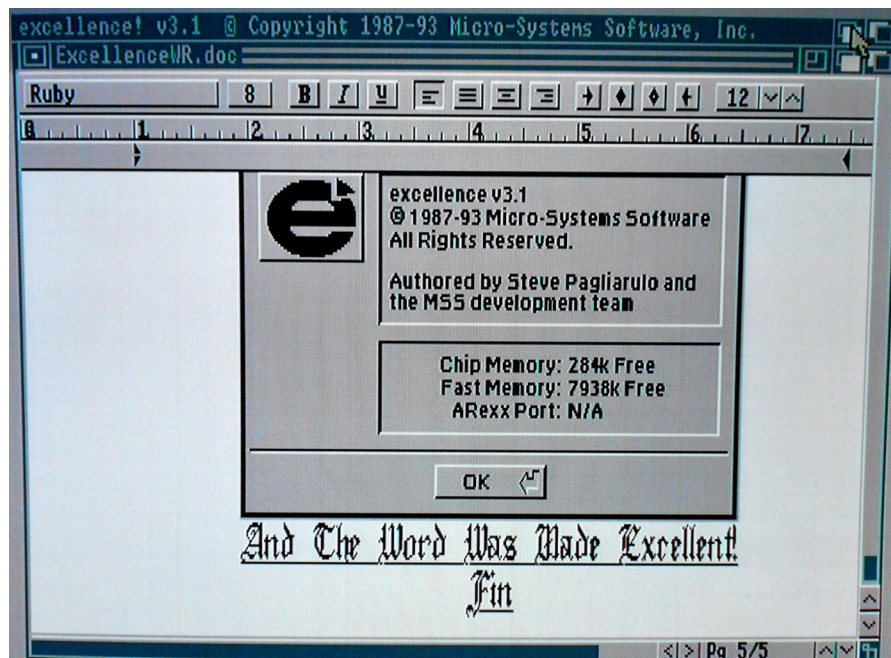
Tak mała zmiana w ustawieniach sprawia, że emulator uruchamia się bardziej naturalnie i dla mnie jest to świetna alternatywa dla drogiego i zawodnego sprzętu klasycznego. U mnie Amibian działa na Malince 3B+, ale teraz polecam od razu zaopatrzyć się w najnowszą Raspberry Pi 4. Emulator Amibian ciągle się rozwija i dodawane są do niego funkcje związane także z innymi komputerami retro.

Workbench 1.3: Excellence

Excellence to program, który wpadł w moje ręce całkiem przypadkowo. Znalazłem o nim informacje na jednej z moich starych płyt z oprogramowaniem, ale tylko jako krótki opis. Dopiero później zacząłem dowiadywać się o nim więcej. Okazuje się, że jest to mały znany u nas procesor tekstu o sporych możliwościach. Najciekawsze jest to, że działa nawet na Kickstarcie 1.3.

Program w swojej historii nie rozwijał się bardzo dynamicznie. Mam tu na myśli fakt, że pierwsza wersja Excellence nie różniła się zbytnio od tej ostatecznej. Wersja 1.11 oferowała obsługę drukarki PostScript, możliwość drukowania i wyświetlania do ośmiu kolorów, automatyczne generowanie

indeksów i spisów treści, obsługę grafiki IFF, bibliotekę językową zawierającą 70 tys. słów, sprawdzanie pisowni (90 tys. słów) oraz 300-stronicową instrukcję obsługi. Program od początku był jednak bardzo intuicyjny. Jeśli jesteś przyzwyczajony do współczesnego edytora tekstu, nie będziesz miał



Program Excellence w Polsce jest prawie nieznan. Posiada spore możliwości i przewidziano go do działania nawet na Workbenchu 1.3.

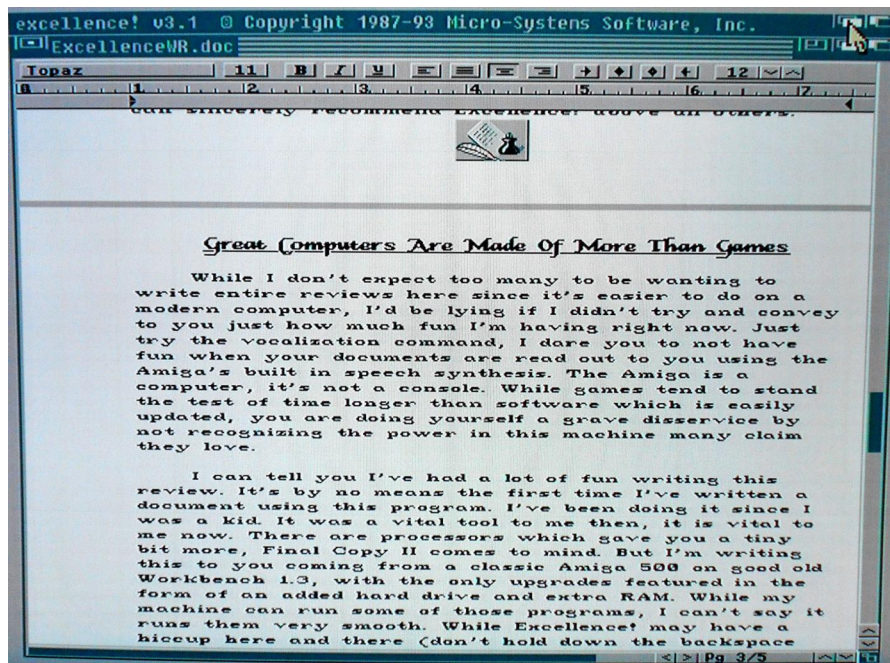
większych problemów z obsługą. Gdy pomyślimy o czasie, w których ten program powstał, szybko okaże się, że jego możliwości są naprawdę imponujące.

Warto dodać, że nie był to pierwszy amigowy edytor tekstów z podglądem układu strony. Pierwszy w historii to program o nazwie Textcraft. Wraz z wydaniem 2.0 program Excellence został ulepszony. Był bardziej stabilny, charakteryzował się szybszym scrollingiem, a szybkość wyświetlania tekstu była stała, niezależnie od tego jak szybko był wpisywany tekst. W wielu podobnych programach wydajność była dużo mniejsza. Do opcji drukowania dodano orientację poziomą, a także obsługę pamięci wirtualnej. Dzięki temu dokumenty mogły mieć wielkość nawet ok. 1000 stron. Dodano również słownik o dość dużym rozmiarze, który osiągnął objętość 140 tys. słów. Zrezygnowano z wersji dla komputerów z 512 KB, a dodane funkcje zaczęły wymagać 1 MB RAM.

W wersji 3.0 wygląd interfejsu programów został zmieniony na podobny do stylu Workbenchu 2.0, zachowując przy tym kompatybilność z wersją 1.3. Dodano tworzenie kolejnych kopii zapasowych w postaci zapisów wykonywanych w określonym czasie. Jeśli zajdzie potrzeba znalezienia się ponownie w istotnej części dokumentu, można skorzystać z dołączonej funkcji zakładki. Nie trzeba się już denerwować, że niekiedy po otwarciu okna z prośbą o zapisanie pliku program się zawiesi. Można też sprawdzić pisownię podczas przewijania dokumentu lub

bezpośrednio podczas pisania. To spore zmiany.

Program domyślnie pracuje w trybie Hires, czyli 640x256 lub 640x200 pikseli. Czcionki są bardzo dobrze zaprojektowane do tej rozdzielczości. Można też przetrząść ekran na tryb Interlace, czyli odpowiednio - 640x512 lub 640x400 pikseli. Czcionki są odczytywane z katalogu systemowego, co w programach dla Workbencha 1.3 wcale nie było regułą. Excellence obsługuje również systemowy schowek (Clipboard). Możemy otworzyć kilka dokumentów jednocześnie i przenosić elementy pomiędzy nimi. Jeżeli chcecie zobaczyć, jak wygląda nowoczesny program pisany dla "gołej" Amigi 500, Excellence jest bardzo dobrym wyborem.



Przykład sformatowanego dokumentu. Zwróćmy uwagę, że wygodny podgląd kartki zazwyczaj oferują programy wymagające Kickstartu 2.0 lub nowszego.

Ciekawe dodatki do Workbencha

W czasach Workbencha 1.3 mało kto myślał o rozszerzaniu jego możliwości. Mieliliśmy zwykle 1-2 MB pamięci RAM i nawet jeśli dysponowaliśmy twardym dyskiem (bardzo rzadko) to Workbench pozostawał bez większych zmian. W końcu liczył się każdy kilobajt wolnej pamięci. Przez lata powstało jednak wiele programów rozszerzających możliwości fabrycznego systemu. Opiszę krótko kilka pozycji, z których większość można dzisiaj znaleźć na Aminecie.

SID - popularny menadżer plików, który ma wbudowaną obsługę dyskietek pecetowych, włącznie z konwersją plików tekstowych. Można powiedzieć, że spełnia podobną

funkcję jak wbudowany program commodity CrosDOS w nowszych wersjach Workbencha.

DMOUSE - akcelerator myszy, który zawiera także inne opcje, w tym wygaszacz ekranu, wygaszacz wskaźnika, funkcję automatycznej aktywacji okna, a także opcję programowania klawisze.

ARP - zastępuje starsze wersje poleceń z katalogu "C". Można też pobrać bibliotekę "arp.library", która usprawnia systemowe okna wyboru i została wydana przed erą popularności biblioteki ReqTools.

HDCLICK - narzędzie do uruchamiania innych programów.

Na starszym Workbenchu nie ma elementów pozwalających na szybki dostęp do ulubionych programów. Za pomocą narzędzia HDClick możemy stworzyć przyciski tekstowe, podobnie jak w programie Tool Manager, który jednak wymaga nowszej wersji systemu. Możliwe jest nawet utworzenie całego menu na osobnym ekranie.

Programy dostępne w serwisie Aminet.net:

misc/antiq/ARP_13.lha
util/libs/arp40_2.lha
util/cli/dmouse.lha
util/boot/HDClick.lha

Workbench 1.3: Ikony 8-kolorowe

Gdy kupiłem swoją pierwszą Amigę 500, bardzo podobał mi się wygląd systemu 1.3. Czułem się z nim jak w domu, bo z jednej strony przypominał mi to, co wcześniej oglądałem na swoim C64, a z drugiej - widziałem w czasopiśmie komputerowych. Ponadto, funkcje systemu wydawały mi się bardzo nowoczesne, a oprogramowanie skrywało wręcz tajemnicze możliwości. Jednak bez względu na to, jak lubię ten niebiesko-pomarańczowy wygląd, nie da się ukryć, że 4 kolory to trochę mało. W końcu nawet zwykła A500 pozwala na uzyskanie ponad 4 tys. kolorów.

Kolejna generacja Workbench, czyli wersje 2.0 i 2.1, mają zdecydowanie bardziej stonowany wygląd, ale nadal są to 4 podstawowe kolory. Dużo lepiej powiększyć paletę blatu do 8 ub 16 kolorów, co daje dużo więcej możliwości w konfiguracji systemu, jednocześnie nie zmniejsza znacząco szybkości odświeżania grafiki.

Trzeba też zdecydować się na określoną rozdzielczość. Ma to znaczenie, bo wiele ikon ma proporcje przystosowane do rozdzielczości typu Hires Interlaced, czy 640x512 punktów. Tymczasem na "gołej" Amidze stosujemy podstawowy tryb Hires (640x256 punktów) i wielu Amigowców nie chce tego zmieniać. Znam osoby, które posiadają rozbudowane komputery i nadal chcą zachować ten charakterystyczny styl z nieproporcjonalną rozdzielczością. Jasne, że ikony można przerobić, choćby za pomocą systemowego

programu IconEdit, ale jest pracochłonne. Warto jednak wiedzieć, że nawet przy użyciu bardzo prostych środków uzyskamy ciekawe rezultaty.

Na początek proponuję instalację pakietu MagicWB. Czy jest wersja działająca na Workbenchu 1.3? Oczywiście! Można ją pobrać z Aminetu, konkretnie chodzi o plik o nazwie "MagicWB1.3.lha" (katalog "pix/mwb"). Nie jest to pełny odpowiednik dużej wersji "Magicznego Workbench", lecz po uruchomieniu pliku głównego ustawia specyficzną paletę kolorów. Można wtedy od razu używać ikon typu MagicWB, bo ich zapis nie różni się od zwykłych. Co prawda Workbench 1.3 ma pewne ograniczenia, ale po uważnym przyjrzeniu szybko okaże się, że nawet prosty IconEdit ma kilka bardzo przydatnych opcji.

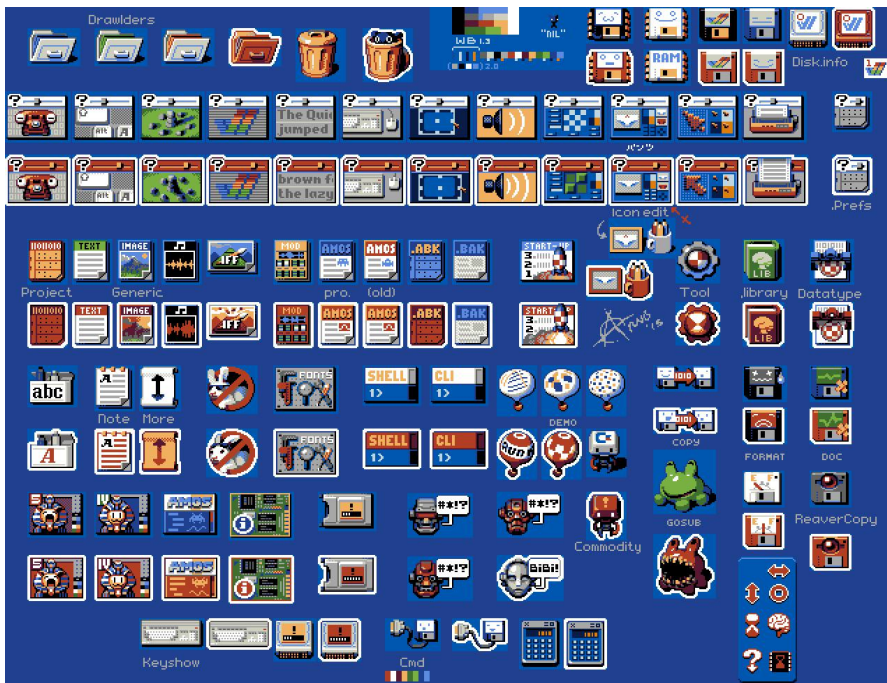
Najbardziej rzuca się w opcji opcja „Zmień kolory (OS1.3->OS2.0)”,

którą znajdziemy w kolejnym menu górnym o nazwie „Dodatki” (ang. „Extras”). Jest ona ważna, bowiem paleta kolorów każdej ikony jest ściśle określona. System Amigi w wersjach niższych niż 2.0 korzysta ze specyficznego zestawu kilku barw, który nie występuje już później. Z tego względu, jeśli wykorzystujemy ikony stworzone w różnych okresach czasu może zajść konieczność zmiany palety, aby grafika wyglądała właściwie zarówno na nowszym, jak i starszym Workbenchu. W tym oczywiście w systemie 1.3.

Pamiętajmy też, że ikony w systemie Amigi mają dwa stany - zaznaczony i niezaznaczony. Dlatego trzeba wziąć pod uwagę obie grafiki przypisane, czy też ściślej mówiąc - zapisane w pliku z ikoną (rozszerzenie ".info"). W edytorze ikon możemy ręcznie zapisać i załadować każdą grafikę, lecz jest to zadanie dość niewygodne.

Dlatego program wyposażony jest w dwie funkcje automatycznej zamiany grafiki stanów ikony. Pierwsza z nich to opcja „Zmień ze sobą” (ang. „Exchange”) umieszczona ciągle w menu górnym „Obrazki”. Zamienia ona obie grafiki w taki sposób, że wygląd ikony zaznaczonej stanie się niezaznaczoną i na odwrót.

Nie zawsze jednak chcemy zamienić obie grafiki jednocześnie. Dlatego można w zamian skorzystać z opcji znajdującej się niżej, opisanej jako „Skopiuj” (ang. „Copy”). Kopiuje ona grafikę ikony jednego stanu do drugiego w zależności od tego, jaki aktualnie jest wybrany. Innymi słowy ujednotolica wygląd ikony w obu stanach przyjmując jako punkt wyjścia grafikę, którą



Ikony na Workbenchu 1.3 mogą wyglądać zdecydowanie bardziej nowocześnie, nawet przy zachowaniu oryginalnej palety barw.

mamy w danym momencie wyświetloną w strefie edycji.

Program IconEdit ma także możliwość wstawienia standardowego wzoru, od którego można zacząć pracę. Jest to najprostsza grafika, której elementy zawiera większość domyślnych ikon na Workbenchu. Aby ją wstawić należy skorzystać z opcji „Wstaw podkład” (ang. „Use Template”). Wzór zostanie umieszczony jako grafika ikony zaznaczonej lub niezaznaczonej zależnie od stanu ustawionego podczas edycji.

Jak możemy sobie jeszcze ułatwić edycję ikon, szczególnie w kontekście Workbenchu 1.3? Można wczytać dowolny plik w formacie IFF, narysowany choćby w nieśmiertelnym edytorze Deluxe Paint. Służą do tego opcje zgrupowane w menu górnym o nazwie „Obrazki”. Trzeba tylko przejść do pozycji „Wczytaj”. Za

pomocą opcji „Normalny” (ang. „Normal Image”) możemy wczytać grafikę, która będzie użyta dla niezaznaczonego stanu ikony.

Jeśli skorzystamy z opcji „Zaznaczony” (ang. „Selected Image”), program wskazany plik wykorzysta jako stan zgodny z nazwą funkcji. Możliwe jest również załadowanie pliku IFF od razu jako oba stany ikony. W tym celu należy użyć funkcji o nazwie „Obydwa” (ang. „Both Images”). Zawartość strefy edycji można też zapisać jako plik graficzny w formacie IFF, aby wykorzystać go później w innym celu. Służą do tego opcja „Zapisz obrazek...” (ang. „Save Image”).

Paletę kolorów zbliżoną do MagicWB można też uzyskać samodzielnie, zmieniając cztery kolory podstawowych barw na Workbenchu. Trzeba pamiętać, że na układach OCS i ECS nie będziemy mieli tak dużej swobody

doboru odcieni, jak na Amidze 1200, ale można przyjąć, że właściwe ustawienia na suwakach RGB są następujące (od koloru 1 do 4):

09, 09, 09
11, 11, 11
11, 10, 09
15, 11, 10

Dzięki temu nie trzeba nic instalować - wystarczy uruchomić program Preferences z dyskietki Workbench 1.3. Na Aminecie można też znaleźć zewnętrzne narzędzie służące do tego celu, mianowicie program Image-Ed (katalog ("gfx/edit"). Jest to program typu Shareware, który pozwala na tworzenie i edycję wszystkich ośmiu kolorów na blacie. Teraz można rzeczywiście ustawić styl MagicWB, trzeba tylko zmienić paletę w odpowiedni sposób. Oto wartości, które należy wprowadzić w programie Image-Ed (w kolejności od kolor 1 do 8):

0A, 0A, 0A
00, 00, 00
0F, 0F, 0F
06, 08, 0B
09, 09, 09
0B, 0B, 0B
0B, 0A, 09
0F, 0B, 0A

Program operuje na liczbach szesnastkowych, stąd nietypowy zapis. Można to wykonać także w systemie 2.0, który w Polsce był najmniej popularny pod względem rozbudowy. Większość ludzi skupiła się na Workbenchu przynajmniej 3.0, zapominając, że na 1 MB czy 2 MB RAM trzeba liczyć każdy kilobajt wolnego obszaru, szczególnie jeśli mamy do dyspozycji tylko pamięć graficzną (typu Chip).

Polskie locale na dyskietce

Jak wiadomo, nie wszyscy Amigowcy lubią mieć system po polsku, ale jeśli już - zazwyczaj instalują pakiet WFMH Locale PL wyprodukowany dawniej przez Marcina Orłowskiego ze swoim zespołem tłumaczy. Gdy mamy Amigę 500 bez twardego dysku, ale z nowym Kickstartem 3.0 lub 3.1, możemy uzyskać ten sam efekt na dyskietce. Mówicie, że nie zmieszczą się wszystkie pliki? Nic z tych rzeczy. Oczywiście na dyskietce nie zmieszczą się kompletnie wszystkie pliki lokalizacyjne, ale możemy w prosty sposób ograniczyć je do minimum, dzięki czemu nawet tak ograniczony objętościowo nośnik będzie w stanie wywołać polską wersję językową systemu.

Na początek uwaga: wszystkie czynności musimy przeprowadzić ręcznie i warto podzielić je na trzy etapy – tworzenie katalogów, kopiowanie plików oraz ustawianie konfiguracji. Na dyskietce z Workbenchem trzeba utworzyć kilka katalogów, za pomocą poniższych linii w oknie Shell:

```
makedir DF0:Locale
makedir DF0:Locale/Catalogs
makedir DF0:Locale/Catalogs/polski
makedir DF0:Locale/Catalogs/polski/Sys
makedir DF0:Locale/Languages
makedir DF0:Locale/Countries
```

Przyjąłem, że dyskietka jest włożona na wbudowanej stacji, zamiast DF0 można też podać nazwę dyskietki - nie ma to znaczenia. Teraz na dyskietce trzeba zapisać pliki należące do plików językowych związanych z Workbenchem i pozostałymi składnikami systemu operacyjnego. Aby to osiągnąć musimy mieć rozpakowane pliki

pakietu WFMH Locale PL, a następnie przejść do katalogu "commodore/locale". Mamy tu dwa archiwa w formacie LZX, które trzeba rozpakować ręcznie.

Dzięki temu stworzone zostaną nowe katalogi oraz - co najważniejsze - pojawi się seria plików z rozszerzeniem ".catalog". Trzeba je skopiować do następującego katalogu:

Locale/Catalogs/polski/Sys

na naszej dyskietce. Zajmują tylko niecałe 100 kilobajtów. Kopiowanie uzyskamy za pomocą kolejnej linii w oknie Shell. Oto ona:

copy #?.catalogs DF0:Locale/Catalogs/polski/Sys/

W katalogu o nazwie „polski” zapisywane są pliki tłumaczeń dotyczących naszego języka. Katalog „Sys” zawiera pozycje dotyczące języka programów

wchodzących w skład systemu operacyjnego. Dzięki temu będzie można ustawić polską wersję całego Workbencha.

To nie wszystkie pliki, które muszą znaleźć się na dyskietce. Następne pozycje to pliki z katalogów „Languages” oraz „Countries” (zostały utworzone do rozpakowania archiwów LZX), które są niezbędne, aby system obsługiwał prawidłową dodatkową wersję językową. Posiadają one nazwę rozpoczynającą się od oznaczenia języka, w brzmieniu właściwym dla danego kraju.

W przypadku Polski będą to więc pliki „polski” oraz „polska”, a nie z angielska - „poland” czy „polish”. Dlatego teraz w oknie Shell, bez zmiany katalogu, wpisujemy kolejne dwie linie:

copy Languages/polski#? DF0:Locale/Languages/

copy Countries/polska#? DF0:Locale/Countries/

Skopiowane pliki z rozszerzeniami: „.language” i „.country” są charakterystyczne dla wymienionych katalogów. To już prawie wszystkie potrzebne pliki, jednak uruchomienie innej wersji językowej wymaga także obecności biblioteki systemowej o nazwie „locale”.

W tym celu trzeba ją zapisać w katalogu „Libs” na naszej nowej dyskietce. Biblioteka ta znajduje się na dyskietce systemowej z Workbenchem, dlatego trzeba przejść do katalogu o nazwie „Libs” i skopiować ją najpierw do pamięci, czyli „Ram Dysku”, w następujący sposób:


```

AmigaShell
Nowy proces Shell nr 3
3.Ram Disk:> copy LOCALE:Catalogs/polski/Sys/#? Ratunek:Locale/Catalogs/polski/Sys/ ALL
workbench.catalog..copied.
utilities.catalog..copied.
system.catalog..copied.
prepcard.catalog..copied.
prefs.catalog..copied.
monitors.catalog..copied.
libs.catalog..copied.
installer.catalog..copied.
dos.catalog..copied.
devs.catalog..copied.
commodities.catalog..copied.
c.catalog..copied.
amigaguide.catalog..copied.
3.Ram Disk:>
4.Ratunek:Locale/Catalogs/polski> mkdir DF0:Locale/Catalogs/polski/Sys
4.Ratunek:Locale/Catalogs/polski> dir
Sys (dir)
4.Ratunek:Locale/Catalogs/polski> dir
Sys (dir)
4.Ratunek:Locale/Catalogs/polski> sys
4.Ratunek:Locale/Catalogs/polski/Sys> dir
prefs.catalog      prepcard.catalog
system.catalog     utilities.catalog
workbench.catalog
4.Ratunek:Locale/Catalogs/polski/Sys>

```

Kopiujemy pliki lokalizacyjne na dyskietkę.

copy locale.library RAM:

a następnie włożyć naszą nową dyskietkę i wpisać kolejną linię:

copy RAM:locale.library DF0:Libs/

Biblioteka zajmuje niecałe 20 kilobajtów, więc nawet na 1 MB pamięci w "Ram Dysku" nie zabraknie miejsca. Gdy stacja przestanie pracować musisz jeszcze wykonać modyfikację sekwencji startowej. Jest to niezbędne, bowiem system będzie odczytywał dane z katalogu „Locale”, do którego musi prowadzić urządzenie logiczne o tej samej nazwie. Dlatego wywołujemy systemowy edytor tekstu:

ed DF0:s/startup-sequence

a następnie odnajdujemy fragment zawierający serię linii z poleceniem ASSIGN i uzupełniamy je o poniższy wpis:

assign LOCALE: SYS:Locale/

Oczywiście ilość poleceń ASSIGN oraz pozostała część pliku będzie zależała od stopnia, w którym rozbudowałeś system na dyskietce. Ważne jest tylko, aby kolejna linia znalazła się przed poleceniem IPREFS, a więc najlepiej zapisać ją w jednym miejscu – tam, gdzie tworzone są wszystkie urządzenia logiczne. Taki układ jest najbardziej czytelny i będzie działał zawsze, niezależnie od innych poleceń zapisanych w pliku sekwencji startowej.

Teraz możemy już zresetować Amigę i wczytać system ponownie za pośrednictwem dyskietki. Gdy zostanie uruchomiony Workbench, trzeba jeszcze wywołać program Locale, który normalnie znajduje się w systemowym katalogu o nazwie „Prefs”. Można tu skorzystać z oryginalnej dyskietki „Extras3.1”.

Zmiana konfiguracji w oknie Locale Preferences będzie wyglądała nieco inaczej niż zwykle, bowiem wyświetlone będą wyłącznie pozycje dla języka polskiego, w zamian wielu języków – jak ma to miejsce po uruchomieniu z twardego dysku. Trzeba jednak wykonać tylko kilka prostych czynności, które polegają na ustawieniu preferowanych języków w kolejności – najpierw wersja polska, a następnie – angielska. Teraz pozostaje już tylko zapisać konfigurację na dyskietce z pomocą przycisku Save. Aktualizacja konfiguracji Workbena będzie trwała nieco dłużej niż z twardego dysku - wszystko powinno trwać maksymalnie kilkanaście sekund. Ekran powinien zostać przełączony, ze standardowej angielskiej wersji językowej na polską. Na razie będzie to jednak dotyczyć tylko programów systemowych, bowiem skopiowaliśmy wyłącznie tłumaczenia zapisane w katalogu „Sys”.

Jeżeli chcemy uzupełnić pliki językowe o pozycje przeznaczone dla zewnętrznego oprogramowania, trzeba skorzystać z katalogu "dodatki/locale/catalogs". Zostały one utworzone po rozpakowaniu pakietu WFMH Locale PL, czyli w tym samym miejscu, co poprzednio. Znowu natrafimy na archiwa LZX. Po rozpakowaniu każdego z nich uzyskamy plik (lub pliki) z rozszerzeniem ".catalogs", które trzeba skopiować do katalogu "DF0:Locale/Catalogs/polski" na dyskietce.

Pamiętajmy, aby tym razem pozostawić katalog "Sys" bez zmian - tam są zapisywane tylko tłumaczenia systemowe. Podobnie,



wiele programów ma dołączone pliki z tłumaczeniami. Zawsze obowiązuje ta sama zasada - pliki ".catalog" kopiujemy do katalogu "Catalogs/polski". Należy też zwrócić uwagę na fakt, że pliki przeznaczone dla innych wersji językowych będą miały podobne nazwy, lecz muszą być zapisywane w katalogach odpowiadających im języków, na przykład „deutsch”. Z tego względu łatwo można popełnić błąd zapisując zły plik, co nie spowoduje wywołania innej wersji językowej, lecz program uruchomi się w domyślnym języku, zwykle – angielskim. Przy kopiowaniu trzeba zawsze sprawdzić, czy korzystamy z katalogu dla naszego języka, czyli „polski”.

```
Ed 2.00
MakeDir: RAM:T RAM:Clipboards RAM:ENV RAM:ENV/Sys
Copy >NIL: ENVARC: RAM:ENV ALL NOREQ

Assign ENV: RAM:ENV
Assign T: RAM:T
Assign CLIPS: RAM:Clipboards
Assign KEYMAPS: DEVS:Keymaps
Assign LOCALE: SYS:Locale

IPrefs
LoadWB
EndCLI

norge.country          polska.country
portugal.country      schweiz.country
suisse.country        sverige.country
swizzera.country      united_kingdom.country
united_states.country oesterreich.country
4.System:Locale/Countries >
```

Tak może wyglądać zawartość pliku startup-sequence dyskietki z polską wersją Workbencha.

Czym jest Workbench 1.4?

System Amigi w wersji 1.4 nigdy nie ukazał się oficjalnie. Został opracowany w 1989 roku, a później prace przeniesiono na wersję 2.0, która posiada wszystkie planowane elementy oraz wiele dodatkowych. Kiedyś była to tylko ciekawostka, natomiast dzisiaj możemy pobrać wydanie 1.4 Alpha lub Beta z sieci i cieszyć się nietypowym Workbenchem na pięćsetce. Jakie zmiany wprowadzono w rozwojowej wersji?

Ogólnie rzecz biorąc, system wygląda podobnie jak 1.3, ale nie uruchomi się bez Kickstartu 1.4. Wersja Alpha posiada paletę kolorów jak stara wersja, podobnie elementy typu czcionki, ikony czy wskaźnik myszki. Natomiast wersja

Beta bardziej przypomina Workbencha 2.0, choć widać, że nie jest ukończona, na przykład w miejscu przycisków na ramce okien widać tylko puste szare kwadraty.

Bardziej rozbudowane jest menu rozwijane i ma prawie takie same opcje jak finalna wersja 2.0. Przyciski na oknach działają inaczej, zmieniono też suwaki i strzałki obok nich. Jedną z największych zalet jest możliwość wyświetlenia całej zawartości katalogu w oknie Workbencha, a nie tylko ikon. Można to zrobić także w formie posortowanej według nazwy, daty lub rozmiaru.

Menu górne posiada opcję "ResetWB", tak samo jak w finalnej

wersji. Listwa tytułowa ekranu ma komunikaty ułożone inaczej - zajmują one mniejszą ilość miejsca, ale pokazują już oddzielnie pamięć Chip i Fast. Zmieniony też wskaźnik zapęłnienia dysku. W sieci można znaleźć wiele wersji systemu 1.4, natomiast najbardziej zbliżoną do finalnej są Alpha 15 i 18.

W dobie programowalnych układów pamięci, bez problemu stworzymy sobie "prototypową" Amigę z Kickstartem 1.4. Może się to przydać nie tylko do pokazów, ale także w coraz popularniejszych muzeach komputerów retro. Wyobraźcie sobie tylko rząd kilkunastu Amig 500, każda z inną wersją systemu operacyjnego.

Kickstart 1.3: Układanka w Amiga Basic

Z pewnością każdy zna grę typu puzzle, w których trzeba ułożyć wszystkie elementy w określonej kolejności. Jednak możemy przesuwac po jednym elemencie, ale tylko tam, gdzie jest wolne miejsce. W naszym wypadku będzie to 15 ponumerowanych pól ułożonych w formie 4x4 z jednym wolnym kwadratem. Zamiast cyfr można też zastosować litery lub inne rzeczy, które muszą być poukładane w określonej kolejności.

Program działa na każdej Amidze, na której można uruchomić Amiga Basic. Należy go wpisać i przed uruchomieniem zapisać na dysku. Następnie w ustawieniach Preferences (z dyskiety Workbench 1.3) włączyć tryb 60 znaków w kolumnie (zamiast 80 znaków). Po uruchomieniu program otworzy małe okienko na ekranie Workbench'a i losowo rozmieści elementy układanki. Aby przesunąć płytkę do wolnego miejsca, należy wskazać ją myszką i nacisnąć lewy przycisk myszy. Jeśli spróbujemy "oszukiwać" przesuwając kafel po przekątnej, program na to nie pozwoli. Przycisk w prawym dolnym rogu okna, zwykle zarezerwowany do zmiany rozmiaru, został zastąpiony znakiem plus. Jeśli go klikniemy, zostanie uruchomiona nowa gra. Pozostałe przyciski na ramce okna zachowują swoje funkcje.

Na koniec, w oknie programu wyświetlany jest czas, jaki upłynął od rozpoczęcia gry oraz wykonana liczba ruchów. W związku z tym można zrobić małe zawody w układaniu, biorąc pod uwagę nie tylko czas, ale też ilość przesunięć elementów. Możliwe jest także uruchomienie kilku programów jednocześnie i każda układanka

Poniższy program zawiera prostą grę logiczną, która otwiera małe okienko na ekranie Workbench'a. Jednak, co ważniejsze, demonstruje on kilka interesujących technik programowania w języku Amiga Basic. Dla niektórych będą to techniki zaawansowane, choć nie są trudne w zastosowaniu.

będzie wyświetlana w swoim oknie. Korzystamy tutaj z wielozadaniowości systemu Amigi, co jest możliwe nawet w starej wersji 1.3.

Prezentowany program został zaadaptowany z podobnej gry logicznej dostępnej na komputerach Apple. Osoby, które znają Microsoft Basic na Macintoshach, od razu zauważą wiele podobieństw do Amiga Basic. Jest to przecież również produkt Microsoftu. Warto zauważyć, że wiele programów z Macintosha można przepisać na Amigę przy niewielkim wysiłku. Oba języki obsługują okna, rozwijane menu, etykiety, wiele rodzajów poleceń graficznych i inne przydatnych funkcji.

Większość programu to standardowe polecenia Basica. Jedyne linie, które zasługują na szczególną uwagę, to zdefiniowane przez użytkownika funkcje w podprogramie o nazwie InIt. Jedną z funkcji obsługuje licznik czasu, a

druga określa czy układanka została ułożona poprawnie.

Szybkim sposobem na określenie czasu dowolnego zdarzenia jest zdefiniowanie funkcji, która odejmie bieżący czas od początkowego czasu, zaraz po uruchomieniu programu. Dokonujemy tego za pomocą funkcji FNlaps. Druga zdefiniowana funkcja to FNwin. Określa ona, czy układanka została ułożona. Jest to wykonywane poprzez porównanie liter w każdym elemencie zapisanym w tablicy. Każda litera we właściwej pozycji zwraca wartość -1 (True, czyli logiczna prawda). Tak więc puzzle jest ułożone poprawnie, gdy otrzymamy wartość -15.

Bardzo ważną funkcją Amiga Basic (i innych dialektów) jest możliwość tworzenia podprogramów. Pozwalają one na dodawanie nowych poleceń do języka i używanie ich tak samo jak zwykłych słów kluczowych. Słowo TALK nie jest poleceniem w języku Amiga Basic,



ale zostało dodane właśnie w formie podprogramu. Umożliwia uruchamiać zarówno polecenia SAY, jak i funkcji TRANSLATE\$ (przekształca tekst w syntezowaną mowę) poprzez proste wpisanie TALK. Podobnie słowo POSITION jest podprogramem, który oczekuje na parametry współrzędnych X oraz Y, aby przyjąć je jako miejsce wyświetlenia tekstu.

Program numer 2 pozwala na stworzenie specjalnej ikony do programu. Jest to oczywiście działanie opcjonalne, ponieważ Amiga Basic automatycznie przypisuje domyślną ikonę po zapisaniu programu. Dane niestandardowej ikony są konwertowane z wartości szesnastkowych w liniach ze słowami DATA na wartości jednobajtowe i zapisuje w pliku "Puzzle.info". Oprócz kształtu i koloru ikony, dane zawierają również informacje powodujące, że Basic jest od razu domyślnym narzędziem dla programu. Oznacza to, że po kliknięciu na ikonę na ekranie Workbench'a, Amiga Basic będzie automatycznie wczytywany przed załadowaniem i uruchomieniem gry.

Wspominałem wcześniej, że program korzysta z multitasking'u. Podkreślam to, bo wielu osobom wydaje się, że taka możliwość była nieznana przed wydaniem Workbench'a 2.0. Jest to standardowa funkcja systemu Amigi i program w Amiga Basic może z niej korzystać bez użycia specjalnych technik programowania. Trzeba tylko pamiętać, że liczy się wolna pamięć RAM, tak więc najlepiej, aby nasza Amiga 500 miała więcej niż 1 MB pamięci.

Program nr 1

```
Start:
    GOSUB Init
    GOSUB DrawScreen
    GOSUB Mix

    WHILE WINDOW(7) <> 0
    GOSUB Play
    WEND

Done: ,
    BEEP
    WINDOW CLOSE 2:WINDOW 1
    END

Init:
    DEFINT a-z
    Talk " "
    WINDOW 1, "Puzzle", (230,48) - (230+138,48+96), 30
    tries= 0:RANDOMIZE TIMER

    FOR y=0 TO 3#
    FOR x=0 TO 3#
    c(x,y)=x+y*4+ASC("A")
    NEXT x#
    NEXT y#

    blankX=x-1:blankY=y-1
    cCblankX,blankY)=ASC(" ")

    DEF FNlaps1=( (TIMER-starttime1)\60)+((TIMER-starttime1)
    Mod 60)/100)
    DEF FNa=(c(0,0)=65)+(c(1,0)=66)+(c(2,0)=67)+(c(3,0)=68)
    DEF FNb=(c(0,1)=69)+(c(1,1)=70)+(c(2,1)=71) +(c(3,1)= 72)
    DEF FNa=(c(0,2)=73)+(c(1,2)=74)+(c(2,2)=78)+(c(3,2)=76)
    DEF FNd=(c(0,3)=77)+(c(1,3)=78)+(c(2,3)=79)+(c(3,3)=80)
    DEF FNwin=(FNa+FNb+FNc+FNd)
    RETURN

DrawScreen:
    FOR y=0 TO 3#
    FOR x=0 TO 3 #
    Position (x+1)'3, (y+1)'2, CHR$(c(x,y))
    x1=x'30+10:y1=y

    LINE (x1,y1)-(x1+30,y1+18),1,b
    LINE (x1-1,y1-1)-(x1+30+1,y1+18+1),1,b
    LINE (10-6,3-3)-(4'30+10+8,4'18+3+3),1,b

    moreX=128:moreY=89
    LINE (moreX,moreY)-(moreX+10,moreY+10),1,bf
    Position 14,11,"+"
    rat(x,y,0)=x1:rat(x,y,1)=y1
    NEXT x
    NEXT y

    Position 3,10,"TIME 0.00"
    Position 3,11,"TRIES 0"

RETURN

Mix:
    x=blankX:y=blankY #

    FOR mixing=333 TO 1 STEP -1
    IF (mixing AND 1)=0 THEN x=INT(RND'4):y=blankY
    ELSE#
```

```

y=INT(RND'4):x=blankX
END IF#
GOSUB CheckCheat#
NEXT mixing #

```

```

Talk "Ready."
starttimel =TIMER
RETURN

```

Play:

```

LOCATE 10,8:PRINT USING "##.##";FN1

```

```

WHILE MOUSE(0)<>0
mouseX=MOUSE(3):mouseY=MOUSE(4)

```

```

FOR y=0 TO 3#
FOR x=0 TO 3 #
IF (mouseX>rat(x,y,0) AND mouseX<rat(x,y,0)+30) AND (mouseY>rat(x,y,1) AND
mouseY<rat(x,y,1)+18) THEN GOSUB CheckCheat:RETURN
NEXT x
NEXT y

```

```

GOSUB More
WEND
RETURN

```

CheckCheat:

```

IF (ABS(x-blankX)> 1 OR ABS(y-blankY)>1) OR ((x<>blankX AND y<>blankY)) THEN
IF mixing=0 THEN Talk "Cheater."
ELSE
SWAP c(x,y),c(blankX,blankY)
Position (x+1)*3,(y+1)*2,CHR$(c(x,y))
SWAP x,blankX:SWAP y,blankY
Position (x+1)*3,(y+1)*2,CHR$(c(x,y))
END IF

```

```

IF mixing=0 THEN
tries=tries+1
LOCATE 11,8:PRINT tries
WHILE MOUSE(0)<>0:WEND
IF FNwin=.15 THEN Talk "We have a winner.":GOTO More
END IF
RETURN

```

More:

```

WHILE MOUSE(0)<>0 OR FNwin=-15
mouseX=MOUSE(3):mouseY=MOUSE(4)

```

```

IF MOUSE(0)=0 AND (mouseX>moreX AND mouseX<moreX+ 10) AND (mouseY>moreY AND
mouseY<moreY+10) THEN GOTO Start
IF WINDOW(7)=0 THEN Done
WEND
RETURN

```

```

SUB Talk(a$) STATIC
SAY TRANSLATE$(a$)
END SUB

```

```

SUB Position(x,y,a$) STATIC
LOCATE y,x:PRINT a$;
END SUB

```



Program nr 2

```
WIDTH 60
PRINT:PRINT SPC(20);"Creating Puzzle Icon":PRINT
DEFINT a-z
CLOSE
OPEN "Puzzle.info" FOR OUTPUT AS 2

FOR ii =1 TO 98
READ long$
FOR i=1 TO 8 STEP 2
byte$=MID$(long$,i,2)
PRINT byte$;
HX byte$
tmp=VAL(byte$)
checksum=checksum+tmp
PRINT #2,CHR$(tmp);
NEXT i
NEXT ii
CLOSE
KILL "Puzzle.info.info"
IF checksum<>12063 THEN BEEP:PRINT:PRINT "Error in data statements"
END

SUB HX(a$) STATIC
a$=UCASE$(a$):n=0:ans=0

FOR i=1 TO LEN(a$)
c=ASC(MID$(a$,i,1))
n=VAL(CHR$(c))

IF c>ASC("9") THEN n=(VAL(CHR$(c-17)))+10
ans=ans*16+n
NEXT i

a$=STR$(ans)
END SUB

DATA E3100001,00000000,00800062,00280017
DATA 00040003,00010001,B8C00000,00000000
DATA 00000000,00000000,00000000,00000000
DATA 04400001,B8F00000,00000000,00700000
DATA 00560000,00000000,00000000,10000000
DATA 00000028,00170002,00021600,03000000
DATA 00000000,00000000,3FFFFFFF,F0002000
DATA 00000400,27FFFFFF,E4002404,08102400
DATA 24040810,24002404,08102400,27FFFFFF
DATA E4002418,7E0F2400,243C3319,A400243C
DATA 33302400,27663E30,E400247E,33302400
DATA 24C33319,A40024C3,7E0F2400,27FFFFFF
DATA E4002404,08102400,24040810,24002404
DATA 08102400,27FFFFFF,E4002000,00000400
DATA 3FFFFFFF,F0000000,00000000,00000000
DATA 00000000,00000000,00000000,00000000
DATA 00000000,00000000,00000000,00000000
DATA 00000000,00000000,00000000,00000000
DATA 00000000,00000000,00000000,00000000
DATA 00000000,00000000,00000000,00000000
DATA 00000000,00000000,00000000,00000000
DATA 00000000,00000000,00000000,00000000
DATA 00000000,00000000,00003A41,6D696761
DATA 42415349,43000000
```

Multi File System

MultiFileSystem nie jest systemem plików działającym w zwykły sposób: jest interfejsem do innych systemów plikowych. Przy montowaniu należy określić, z jakich file systemów chcemy korzystać. Następnie, gdy do stacji dysków włożona zostanie dyskietka, program rozpozna nośnik i przekaże polecenia do odpowiedniego systemu plikowego. Oznacza to, że montując tylko jedno urządzenie w systemie można uzyskać dostęp do różnych formatów nośników, na przykład MS-DOS, bez potrzeby aktywacji dodatkowego urządzenia (tutaj: "PC0:").

To wszystko jest możliwe dzięki wszechstronności systemu AmigaDOS. Widzimy to już na Workbenchu 2.0 lub wyższym, do którego dołączona jest ekonomiczna wersja pakietu CrossDOS pozwalającego na odczytywanie pecetowych dyskietek. Pełna wersja pozwala też na obsługę dysków twardych, więc MultiFileSystem może być przydatny nie tylko dla fanów dyskietek.

Jeśli jednak korzystamy z dwóch lub trzech różnych formatów, czyli wielu systemów plikowych, zwykle pojawiają się pewne niedogodności. W zależności od rodzaju włożonego dysku należy wywołać to samo urządzenie (na przykład stację dysków), ale za pomocą różnych nazw urządzeń ("DF0:" i "PC0:"). Po włożeniu dyskietki zapisanej w formacie MS-DOS, na Workbenchu zostanie wyświetlona ikona "DF0:???" i na odwrot - po włożeniu dyskietki amigowej zobaczymy ikonę "PC0:???".

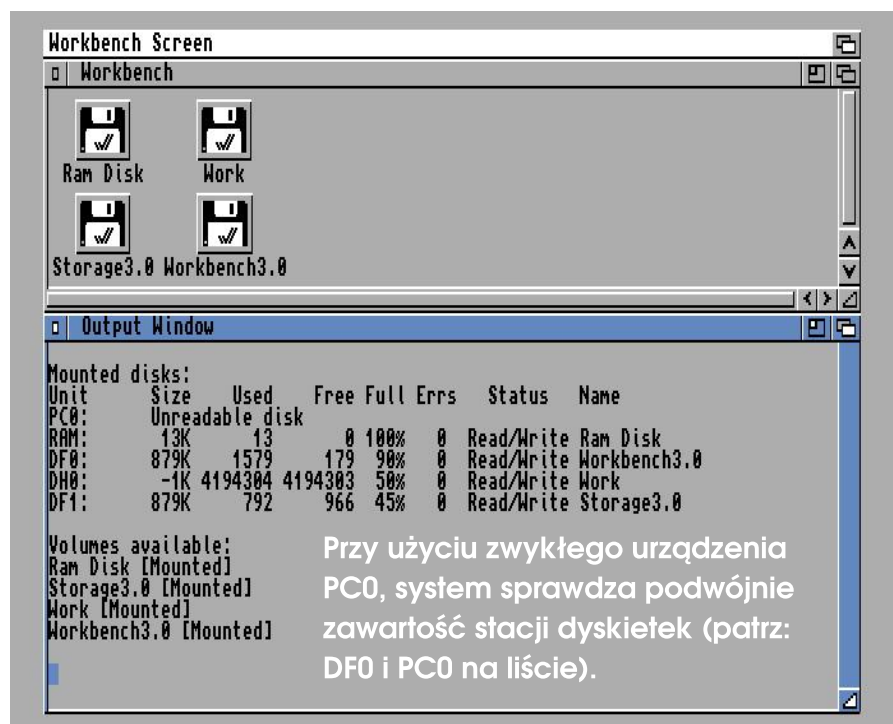
Rozpoznawanie takiej dyskietki będzie dłuższe, bo sprawdzana jest wtedy przynajmniej dwukrotnie.

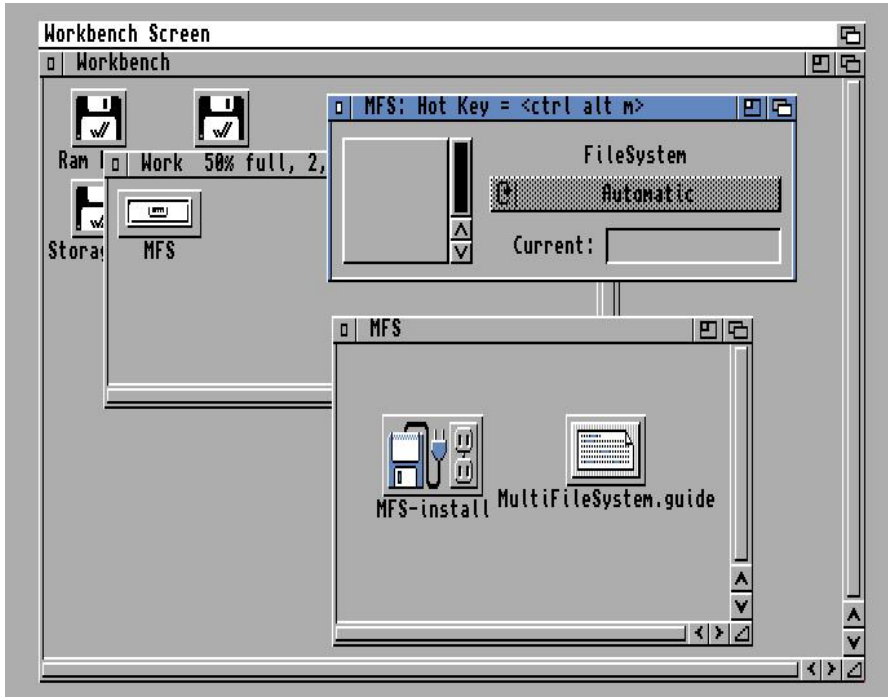
Ponadto, jeśli korzystamy z programów blokujących czasowo dostęp do systemu plikowego (na przykład podczas kopiowania),

pozostałe urządzenia nadal mają dostęp do nośnika, mimo że fizycznie obsługują to samo urządzenie. Jest to sprzeczne z filozofią systemu Amigi, poza tym może powodować problemy. MultiFileSystem rozwiązuje wszystkie te kłopoty.

Instalacja pakietu jest łatwa, bo autor dołączył instalator. Kopiowany jest także program "MFS" do katalogu "SYS:Tools/Commodities", jak również modyfikuje polecenia systemowe Format i DiskCopy, aby wykorzystywały funkcje MultiFileSystemu. Aby wszystkie zmiany zostały uruchomione należy zrestartować system, program instalacyjny nie robi tego automatycznie.

Skrypt instalacyjny tworzy w katalogu "DEVS:DOSDrivers" specjalny katalog dla każdej dostępnej stacji dysków. Dlatego, jeżeli w tym katalogu zapisane są ikony takie jak "PC0", to zostaną one przeniesione do





Program Commodity pozwala wybierać docelowy format nośnika. Gdy nie mamy aktywnego sterownika MFS, lista nie jest aktywna.

nowego katalogu. Jednak program może nie rozpoznać poprawnie wszystkich plików, dlatego niektóre ikony trzeba skopiować ręcznie.

Podczas uruchamiania, wspomniany katalog "DEVS:DOSDrivers" zostanie sprawdzony, czy znajdują się w nim katalogi, zamiast zwykłych sterowników używanych przez standardowe systemy plikowe. Dla każdego katalogu utworzone zostanie urządzenie systemowe o tej samej nazwie. MultiFileSystem będzie współdzielić formaty nośników zapisane w katalogu.

Programy systemowe nie mogą rozpoznać nowych sterowników, dlatego autor dotychczas do archiwum łątki zmieniające zawartość poleceń Format i DiskCopy tak. Dodajmy, że zmiany są wykonywane podczas instalacji,

dlatego nie musimy tego robić ręcznie. Później, po kliknięciu na ikonę dyskietki i wybraniu polecenia formatowania, program zapyta, który system plikowy ma być użyty.

Przy uruchomieniu z poziomu okna Shell, dostępny jest nowy argument:

FS=FILESYSTEM

Pozwala na określenie formatu, który jest wyświetlany w oknie MFS. Może to wyglądać na przykład tak:

Format DRIVE=DF0: NAME="EMPTY" FS=MS-DOS

W przypadku systemowego programu DiskCopy, przeciągnięcie ikony dyskietki spowoduje automatyczne wybranie

odpowiedniego systemu plikowego w urządzeniu docelowym.

Wracając do programu MFS, który jest programem typu Commodity, został napisany, aby zapewnić możliwość zablokowania określonego systemu plikowego przed sformatowaniem lub skopiowaniem dyskietki. Program może być szczególnie przydatny, gdy wymagana jest dokładna kontrola nad formatem nośnika. Wyświetla on również system plikowy używany przez aktualnie włożony nośnik.

Obsługa jest bardzo prosta - wystarczy wybrać urządzenie z listy i wybrać system plikowy, który ma być używany. Domyślnym ustawieniem jest "Automatic" i oznacza to, że MultiFileSystem automatycznie wybierze format w zależności od włożonej dyskietki. Pozostałe wpisy w przycisku cyklicznym to nazwy systemów plikowych.

MFS może być świetnym zamiennikiem dla duetu DF0-PC0, który osobiście używam bardzo często. Tak się dzieje, bo formatowanie dyskietek w formacie MS-DOS na Linuksie i zewnętrznej stacji dysków USB nie należy do najwygodniejszych czynności. Z takich dyskietek potrzebuję korzystać, ponieważ na biurku mam również retro peceta nieposiadającego CD-ROM-u. Myślę więc, że mój opis może przydać się wielu osobom przenoszącym dane pomiędzy różnymi komputerami. Pamiętajmy, że nie musi być to MS-DOS, MultiFileSystem jest elastyczny i pozwala na obsługę praktycznie dowolnego systemu plikowego.

Własny styl Workbencha

CZĘŚĆ 1: KOLORY I IKONY

Konfiguracja systemu Amigi nie należy do najłatwiejszych. Z drugiej strony uzyskujemy możliwości, o których właściciele innych platform nie mogli pomyśleć przez długi czas lub nawet teraz mają z tym problemy. Co innego zmiana wyglądu Workbencha - jest to dużo łatwiejsze. Niestety i pod tym względem napotkamy wiele przeszkód, bo oryginalny system zawiera tylko podstawowe narzędzia. Jeśli chcemy zmienić coś więcej niż kolory, czcionki czy podkłady, trzeba zainstalować dodatkowe oprogramowanie. W wielu wypadkach trzeba uważać, aby nie zablokować sobie możliwości uruchomienia całego systemu operacyjnego.

Konfiguracja systemu Amigi nie należy do najłatwiejszych. Z drugiej strony uzyskujemy możliwości, o których właściciele innych platform nie mogli pomyśleć przez długi czas lub nawet teraz mają z tym problemy. Co innego zmiana wyglądu Workbencha - jest to dużo łatwiejsze. Niestety i pod tym względem napotkamy wiele przeszkód, bo oryginalny system zawiera tylko podstawowe narzędzia. Jeśli chcemy zmienić coś więcej niż kolory, czcionki czy podkłady, trzeba zainstalować dodatkowe oprogramowanie. W wielu wypadkach trzeba uważać, aby nie zablokować sobie możliwości uruchomienia całego systemu operacyjnego.

Pierwsza ważna sprawa to ustalenie palety kolorów, co może się wydawać błahym problemem. Jednak system Amigi jest tak

skonstruowany, że dokonuje zmiany kolorystyki w sytuacji, gdy ekran nie jest uruchomiony w trybie 24-bitowym. Ścisłej mówiąc, jeśli wyświetlana grafika nie może być pokazana w oryginalnej ilości kolorów, paleta jest dostosowywana tak, aby obraz wyglądał jak najlepiej. Niestety, przy okazji zmieniane są kolory interfejsu użytkownika.

Z jednej strony to zaleta, bo nawet jeśli mamy na Workbenchu ustawiony tryb 8-, 16- czy 32-kolorowy, nadal możemy wyświetlać grafikę w niezłej jakości. Z drugiej strony, chyba nikt nie chce mieć ramek okien i przycisków pokazywanych w różnych, często nieprzewidywanych kolorach - i to po każdym załadowaniu systemu.

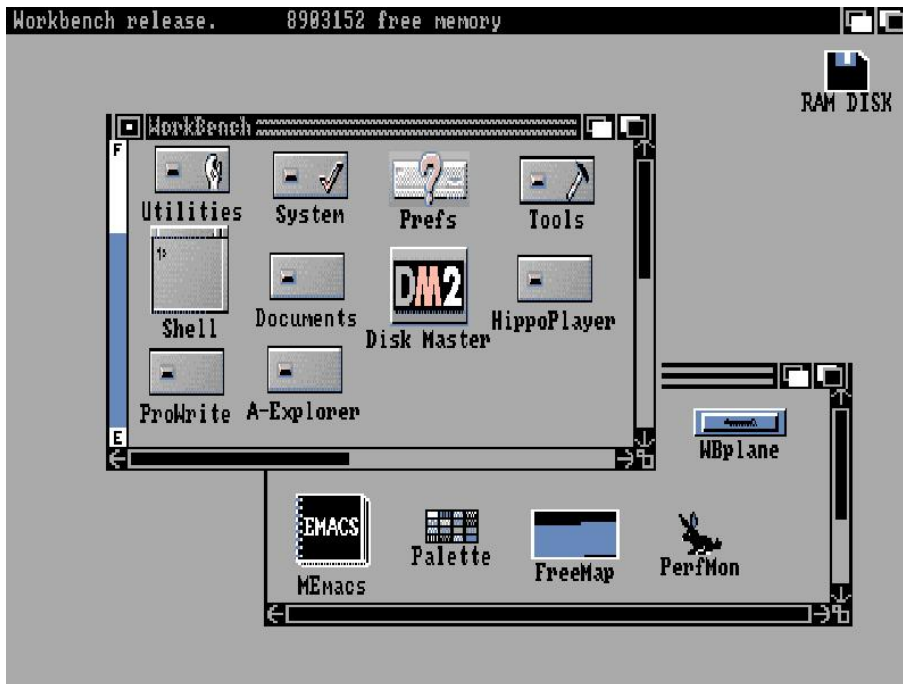
Z pomocą przychodzą programy, które można znaleźć w serwisie

Aminef.net. Zanim jednak tam wejdziemy, musimy ustalić ilość kolorów na naszym Workbenchu. Zależy to oczywiście od modelu i konfiguracji Amigi jaką posiadamy. A to już prawdziwy temat rzeka, dlatego postaram się go przybliżyć w jak największym skrócie.

Jeśli posiadamy Amigę z chipsetem OCS lub ECS, czyli A500 czy A600, wtedy najlepszym rozwiązaniem jest korzystanie z Workbencha w 4 lub 8 kolorach. Pozwala to uzyskać dużą szybkość odświeżania grafiki i oszczędzać pamięć. Dodatkowo, można pokusić się o ustawienie trybów zgodnych z monitorami VGA (np. Multiscan), ale na układach ECS jest to ograniczone właśnie do 4 kolorów jednocześnie.

Nasz pulpit nie będzie zbyt kolorowy, ale niebiesko-szary styl może wyglądać elegancko. Zachowamy więc oryginalną kolorystykę systemu. Na Aminecie można znaleźć 4-kolorowe ikony, jak również podkłady na pulpit, nie jesteśmy więc ograniczeni do dyskietek z Workbenchem. Trzeba jednak wykazać się inwencją twórczą, aby uzyskać ładny wygląd.

W dużo lepszej sytuacji będą ci użytkownicy, którzy zdecydują się na włączenie 8- lub 16-kolorowego trybu graficznego. Najbardziej polecam ten pierwszy, bowiem daje to możliwość bezproblemowej instalacji znanego pakietu MagicWB. Ustala on specyficzną paletę kolorów, która jest dość stonowana, ale Aminef jest wręcz zalany dodatkami do "magicznego Workbencha". Każdy znajdzie tu coś dla siebie. Dodatkową zaletą jest fakt, że w tym wypadku paleta kolorów jest zablokowana. Dzięki temu nie jesteśmy narażeni na



Charakterystyczny styl MagicWB może być bez problemu obsługiwany na Workbenchu 1.3. Nie trzeba do tego mieć dużej ilości pamięci, wystarczy 1-2 MB RAM.

zmianę kolorów, o czym mówiłem wcześniej.

Z tego, co obserwuję na forach internetowych, wiele osób ma wciąż kłopoty z instalacją rozszerzeń do Workbenchu, dlatego jeszcze raz powiedzmy więcej na temat działania MagicWB oraz jego instalacji. Technicznie rzecz biorąc, pakiet zawiera definicję palety kolorów oraz ikony, które razem składają się na charakterystyczny wygląd Workbenchu.

Ikony tego rodzaju są nazywane, od nazwy pakietu, ikonami typu MagicWB. Nie różnią się formatem od zwykłych ikon systemowych, lecz zostały stworzone przy użyciu charakterystycznej palety 8 podstawowych kolorów. Stąd ich nietypowy wygląd. Zachowują wszystkie cechy zwykłych ikon Amigi. Program instalacyjny MagicWB aktualizuje automatycznie

ustawienia systemowe, więc użytkownik nie musi ręcznie dokonywać konfiguracji. Instalacja podzielona jest na dwa podstawowe etapy. W pierwszym należy rozpakować wszystkie niezbędne pliki, w drugim – zainstalować je w odpowiedni sposób na dysku. Najłatwiej skorzystać z dyskietki instalacyjnej MagicWB, gdyż program instalacyjny samodzielnie wykona wszystkie operacje. Obraz dyskietki można łatwo znaleźć w sieci.

Alternatywnie, można pobrać archiwum z Aminetu. Przechodzimy do katalogu „util/wb” i pobieramy plik o nazwie „MagicWB21p.lha”. Jest to archiwum LHA, dlatego trzeba je rozpakować, na przykład za pomocą standardowego okna „Wykonaj polecenie”. Trzeba w nim wpisać następującą linię:

lha x MagicWB21p.lha RAM:

Wszystkie pliki zostaną wtedy zapisane w „Ram Dysku”. Jeśli posiadamy dyskietkę instalacyjną, ten etap można pominąć. Po rozpakowaniu plików w katalogu znajdziemy ikonę o nazwie „Unpack MagicWB”. Ma charakterystyczną kolorystykę, ale na razie jest wyświetlana nieprawidłowo. Wszystko zmieni się po zakończeniu instalacji. Ważne jest, aby zapamiętać, że pakiet korzysta z pierwszych 8 kolorów palety barw, ustawiając je w określony sposób. Wpływa to na wygląd wszystkich elementów graficznych systemu, przez co daje spójny styl nazywany po prostu MagicWB lub w skrócie - MWB.

Po wykonaniu dwukliku na ikonie „UnpackMagicWB”, na ekranie pojawi się okno instalacyjne, lecz bardzo nietypowe, bo będzie przypominało zwykłe okno AmigaDOS. Obok podstawowych informacji o pakiecie widoczne jest pytanie: "Would you like to unpack MagicWB?". Należy wpisać literę Y i nacisnąć ENTER. Teraz w oknie pojawi się nowy komunikat z pytaniem o miejsce, gdzie mają zostać rozpakowane pliki. Jak zawsze można skorzystać z „Ram Dysku”, o ile oczywiście posiadamy wolną pamięć.

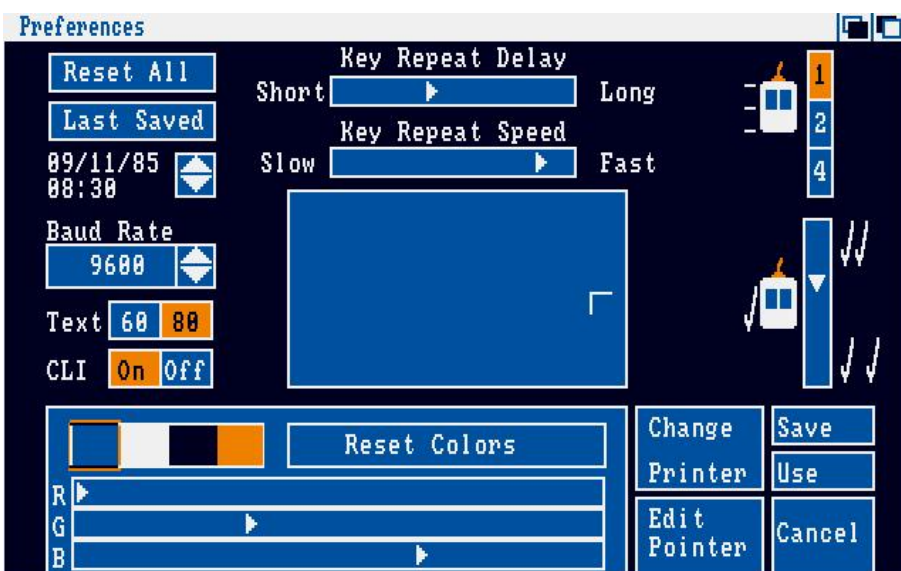
Jeśli nasza Amiga ma przynajmniej 2 MB wolnej pamięci, bez problemu korzystamy z tej opcji, czyli wpisujemy: "RAM:" (oczywiście bez cudzysłowu) i znowu naciskamy ENTER. Zamiast tego można rozpakować dane na jedną z partycji dysku twardego. W tym celu wpisujemy jej nazwę i kończymy znakiem dwukropka, czyli na przykład "Work:".

Program rozpocznie dekompresję plików, co w zależności od szybkości komputera może potrwać od kilkunastu sekund do nawet kilku minut. Trzeba poczekać na zakończenie operacji, co program zasygnalizuje komunikatem: "MagicWB has been successfully unpacked". Po naciśnięciu klawisza ENTER okno zniknie. Pierwszy etap instalacji mamy już za sobą.

Teraz musimy zainstalować niezbędne pliki w systemie. W tym celu wykonujemy dwuklik na ikonie „MagicWB” - cały czas w miejscu, gdzie rozpakowane zostały pliki. W

Naciskamy ENTER, co spowoduje, że w oknie pojawi się nowy tekst. Tym razem mówi on o tym, że pakiet będzie działał prawidłowo tylko wtedy, gdy w systemie ustawimy tryb wyświetlania posiadający przynajmniej 8 kolorów.

Jednocześnie program informuje, że 4 pierwsze kolory palety nie zostaną zmienione. Dotyczy to barw: szarej, czarnej, białej i niebieskiej. Dzięki takiemu rozwiązaniu nowy wygląd Workbencha nie koliduje z wcześniejszymi zmianami oraz starszymi ikonami. Tutaj również należy nacisnąć ENTER.



Preferencje systemu 1.3 wyglądają zupełnie inaczej niż w każdej kolejnej wersji Workbencha. W starym oknie można jednak bardzo łatwo zmieniać kolory na ekranie.

oknie pojawi się kilkanaście ikon. Znowu klikamy dwukrotnie, ale tym razem na ikonie „Install”. Pojawi się nowe okno, w nim informacja mówiąca o tym, że MagicWB zostanie zainstalowany na dysku systemowym. Użyte zostanie urządzenie „SYS:”, co gwarantuje brak problemów, nawet jeśli mamy podłączoną większą ilość dysków.

Kolejna informacja to porada sugerująca, aby ustawić tryb wyświetlania typu Hires Interlaced, czyli rozdzielczość 640x512 punktów. Pakiet jest przystosowany do rozdzielczości zapewniających tzw. kwadratowy piksel, czyli sytuacji, gdy pojedynczy punkt na ekranie posiada tę samą szerokość co wysokość. Można oczywiście

zastosować dowolny tryb, jednak musimy zwrócić uwagę, że w niektórych elementy graficzne będą rozciągnięte w pionie lub poziomie.

Choć MagicWB w zamierzeniach autora nie powinien być używany w takich trybach, nie ma żadnych formalnych ograniczeń i wszystko zależy od preferencji użytkownika. Poniżej widać pytanie: "Start the installation?". Naciskamy klawisz Y, a potem znowu ENTER. Program zacznie kopiować pliki potrzebne do pracy. Teraz należy poczekać aż instalator zakończy pracę. Powinno to potrwać kilkanaście do kilkudziesięciu sekund.

Następnym pytaniem jakie zostanie wyświetlone jest: "Do you have the AA (AGA) Chipset?". Musimy odpowiedzieć, czy nasza Amiga wyposażona jest w układy AGA. Należy odpowiedzieć zgodnie ze stanem faktycznym. Jeżeli wskażemy, że posiadamy AGA, program pokaże dodatkowy komunikat. Mówi on o instalacji plików, które można wykorzystać jako tło pulpitu lub okien. Ponadto zainstalowany zostanie program o nazwie „Magic Copper”, dzięki któremu tło może zawierać płynne przejście tonalne między dwoma barwami, czyli popularnie mówiąc - gradient. Wymaga to skorzystania z trybu wyświetlania przynajmniej 16-kolorowego.

Niżej instalator zada kolejne pytanie, będzie to tekst o treści: "Do you want to have this Special AGA-Background installed?". Program może zainstalować pliki zawierające obrazy tła Workbencha. Nie zajmują one wiele miejsca na dysku, dlatego odpowiadamy twierdząco. Po raz kolejny widoczne będą komunikaty



o kopiowaniu plików na dysk. Po naciśnięciu klawisza ENTER, uruchomiony zostanie program ScreenMode, za pomocą którego możemy wybrać tryb wyświetlania, jaki będzie używany na Workbenchu.

Jak już mówiłem wcześniej, najlepiej pozostawić 8 kolorów. Ustawienia te nie są oczywiście przypisane na stałe do systemu i możesz zmienić je później. Należy pamiętać, że większa ilość kolorów na ekranie oznacza jednocześnie zwiększone zużycie pamięci. Po ustawieniu ilości kolorów wybierz przycisk „Zapisz”.

Teraz na ekranie pojawi się poprzednie okno instalacji wraz z kilkoma następującymi po sobie komunikatami „Installing...”. Ostatni z nich dotyczy skopiowania ikony dysku systemowego. Powoduje to wyświetlenie kolejnej informacji: Do you want to have the HardDisk-Icon installed to other partition? Można zainstalować nową ikonę na innych podłączonych dyskach. Najprościej jednak odpowiedzieć N i nacisnąć ENTER.

W dalszej kolejności zobaczymy pytanie o to, czy mamy zainstalowany napęd SYQUEST. Jest to rodzaj dysku twardego o pojemności od 44 do 200 MB charakteryzujący się wymiennym nośnikiem, podobnie jak stacja dyskietek. Dziś jest praktycznie nie do zdobycia. Ponownie, odpowiadamy N i naciskamy klawisz ENTER.

Dalej, program wyświetli komunikat o możliwości zainstalowania dodatkowych małych grafik, które mogą znaleźć zastosowanie w programie Tool Manager. Służy on do tworzenia pasków ikon, dzięki



Tego typu ikony znajdziemy w serwisie Aminet.net Mogą mieć bardziej nietypowy wygląd, ale będą pasować do specyficznej palety ośmiu podstawowych kolorów.

którym można w łatwy sposób uruchamiać często używane programy. Pliki grafik nie zajmują dużo miejsca na dysku, a mogą się później przydać. Dlatego na pytanie: Would you like to have them installed? - odpowiadamy twierdząco.

Po chwili instalator wyświetli tekst świadczący o prawidłowo zakończonej instalacji w postaci: The installation is now successfully finished. Teraz trzeba kilka razy nacisnąć ENTER. Na koniec pojawi się okno „About MagicWB”. Zawiera podstawowe informacje o autorze pakietu. Wybieramy przycisk „Cancel”. Okno zniknie i powrócimy do Workbenchu. Instalacja została zakończona i aby zobaczyć zmiany powinniśmy zresetować Amigę.

Tak wygląda proces instalacji pakietu MagicWB. Po zresetowaniu komputera na Workbenchu będziemy mieli ustawioną prawidłową paletę kolorów tak, aby

wszystkie elementy były wyświetlane prawidłowo - ikony i grafika pasująca do 8-kolorowego stylu MagicWB. Proponuję teraz pobrać z Aminetu pliki pasujące do tej kolorystyki i zmienić ikony oraz podkłady na pulpicie według własnego uznania.

Na początek polecam sprawdzić zawartość plików takich jak: "MagicBGs5.lha", czy "MagicD3.lha". Pierwszy zawiera podkłady, drugi - ikony, które można bardzo łatwo używać nawet na małej rozbudowanej Amidze.

W kolejnym odcinku pokażę jak stworzyć własną paletę barw oraz zmienić wygląd systemowych okien tak, abyśmy nie mieli problemów z nieprawidłowo wyświetlanymi elementami interfejsu użytkownika. Przy okazji okaże się, że dostosowywanie palety barw na Workbenchu wcale nie działa tak źle, jak to twierdzą niektórzy internetowi forumowicze.

Installer: aktywność użytkownika

Pierwszą możliwością jest prośba o wybranie katalogu, którą widzimy w prawie każdym programie instalacyjnym. Aby ją wywołać używamy polecenia ASKDIR, na przykład:

```
(askdir
(prompt "Wybierz katalog
docelowy")
(help @ask-help)
(default "SYS:C/")
)
```

Funkcja ta pozwala użytkownikowi utworzyć nowy katalog lub wskazać już istniejący. W linii zawierającej argument „default” wskazujemy domyślną ścieżkę dostępu. Wynikiem działania jest ciąg tekstowy zawierający ścieżkę wybraną przez użytkownika. Dokładnie tak samo możesz wywołać prośbę o wybór pliku zapisanego na dysku. Należy tylko skorzystać z polecenia ASKFILE. W pozycji „default” cały czas podajemy jedynie ścieżkę dostępu, a nie nazwę pliku. Poza tym zapis całego fragmentu pozostaje niezmienny. W rezultacie otrzymasz nazwę pliku uzupełnioną o ścieżkę – co ważne – w jednym ciągu tekstowym. Trochę inaczej uruchamiamy prośbę o włożenie nośnika do dowolnej stacji dysków. Tym razem określamy nazwę, na jaką będzie czekał „Installer”:

```
(askdisk
(prompt "Włóż drugą dyskietkę
instalacyjną")
(help @ask-help)
(dest "Fonts_PL")
(newname "DYSK2")
)
```

W tym przykładzie program oczekuje na dyskietkę o nazwie podanej w linii ze słowem „dest”,

Systemowy program Installer posiada przynajmniej kilka zamienników, które rozszerzają jego możliwości. Jednak nawet stara wersja ma wiele przydatnych funkcji. Za pomocą jednej z nich możemy poprosić użytkownika o wpisanie określonej wartości albo wybranie miejsca na dysku odpowiedniego do skopiowania plików. Jest to oczywista funkcja w każdym instalatorze programu czy gry, ale trzeba wiedzieć jak jej użyć. Wbrew pozorom jest to bardzo łatwe, wystarczy zastosować kilka poleceń o nazwach rozpoczynających się od angielskiego słowa ASK. Większość z nich obsługujemy w bardzo podobny sposób.

czyli u nas - „Fonts_PL”. Zwróć uwagę, że symbolu nie należy kończyć znakiem dwukropka, tak jak zwykle. Argument „newname” nie jest obowiązkowy, lecz umożliwia ustalenie nowej nazwy nośnika. Będzie można postąpić się nią zamiast oryginalnej, ale tylko w ramach skryptu instalacyjnego. W powyższym przykładzie dyskietka „Fonts_PL” będzie traktowana tak, jakby posiadała nazwę „DYSK2”. W tym przypadku również nie podajemy dwukropka na końcu.

Opcja ta może mieć szerokie zastosowanie, na przykład gdy instalowany program zajmuje wiele dyskietek o różnych nazwach. Wtedy dużo łatwiej używać jednolitych symboli. W skrypcie możemy również zawrzeć prośbę o wprowadzenie liczby lub tekstu.

Służą do tego polecenia ASKNUMBER oraz ASKSTRING. Oba należy wywoływać w ten sam sposób, według następującej formuły:

```
(askstring
(prompt "Podaj nazwę sterownika")
(help @ask-help)
(default "scsi.device")
)
```

lub

```
(asknumber
(prompt "Podaj numer urządzenia")
(help @ask-help)
(range 0 3)
(default 0)
)
```

W pierwszym przypadku nie ma żadnych nowych opcji, wszystkie



Taki błąd zwykle znaczy, że mamy zbyt starą wersję Installera na dysku. Tutaj: wymagana jest 43.3 lub nowsza.

Bardzo podobnie możemy wstawić możliwość wyboru jednej z opcji za pomocą wskazania przycisku widocznego w dolnej części okna programu instalacyjnego. Jednocześnie możliwe jest wyświetlenie dwóch przycisków. Robimy to za pomocą polecenia o nazwie ASKBOOL w poniższy sposób:

```
(askbool
(prompt "Czy skopiować plik?")
(help "Opis sprawdź w instrukcji.")
(choices "Tak" "Nie")
(default 0)
)
```

Większość argumentów ma takie samo znaczenie jak w poprzednio. Teraz jednak, zamiast listy, napisy będą wyświetlane na przyciskach. Gdy użytkownik wskaże opcję widoczną po lewej stronie, wynikiem działania będzie wartość 1. W sytuacji, gdy wybrany zostanie drugi przycisk – wartość 0. Słowo „default” określa, który wybór stanie się domyślnym. Gdy aktywny jest tryb dla użytkownika początkującego, pytanie jest pomijane w całości, a w rezultacie otrzymujemy wartość 0.

Powyższe funkcje pozwalają tworzyć skomplikowane instalatory, jak również pogramy konfiguracyjne opcje innych narzędzi. Przykładem może być tutaj "Installer" do programu Directory Opus 5, za pomocą którego możemy między innymi zdecydować, czy domyślnym błatem ma być Workbench czy Opus. W tym przypadku pomijane są opcje instalacyjne programu, a wywoływana jest tylko jedna funkcja modyfikująca systemową sekwencję startową. Podobny schemat można wykorzystać bardzo szeroko do prawie każdego celu.

zachowują funkcje analogiczne do poprzednich. Linia z argumentem „range” definiuje natomiast przedział, w którym musi zawierać się wartość podana przez użytkownika.

Rezultatem działania tych funkcji będzie liczba lub ciąg tekstowy. Gdy uruchomiony będzie tryb przeznaczony dla użytkownika „początkującego” - wynikiem będzie 0 lub pusty ciąg.

Jeżeli chcemy utworzyć listę, z których użytkownik będzie mógł wybrać określoną jedną opcję, należy zastosować polecenie ASKCHOICE. Musi być zapisane wraz z określoną strukturą, która definiuje podstawowe parametry. Oto przykład:

```
(askchoice
(prompt "Wybierz język")
(help @askoptions-help)
(choices "english" "deutsch"
"polski")
(default 3)
)
```

Po słowie „prompt” należy wprowadzić pytanie, które zadajemy użytkownikowi, natomiast po „choices” wpisz listę opcji. Każdą z nich trzeba ująć w cudzysłowy i rozdzielać znakiem SPACJI. Argument „default” nie jest wymagany, lecz jeśli zostanie użyty określa domyślnie wybraną pozycję.

W naszym przykładzie mamy jeszcze linię zawierającą słowo „help” oraz zmienną rozpoczynającą się od znaku tzw. „małpy”, czyli „@”. Dzięki niej program zawierać będzie podręczną pomoc, której treść zostanie wstawiona z puli z góry określonych zmiennych. W rezultacie działania polecenia ASKCHOICE uzyskamy wartość liczbową liczoną od zera, określającą jaka opcja została wybrana.

Nie ma ograniczeń w długości listy, lecz trzeba wziąć pod uwagę typową rozdzielczość ekranu, aby cały komunikat był widoczny. W praktyce najlepiej stosować maksymalnie kilkanaście opcji, a jeśli chcemy ich użyć więcej – rozdzielać nazwy na kilka osobnych list. Jeżeli aktywny jest tryb użytkownika początkującego, program przyjmuje domyślnie jako wynik wartość 0.

Bliźniaczą funkcją jest ASKOPTIONS. Za jej pomocą użytkownik może jednak wybrać nie jedną, a wiele pozycji jednocześnie. Wynikiem działania będzie liczba, która w zapisie dwójkowym będzie posiadała bity ustawione według wskazanych pozycji na liście. I znowu - w trybie użytkownika początkującego funkcja ta przyjmuje wszystkie dostępne opcje za aktywne. Pozostałe uwagi są analogiczne do ASKCHOICE.

Tworzymy Amiga Guide

Amigowe oprogramowanie ma tradycyjnie dołączoną dokumentację w formie tekstowej lub jako dokument w formacie AmigaGuide (zwykle posiadają rozszerzenie „.guide”). Można go napisać w zwykłym edytorze tekstu, a jednocześnie przeglądanie zawartości jest dużo szybsze i łatwiejsze niż typowego pliku "Przeczytaj_Mnie". AmigaGuide to zwykły plik tekstowy, w ramach którego umieszczamy odpowiednie słowa kluczowe. Są to specyficzne ciągi znaków, za pomocą których tworzymy fragmenty wywołujące różne funkcje. Jak się zabrać do pracy?

Aby utworzyć plik AmigaGuide najprościej posłużyć się systemowym edytorem tekstu Ed. Nie jest on najwygodniejszy, ale uruchamia się na praktycznie każdej Amidze. Bardzo dobrym zamiennikiem będzie także Cygnus Editor lub Blacks Editor dostępny na Aminecie (katalog "text/edit"). W zasadzie rozszerzenie „.guide” nie jest konieczne, ale jest to najwygodniejszy sposób na oznaczenie naszego nowego dokumentu.

Plik w formacie AmigaGuide musi zaczynać się od następującej linii:

@DATABASE tytuł

gdzie „tytuł” to dowolna nazwa jaką chcemy nadać dokumentowi. Nie musi ona pokrywać się z nazwą pliku, jeśli natomiast zawiera znaki SPACJI powinna zostać ujęta w cudzysłów. Możemy więc wpisać linię:

@DATABASE Spis1

lub
@DATABASE "Spis 1"

W dalszej kolejności tworzymy rozdziały, które muszą rozpoczynać się od linii:

@NODE rozdział

a kończyć:

@ENDNODE

W miejscu słowa „rozdział” podajemy wewnętrzny tytuł rozdziału, co jest potrzebne do późniejszego tworzenia odnośników. Słowo ENDNODE służy do oznaczenia końca rozdziału, a pomiędzy powyższymi liniami umieszczamy właściwą treść rozdziału. Nazwa nie ma związku z tekstem, który wyświetlany będzie podczas przeglądania dokumentu.

Należy jednak zwrócić uwagę na to, że każdy dokument musi zawierać przynajmniej jeden rozdział i jeden z nim musi mieć nazwę „main”.

Dlatego na początek najlepiej stworzyć rozdział przy użyciu następujących linii:

@NODE main

oraz

@ENDNODE

Dopiero później uzupełnimy pozostałą część pliku. Rozdział „main” jest pierwszym, który zostanie pokazany po wczytaniu pliku do przeglądarki. Standardowo powinien zawierać spis treści, lecz można w nim umieścić inny tekst. Pozostałe rozdziały mogą posiadać dowolne nazwy, również ich kolejność w pliku nie ma znaczenia.

Początkowe linie, począwszy od słowa DATABASE, stanowią nagłówek, który możemy rozbudować o dodatkowe informacje. Muszą one znaleźć się przed treścią rozdziałów, czyli przed liniami rozpoczynającymi się od NODE. Można podać notkę o autorze, prawach autorskich oraz wersji dokumentu wraz z przypisaną do niej datą. W tym celu należy utworzyć kolejne linie zaczynające się od słów kluczowych:

AUTHOR

(c)

\$VER

Każde z nich musi poprzedzać znak „małpy”, czyli „@”, tak jak wcześniej. Całość może mieć poniższą formę:

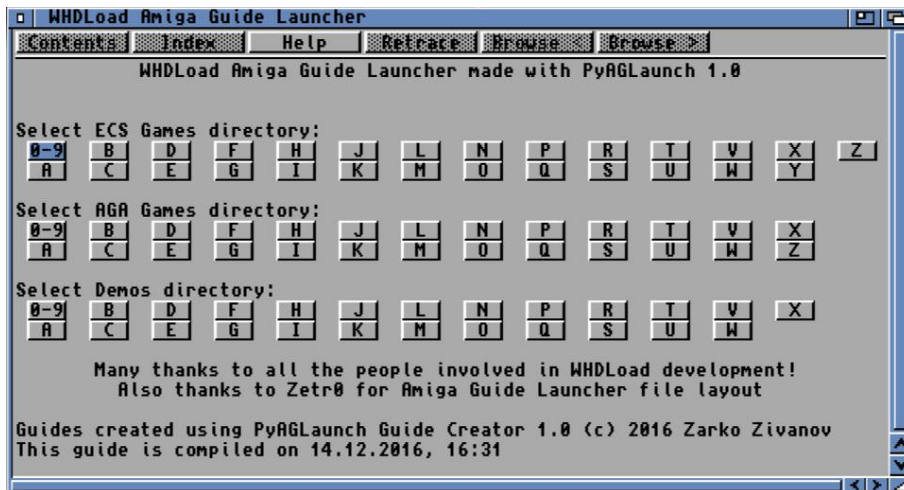
@DATABASE "Mój spis"

@AUTHOR "Adam Piotr"

@(c) 2020

@\$VER 1.0 11-Sty-20

Informacje te można podać również w jednej linii, jednak zachować należy odstęp przed każdym



Dokument w formacie Amiga Guide może zawierać tekst, lecz również przyciski. Za ich pomocą możemy przenosić się do innych części dokumentu lub innych plików na dysku.

znakiem „@”. W naszym przykładzie będzie to wyglądało tak:

@DATABASE "Spis 1"
@AUTHOR "Adam Zalepa" ©(c) 2014
@\$VER 1.0 11-Lip-14

W nagłówku możesz także umieścić nazwę i rozmiar czcionki jaki ma być użyty podczas przeglądania dokumentu. W tym celu korzystamy ze słowa FONT, umieszczając je analogicznie do poprzednich - w kolejnej linii przed treścią rozdziałów:

@FONT opal.font 11

Musi być to krój aktualnie dostępny na dysku, w przeciwnym razie wczytana zostanie domyślna czcionka ustawiona jako systemowa w programie preferencyjnym „Font”. Określanie kroju nie jest konieczne, ale dzięki temu możemy mieć większy wpływ na wygląd dokumentu.

Tekst może być sformatowany w taki sposób jak został wprowadzony przez autora albo tak, by linie miały

zawsze równą szerokość - zgodną z aktualnym rozmiarem okna.

W takiej sytuacji treść będzie wyglądała inaczej po każdym zmianie wielkości okna przeglądarki. Dodatkową korzyścią jest fakt, iż zobaczymy całą treść dokumentu bez konieczności przewijania zawartości w poziomie.

Aby uaktywnić tę funkcję należy dopisać do nagłówka nową linię o następującej treści:

@WORDWRAP

Cały nagłówek może więc wyglądać na przykład tak jak poniżej:

@DATABASE "Mój spis"
@AUTHOR "Adam Piotr"
©(c) 2020
@\$VER 1.0 11-Sty-20
@FONT topaz.font 8

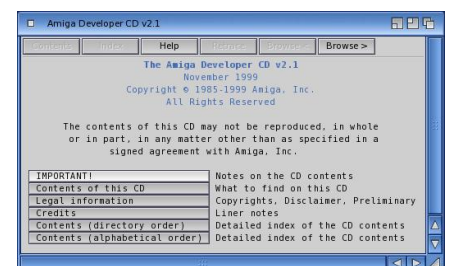
lub

@DATABASE "Mój spis"
@AUTHOR "Adam Piotr"
©(c) 2020
@\$VER 1.0 11-Sty-14
@FONT opal.font 11
@WORDWRAP

To najbardziej potrzebne informacje, które pozwalają utworzyć własny dokument w formacie AmigaGuide. Oczywiście dostępnych jest dużo więcej funkcji, które możemy omówić w innym artykule. Warto dodać, że instrukcję można znaleźć w sieci pod poniższym adresem:

<http://www.lysator.liu.se/amiga/code/guide/amigaguide.guide>

Jak widać, jest to dokument... AmigaGuide! Można go więc bardzo wygodnie przeglądać na Workbenchu. W serwisie Aminet.net można znaleźć również przeglądarkę takich plików dla systemu Windows. Wystarczy przejść do katalogu "misc/emu" i pobrać plik o nazwie "winguide". Program działa nawet na Windows 95, więc z powodzeniem można go wykorzystać na sprzęcie uznawanym już dzisiaj za pełnoprawne retro.



Elementy w oknie będą wyglądały nieco inaczej na każdej wersji systemu Amigi. Tutaj widać domyślny wygląd w systemie AmigaOS 4.

Najprostsze zabezpieczenie dysku

Jak można najprościej zabezpieczyć dysk przed zapisem? Zazwyczaj będziemy myśleć o dodatkowych programach jakie można znaleźć na Aminecie lub innych źródłach. Można też zastosować programy szyfrujące dane, wtedy co prawda zapis będzie możliwy, ale nie będzie miał sensu. Można jednak prościej - użyć jednego z podstawowych poleceń AmigaDOS w systemie 3.0.

Jak wiadomo, każde urządzenie może zostać zabezpieczone przed zapisem. W przypadku dyskietek jest to osiągane przez mechaniczny przełącznik z tyłu dyskietki. Gdy mamy do czynienia z twardym dyskiem nie ma takiej możliwości. Możemy za to skorzystać z polecenia LOCK w oknie Shell. Należy tylko wprowadzić nazwę urządzenia oraz argument ON lub OFF oznaczający włączenie i wyłączenie zabezpieczenia. Na przykład:

lock DH0: ON

spowoduje zabezpieczenie przez zapisem dysku o symbolu „DH0:”, natomiast:

lock DH2: OFF

wyłączy zabezpieczenie na dysku „DH2:”. Po wprowadzeniu każdej linii zobaczymy stosowny komunikat o tym, że funkcja została włączona:

DH0: lock

lub wyłączona:

DH2: unlocked

Gdy na zabezpieczonym dysku będziemy chcieli użyć opcji zapisu, wyświetlone zostaną informacje świadczące o braku takiej

możliwości, tak samo jak na dyskietce. Mogą mieć formę okna albo zwykłego tekstu w oknie Shell:

dysk jest zabezpieczony przed zapisem

Co ciekawe, dodatkowo możliwe jest ustalenie hasła, które będzie wymagane do wyłączenia zabezpieczenia. Należy je podać na końcu linii polecenia, na przykład:

lock DH0: ON 56B6h86Okp92

lub

lock DH0: ON „Amiga to jest to”

Jeżeli w hasło zawarte są odstępy, całość trzeba wpisać w cudzysłowie. Jest to typowe zachowanie AmigaDOS i warto to dobrze zapamiętać. Aby odbezpieczyć dysk należy później wprowadzić hasło w taki sam sposób, na przykład:

lock DH0: OFF „Amiga to jest to”

W przeciwnym razie operacja nie zostanie wykonana, a w zamian zobaczymy komunikat o treści:

Attempt to lock drive DH0: failed

Omówione zabezpieczenie jest najprostszą formą zablokowania dostępu do wykonywania zmian na wybranym urządzeniu. Tak jak wspomniałem na wstępie, można oczywiście wykorzystać zewnętrzne programy, ale wtedy musimy je mieć zainstalowane na stałe w systemie.

Użycie polecenia LOCK pozwala na proste zabezpieczenie korzystając wyłącznie z oryginalnych dyskietek systemowych, tak więc każdy powinien sobie z tym bez problemu poradzić.

```

AmigaShell
New Shell process 4
4.Ram Disk:> lock df0: on
df0: locked
4.Ram Disk:> lock df0: off
df0: unlocked
4.Ram Disk:> lock df0: on 12p5
df0: locked
4.Ram Disk:> lock df0: off
Attempt to lock drive df0: failed
4.Ram Disk:> lock df0: off 12p5
df0: unlocked
4.Ram Disk:>
  
```

Przykład bardzo prostego, ale jednak skutecznego sposobu zabezpieczenia dowolnego dysku w oknie Shell.

Przypisania raz jeszcze

Najprostszym sposobem utworzenia przypisania jest użycie systemowego polecenia ASSIGN w następującej formie:

assign Katalog: SYS:Devs/DOS-Drivers/

Jest to oczywiście tylko przykład. Powyższe polecenie spowoduje dodanie do systemu urządzenia logicznego o nazwie „Katalog”. Będzie prowadzić do katalogu DOSDrivers zapisanego w Devs na dysku systemowym. Dzięki temu, gdy chcemy do niego przejść, wystarczy wpisać linię:

cd Katalog2:

Bez użycia przypisania trzeba wpisać całą długą ścieżkę dostępu, czyli:

cd SYS:Devs/DOSDrivers/

To samo dotyczy innych operacji wykonywanych w oknie Shell. Pozwala to w wielu przypadkach uprościć i przyspieszyć pracę. Sposób ten jest często wykorzystywany podczas instalowania programów lub gier na twardym dysku. Na przykład, gdy system będzie chciał odczytać dyskietkę o nazwie Gra, możemy przegrać jej zawartość na dysk i utworzyć skrót w postaci urządzenia logicznego. Wystarczy napisać:

assign Gra: Work:Gry/gra3/

Ścieżka dostępu „Work:Gry/gra3/” wskazuje oczywiście na katalog, do którego skopiowane zostały wszystkie pliki i katalogi umieszczone wcześniej na dyskietce. Sposób ten nie zawsze się sprawdza ze względu na nietypowe wymagania wielu

Już kilka lat temu sądziłem, że każdy Amigowiec wie, czym są urządzenia logiczne, czyli popularne "przypisania". To jest dość prosty mechanizm, ale jak się okazuje, wiele osób ciągle nie wie, jak je tworzyć lub zmieniać. Dlatego w tym artykule zebrałem najważniejsze informacje, które mogą być przydatne szczególnie dla nowych Amigowców lub osób powracających do Amigi po latach.

gier, lecz bardzo dobrze obrazuje to możliwości wykorzystania polecenia ASSIGN.

Można więc powiedzieć, że „przypisujemy” katalog na dysku do urządzenia i określonej nazwie. Trzeba pamiętać, że przypisania nie są automatycznie zapisywane na dysku. Trzeba je dodać do pliku „user-startup”, który znajduje się w systemowym katalogu „S”. Należy wprowadzić tę samą treść jak podajesz w oknie Shell. W zasadzie, można je dodać do dowolnego pliku tekstowego, ale „user-startup” jest wywoływany podczas rozruchu systemu, więc jest to najlepsze miejsce.

Warto wiedzieć, że system operacyjny tworzy bardzo wiele urządzeń logicznych podczas startu Workbenchu. Niektóre związane są z fizycznymi gniazdami na płycie głównej, jak na przykład „SER:” (port szeregowy) lub „PAR:” (port równoległy). Część z nich jest przypisanych do funkcji, które nie mają swojego odzwierciedlenia sprzętowego, umożliwiając jednak

korzystanie z dodatkowych możliwości w ramach AmigaDOS.

Bardzo często korzystamy z „Ram Dysku”. On również jest utworzony za pomocą urządzenia logicznego „RAM:”. Jak w przypadku każdego urządzenia, jego nazwę należy zakończyć znakiem dwukropka. Dotyczy to wszystkich urządzeń w systemie Amigi i należy to zapamiętać, w przeciwnym razie nie uruchomimy wielu podstawowych funkcji.

Specyficznym urządzeniem jest natomiast „NIL:”. Jest traktowany jako systemowy śmietnik, lecz nie ma to związku z funkcją Kosza na Workbenchu. Dane wysłane do „NIL:” po prostu... znikają. Używamy go wtedy, gdy nie chcemy wyświetlać na ekranie informacji generowanych przez program. Można to wykorzystać przy dowolnej operacji, na przykład podczas rozpakowania archiwum LHA. Czasem może to doprowadzić do przyspieszenia pracy. Przykładowo, zamiast pisać:

Iha x Work:Archiwum.Iha RAM:

możesz użyć linii:

Iha x >NIL: Work:Archiwum.Iha RAM:

Nie spowoduje to zmiany funkcji wywoływanego programu, lecz wyłączy jego komunikaty. Dotyczy to jednak wszystkich informacji, także tych związanych z ewentualnymi błędami, trzeba więc uważać, aby program mógł zadziałać prawidłowo. Symbol „NIL:” jest szeroko stosowany i aby się o tym przekonać najlepiej zajrzeć do pliku „startup-sequence” w katalogu „S”. Znajdziemy go w wielu liniach, obok podstawowych funkcji wywoływanych podczas uruchamiania systemu.

Polecenie ASSIGN pozwala nie tylko tworzyć nowe urządzenia, lecz także wyświetlić ich listę. Wystarczy wpisać samą nazwę lub uzupełnić ją o dodatkowy argument LIST, tak jak poniżej:

assign LIST

W oknie zobaczymy wtedy długą listę urządzeń podzieloną na trzy grupy: Volumes, Directories i Devices. Pierwsza odnosi się do wolumenów, czyli nazw dysków, w odróżnieniu od symboli urządzeń. Druga grupa to przypisania do katalogów, w trzeciej natomiast ujęte są wyłącznie urządzenia fizyczne. Każdą z tych grup możesz wyświetlić osobno, wystarczy użyć argumentu DEVICES, VOLS lub DIRS.

Każde urządzenie logiczne można również usunąć, nie tylko poprzez zresetowanie Amigi. Robimy to korzystając z argumentu o nazwie REMOVE. Na przykład, aby

```

AmigaShell
4.Ram Disk:> assign list
Volumes:
Ram Disk [Mounted]
Workbench3.0 [Mounted]

Directories:
HELP          <LOCALE:Help>
KEYMAPS       Workbench3.0:Devs/Keymaps
PRINTERS      Workbench3.0:Devs/Printers
REXX          Workbench3.0:S
CLIPS         Ram Disk:Clipboards
T             Ram Disk:T
ENV           Ram Disk:ENV
ENVARC        Workbench3.0:Prefs/Env-Archive
SYS           Workbench3.0:
C             Workbench3.0:C
S             Workbench3.0:S
LIBS          Workbench3.0:Libs
              + Workbench3.0:Classes
DEVS          Workbench3.0:Devs
L             Workbench3.0:L

Devices:
PIPE RAM CON RAW SER
PAR PRT DFB DFI
4.Ram Disk:>
  
```

Domyślne przypisania systemowe Workbencha 3.0.

skasować przypisanie o nazwie Atari wystarczy wprowadzić:

assign Atari: REMOVE

W podobny sposób możemy usunąć z systemu zamontowane urządzenie fizyczne. Należy użyć argumentu DISMOUNT, czyli dla dysku „S” będzie to wyglądało następująco:

assign DH2: DISMOUNT

Gdy spróbujemy utworzyć przypisanie do katalogu, którego nie ma na dysku zobaczymy komunikat zawierający słowa „Can't find”. Jego też można wyłączyć dodając na końcu argument DEFER. Urządzenie możemy powiązać z wymiennym nośnikiem, takim jak dyskietka czy płyta CD. System operacyjny będzie wtedy żądał, aby za każdym razem użyć tego samego nośnika, na przykład konkretnej dyskietki z programem. W niektórych przypadkach

wygodniej byłoby, gdyby system odczytywał dane nie z tego samego katalogu, lecz z katalogu o tej samej nazwie. Dzięki temu możliwe jest korzystanie z wielu dyskietek czy płyt i odczytywanie różnych plików. Wystarczy, że nośnik włożony będzie do tego samego urządzenia i posiadał będzie katalogi o tych samych nazwach. Można to osiągnąć przez użycie argumentu PATH.

Zobaczmy przykład:

assign FONTS: DF0:Czcionki PATH

Dzięki temu, jako systemowe urządzenie „FONTS:” traktowany będzie katalog o tej nazwie Czcionki na dyskietce włożonej do wewnętrznej stacji. Gdy włożymy kolejną dyskietkę system odczyta nowe pliki, które będą na niej zapisane. Musimy tylko pamiętać, aby był na niej być utworzony katalog Czcionki, w przeciwnym razie dyskietka zostanie pominięta.



Podobnie można tworzyć urządzenie logiczne obejmujące wiele różnych katalogów na dysku. Służy do tego argument o nazwie ADD. Odnosząc to do poprzedniego przykładu, przyjmijmy, że mamy wiele plików zapisanych na kilka katalogów, na przykład różne kroje czcionek. Dzięki temu można szybciej odczytywać ich zawartość, co ma znaczenie jeżeli zapiszemy dużą ilość plików.

Aby jednak wszystkie kroje były dostępne w systemie, do urządzenia „FONTS:” musimy przypisać wszystkie katalogi. Może to wyglądać tak:

assign FONTS: Work:Czcionki1 ADD

assign FONTS: Work:Czcionki2 ADD

assign FONTS: Work:Czcionki3 ADD

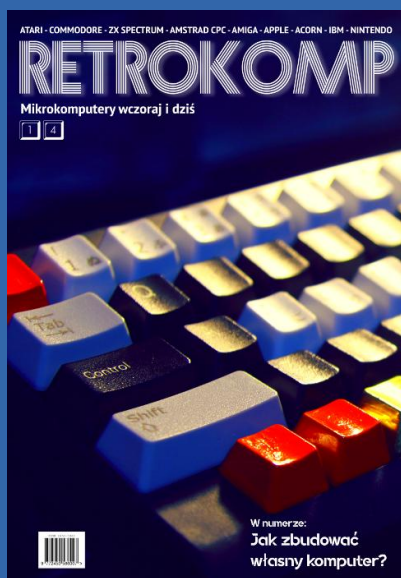
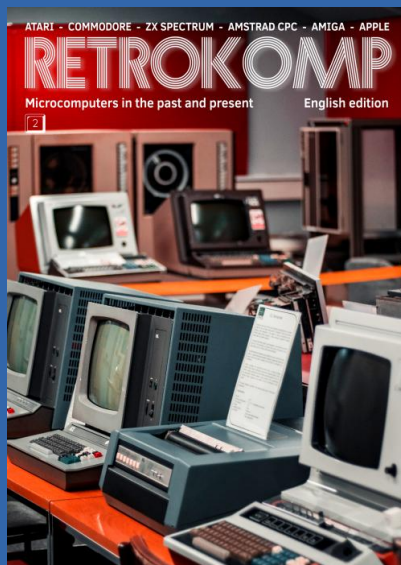
Wprowadzamy kilka poleceń i każde następne dopisuje nowy katalog do tego samego urządzenia systemowego. Poprzednie dane nie są tracone. To samo można osiągnąć za pomocą jednego polecenia:

assign FONTS: Work:Czcionki1 Work:Czcionki2 Work:Czcionki3

Jak widać wystarczy podać po kolei wszystkie nazwy katalogów, które mają zostać uznane za jedno urządzenie logiczne, czyli przypisanie.

Wszystkie katalogi będzie można odczytać na liście urządzeń wywoływanej przez argument LIST lub po prostu wpisując samą nazwę polecenia. Dzięki temu możemy sprawdzić czy wprowadzone zostały właściwe polecenia i czy system na pewno będzie odczytywał prawidłowe katalogi.

CO NOWEGO U NAS?



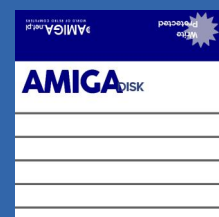
Magazyn RetroKomp
teraz drukowany jest
na papierze kredowym
w pełnym kolorze

Do najnowszego numeru
dołączamy demo nowej
gry Verge World w wersji
na Amigę oraz Atari ST

NIE PRZEGAP OKAZJI!



Pierwsza płyta z serii "Extra CD". Zawiera zestaw ratunkowy pozwalający uruchomić system operacyjny Amigi oraz oprogramowanie narzędziowe (menadżery plikowe, programy do odzyskiwania danych, sterowniki napędu CD, czytnika kart CF i inne) bez korzystania z twardego dysku.



W komplecie również
dyskietka ratunkowa oraz
nalepki na dyskietki
Amiga Disk
(10 sztuk 2 x 5 kolorów).

Stare metody kompresji

Czy w MorphOS-ie można używać starych formatów kompresji plików? Oczywiście! A czy trzeba instalować emulator? Nic z tych rzeczy. Programy napisane w zgodzie z systemem uruchomią się bez problemów, nawet jeśli stworzono je dawno temu, a konkretnie w pierwszej połowie lat '90-tych. Weźmy za przykład archiwizery ARC i ZOO, które są charakterystyczne dla dyskietek przeznaczonych dla systemu 1.3 i zwykłej Amigi 500. Przy okazji uwaga: w taki sam sposób można korzystać z obu programów na praktycznie każdym systemie Amigi. Nie jest to więc cecha szczególna systemów NG.

Aby skorzystać z wymienionych programów w systemie MorphOS wystarczy pobrać z Aminetu odpowiednie pliki. Wiele osób dawno zapomniało, jak z nich korzystać, dlatego warto się im przyjrzeć po latach. Zaczniemy od ZOO. Jest to dość egzotyczny produkt, który zyskał większą popularność tylko na początku lat '90-tych. Co ciekawe, na klasycznej Amidze potrafi działać bardzo niestabilnie, co ujawnia się częstym występowaniem okna „Guru Meditation”. Warto go poznać głównie dlatego, że na niektórych starszych składankach oprogramowania można znaleźć pliki z rozszerzeniem „.zoo”.

Program znajduje się na Aminecie w katalogu „util/arc”. Należy pobrać plik o nazwie „zoo2-10.lzh” (mimo rozszerzenia, wciąż jest to archiwum programu LHA), a

następnie rozpakować w zwykły sposób. Archiwum zawiera tylko jeden plik „zoo2-10”, który proponuję skopiować do systemowego katalogu „C”.

Program ma funkcje zbliżone do najbardziej typowego dla Amigi archiwizera LHA. Jeśli wpisemy samą nazwę programu, wywołane zostaną podstawowe informacje, które tym razem są bardzo skromne. Aby zobaczyć dłuższy opis możliwości trzeba wpisać poniższą linię:

zoo h

W ten sposób możesz uzyskać podręczną pomoc dotyczącą poszczególnych możliwości.

Dodawanie plików do nowego archiwum to podstawowa funkcja każdego archiwizera. Tutaj

wykonujemy ją przy użyciu funkcji „a” lub dłużej „-add”. W tym miejscu warto wspomnieć, że w przypadku głównych operacji program przyjmuje dwojakie nazwy. Pozwala to łatwiej zapamiętać najczęściej wykonywane czynności, dlatego będziemy podawać oba warianty. Aby utworzyć archiwum stosujemy poniższy schemat:

zoo a Work:Dane.zoo SYS:Libs/asl.library

lub dłużej:

zoo -add Work:Dane.zoo SYS:Libs/asl.library

W oknie pojawi się komunikat zawierający liczbę wyrażoną w procentach określającą szacowany zysk z kompresji. Możesz też spowodować, aby dane były automatycznie usuwane z katalogu źródłowego po spakowaniu. Dzięki temu nie trzeba ręcznie sprawdzać które pliki są już w archiwum, a które trzeba pozostawić na dysku. Aby tak się stało używamy dodatkowej opcji „M”, dodając ją do krótszej wersji polecenia:

zoo aM Work:Dane.zoo SYS:Libs/asl.library

albo zamieniając pełną nazwę funkcji:

zoo -move Work:Dane.zoo SYS:Libs/asl.library

Poza usunięciem pliku program zachowuje się identycznie jak w przypadku funkcji „-add”. Podobnie możemy spowodować wyświetlenie nazw plików umieszczonych w archiwum. W tym przypadku użyjemy funkcji „l”, „v” lub „-list”. Wszystkie dają ten sam efekt i należy



je wpisać bezpośrednio po nazwie polecenia, a następnie uzupełnić linię przez podanie nazwy archiwum wraz z pełną ścieżką dostępu. Na przykład:

[zoo I Work:Dane.zoo](#)

W dokładnie ten sam sposób można sprawdzić czy archiwum nie zawiera błędów. Do dyspozycji jest jednak tylko jedna możliwość w postaci funkcji „-test”. Używamy jej tak:

[zoo -test Work:Dane.zoo](#)

Operacja dekompresji plików jest równie łatwa. Możemy skorzystać z funkcji „e” lub „x” - obie działają tak samo. Alternatywnie można wprowadzić cały wyraz „extract” poprzedzając go myślnikiem. Całość może mieć następującą formę:

[zoo e Work:Dane.zoo](#)

lub

[zoo -extract Work:Dane.zoo](#)

Domyślnie program rozpakowuje dane bez zachowania struktury katalogów. Można to zmienić dodając znak ukośnika do oznaczenia funkcji:

[zoo e/ Work:Dane.zoo](#)

albo

[zoo x/ Work:Dane.zoo](#)

Jeżeli nie chcemy obserwować efektów pracy programu, możesz je ukryć. Ma to zastosowanie przede wszystkim wtedy, gdy operujemy na większej ilości plików. W takiej sytuacji usunięcie wyświetlania zbędnych informacji przyspieszy działanie. Aby tak się stało do wprowadzanej funkcji dodaj literę „q” (ang. quiet), na przykład:

aminet

Search

Simple search - [Advanced search](#)

zoo arc

Found 7 matching package(s):
> 1<

name:	version:	path:	dls:	size:	date:	arch:	desc:
acs_zoom.lha		demo/ecs	2137	47K	1994-09-18		Archaio
arc.lha		util/arc	4246	36K	1994-04-26		Nine ye
LAZI-1_1.lzh		util/arc	548	112K	1992-04-16		Lharc/A
LHCon.lha	1.01	util/arc	406	20K	1993-02-15		Conver
lzconv2.lha	2.0	util/arc	528	102K	1997-10-23		Conver
LZX_Repack204b.lha		util/arc	632	134K	1995-12-18		LHA/DI
zoo-2.10.tar.Z		misc/unix	1015	240K	1993-03-19		Archive

Found 7 matching package(s):
> 1<

Stare archiwizery można nadal pobrać z serwisu Aminet. Choć wydano je w pierwszej połowie lat '90-tych, bez problemu działają w MorphOS-ie.

[zoo aq Work:Dane.zoo SYS:Libs/asl.library](#)

W rezultacie „ZOO” przeprowadzi kompresję pliku w zwykły sposób, lecz nie pokaże żadnej informacji w oknie Shell. Program jest dość prosty, ale udostępnia większość najbardziej potrzebnych funkcji.

Drugim archiwizerem jest ARC. Jest to najstarszy archiwizer z jakim możemy mieć do czynienia na Amidze. Powstał w 1985 roku i – co ciekawe – był pisany także w wersji dla komputerów PC, ale nie uzyskał większej popularności. Program posiada spore możliwości, działa też dużo bardziej niezawodnie. Znajdziemy go oczywiście na Aminecie, w katalogu „util/arc”. Plik, który pobierzemy nosi nazwę „arc.lha” i należy go rozpakować w standardowy sposób. Wewnątrz znajdziemy plik o nazwie „Arc”, który znowu kopiujemy do systemowego

katalogu „C”. Dodawanie plików do archiwum jest bardzo podobnie jak w przypadku poprzednich archiwizerów. ARC posiada jednoliterowe nazwy poszczególnych funkcji. W tym przypadku korzystamy z funkcji „a”, która powoduje kompresję wybranego pliku. Na przykład:

[arc a Work:Dane.arc C:Installer](#)

Spowoduje to wyświetlenie komunikatów takich jak „analyzing” i „crunching”. Zamiast tego można także zobaczyć napis „storing”, co oznacza, że plik nie może być spakowany tak, aby uzyskać mniejszy rozmiar. W takiej sytuacji będzie zajmował tyle samo co przed dodaniem do archiwum.

Możemy również zastosować funkcję „m”, która różni się od poprzedniej tym, że po dodaniu pliku do archiwum zostanie on

```

AmigaShell
4.Work:> zoo2-10
zoo 2.1 $Date: 91/07/09 02:10:34 $

Usage: zoo {acDeglLPtUvX}[aAcCdEfInMNoOpPqul!/.@n] archive file
("zoo h" for help, "zoo H" for extended help)

Novice usage: zoo -cmd archive[.zoo] file... where -cmd is one of these:
-add -extract -move -test -print -delete -list -update -freshen -comment
4.Work:>

```

Program ZOO nie ma zbyt dużej ilości funkcji, ponadto plik z pomocą jest bardzo skromny.

automatycznie skasowany z dysku. Za pomocą ARC nie można niestety pakować całych katalogów. Gdy spróbujemy to zrobić, na przykład podając samą ścieżkę dostępu, program stwierdzi brak operacji do wykonania.

Aby zobaczyć jakie pliki umieszczone zostały w archiwum trzeba użyć kolejnej typowej funkcji „l” w taki sam sposób, czyli przykładowo:

arc l Work:Dane.arc

W oknie Shell pojawi się tabela zawierająca trzy kolumny. Są to nazwy plików poddanych kompresji, ich objętości wyrażone w bajtach oraz daty utworzenia. Na samym końcu widzimy podsumowanie całości. Informacje te w większości przypadków wystarczają do określenia zawartości archiwum, możemy jednak skorzystać z kolejnej

funkcji „v”, które pokaże więcej parametrów.

W dodatkowych kolumnach będzie widać zysk z kompresji („SF”), a w podsumowaniu – średnią objętość, o którą różnią się wszystkie spakowane pliki w stosunku do swoich źródłowych wersji. Ponadto widzimy rozmiar po spakowaniu (Size now), godzinę utworzenia lub modyfikacji (Time) oraz sumę kontrolną pliku (CRC). Jest to wartość obliczona według algorytmu tzw. cyklicznego kodu nadmiarowego (Cyclic Redundancy Check)

W pozycji „Stowage” powinien natomiast pojawić się wpis „Crunched” lub tylko znaki myślnika. Ten pierwszy oznacza zwykłą sytuację, czyli skompresowany plik. Dwa myślniki pojawiają się, gdy podczas dodawania pliku do archiwum program nie uzyska

żadnego zysku z kompresji. Można to łatwo sprawdzić, bowiem w takiej sytuacji w kolumnie „Length” oraz „Size now” będzie widoczna ta sama wartość.

Dokładnie tak samo możemy wywołać funkcję testowania prawidłowości pliku archiwum. Zamiast „l” musimy jednak wprowadzić literę „t”:

arc t Work:Dane.arc

W oknie zaczną pojawiać się linie rozpoczynające się od słów „Testing file”. Dalej umieszczona będzie nazwa sprawdzanego pliku. Jeżeli nie wystąpią żadne błędy na końcu pojawi się napis „No errors detected”. Mogą też pojawić się napisy takie jak „Cannot read archive” lub „is not an archive or is out of sync”.

W pierwszym przypadku oznacza to kłopoty z odczytaniem pliku, na przykład wtedy, gdy podaliśmy złą nazwę lub uszkodzony jest nośnik, gdzie zapisane zostało archiwum. Drugi komunikat wskazuje na możliwość uszkodzenia struktury pliku albo wskazanie pozycji, która nie stanowi archiwum – przynajmniej w rozumieniu mechanizmów programu ARC.

Po utworzeniu archiwum można stwierdzić, że nie potrzebujemy już niektórych zapisanych w nim plików. Normalnie trzeba rozpakować całość i jeszcze raz poddać kompresji wybrane dane. ARC umożliwi usunięcie pozycji bez takiej potrzeby, co ogromnie przyspiesza i ułatwia pracę.



Wystarczy skorzystać z funkcji „d” w następujący sposób:

arc d Work:Dane.arc Avail

Następną typową czynnością jest dekompresja plików. Możemy ją wywołać bardzo łatwo – przez użycie funkcji „e” lub „x”, analogicznie do poprzednich przykładów:

arc e Work:Dane.arc

albo

arc x Work:Dane.arc

Obie działają tak samo, pozostawienie obu symboli ma na celu zachowanie zgodności z innymi archiwizatorami. Jeśli podczas rozpakowania archiwum program odnajdzie na dysku plik o tej samej nazwie co w archiwum – wyświetli komunikat „WARNING: File already exists! Overwrite it (y/n)?”. Trzeba

```
AmigaShell
New Shell process 4
4.Ram Disk:> work:
4.Work:> arc
4.Work:Arc> arc a ram:archiwum c:iprefs c:avail c:info c:list
(creating new archive: RAM:ARCHIWUM.ARC
Adding file: Avail      analyzing, crunching, done.
Adding file: Info      analyzing, crunching, done.
Adding file: IPrefs     analyzing, crunching, done.
Adding file: List      analyzing, crunching, done.
4.Work:Arc>
```

Tworzenie przykładowe archiwum z 4 plikami za pomocą programu ARC.

odpowiedzieć, czy chcemy nadpisać plik o takiej samej nazwie znajdujący na dysku.

Podobna informacja pojawi się dla każdego pliku, który może zostać zastąpiony. Jeżeli zdecydujemy się pozostawić stary plik na dysku, w oknie pojawi się kolejna linia zawierająca nazwę pliku oraz napis „not extracted”. Przy wywoływaniu operacji dotyczących większej ilości danych może zaistnieć potrzeba

zatrzymania pracy programu. W tej sytuacji naciskamy kombinację klawiszy CONTROL oraz C, czyli tak jak zatrzymujemy każdy skrypt wykonywany w oknie Shell. Zwykle w rezultacie pojawia się tylko stosowna informacja, jednak w przypadku ARC zobaczymy oddzielne okno. Za pomocą przycisków „CONTINUE” i „ABORT” możemy kontynuować pracę lub przerwać aktualną czynność.

Warto dodać, że jeśli wskażemy pierwszy z nich, operacja nie zostanie przerwana, a jedynie zatrzymana na czas wyboru odpowiedniej opcji. Dzięki temu mamy możliwość przywrócenia poprzedniego stanu nawet wtedy, gdy teoretycznie praca została już wcześniej zatrzymana. Nawet stare archiwizery posiadają wiele ciekawych opcji.

Oczywiście nie namawiam Was, abyście nagle zaczęli używać mało znanych programów zamiast pozycji takich jak LHA, ZIP czy RAR. Jednak czasami warto sprawdzić, czy przypadkiem stary i teoretycznie kiepski program nie może godnie zastąpić popularnych odpowiedników. W wielu wypadkach można się zdziwić.

```
AmigaShell
2) It is distributed ONLY in its original, unmodified state.

If you like this program, and find it of use, then your contribution will
be appreciated. You may not use this product in a commercial environment
or a governmental organization without paying a license fee of $35.

If you fail to abide by the terms of this license, then your conscience
will haunt you for the rest of your life.

Usage: ARC {amufdxerplvtc}[bswn][g<password>] <archive> [<filename> . . .]
Where:  a = add files to archive
        m = move files to archive
        u = update files in archive
        f = freshen files in archive
        d = delete files from archive
        x,e = extract files from archive
        p = copy files from archive to standard output
        l = list files in archive
        v = verbose listing of files in archive
        t = test archive integrity
        c = convert entry to new packing method

        b = retain backup copy of archive
        s = suppress compression (store only)
        w = suppress warning messages
        n = suppress notes and comments
        g = Encrypt/decrypt archive entry

Please refer to the program documentation for complete instructions.
4.Work:Arc>
```

Archiwizer ARC jest dużo bardziej rozbudowany, do tego podręczna pomoc opisuje dużo lepiej wszystkie jego funkcje.

Okna w Blitz Basicu

Okna są podstawowym elementem interfejsu graficznego systemu Amigi. Korzysta z nich prawie każdy program, a nawet jeśli uruchamiamy tylko polecenie AmigaDOS, wpisujemy je w oknie „Shell”. Jeśli chcemy stworzyć nową grę być może nie będzie trzeba używać standardowych okien, natomiast w przypadku program użytkowego należy zachowywać wszystkie możliwe elementy systemu operacyjnego.

Blitz Basic pozwala na korzystanie z okien, podobnie jak ekranów, o których mówiliśmy w poprzednich numerach Amigazynu. Do dyspozycji mamy szereg poleceń, za pomocą których można otwierać, zamykać okna oraz wywoływać wiele operacji potrzebnych użytkownikowi podczas obsługi programu.

Na początek przyjrzymy się bliżej prostemu przykładowemu listingowi, który otwiera okno na ekranie Workbench:

FindScreen 0

Window 1,10,70,500,80,\$10,"Moje okno",1,2

Activate 1

MouseWait

Jak widać okno można stworzyć w bardzo łatwy sposób - wystarczy wykorzystać polecenie WINDOW. Okno będzie wyświetlane na aktywnym ekranie, a więc należy otworzyć własny lub użyć ekranu Workbench. Standardowe okno posiada przyciski na ramce, ale znane są też okna pozbawione niektórych elementów.

Dlatego WINDOW ma kilka argumentów, za pomocą których ustawiamy parametry okna. Aby otworzyć okno w najprostszy sposób, obok nazwy polecenia należy podać następujące argumenty:

- numer okna,
- współrzędną poziomą lewego górnego rogu,
- współrzędną pionową lewego górnego rogu,
- szerokość okna,
- wysokość okna,
- flagi przypisane do okna,
- tytuł okna.

Oznacza to użycie linii według poniższego schematu:

Window 1,100,100,300,50,\$9,"Moje okno",1,2

Większość pozycji nie wymaga komentarza, ale czym są tajemnicze flagi? Są to wartości określające parametry okna, na przykład czy ma ono zawierać przycisk zamknięcia, czy możliwe ma być zmiana rozmiaru i inne. Wartość flag należy podawać rozpoczynając znakiem dolara (\$). Listę wraz z

krótkim objaśnieniem przedstawiam w ramce.

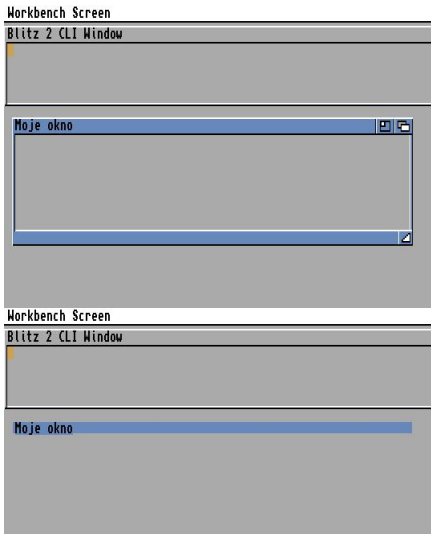
Po ustaleniu, z jakich pozycji chcemy skorzystać, wartości należy zsumować, a uzyskaną wartość wpisać jako jeden argument. Na przykład:

Window 1,0,0,200,35,\$25,"Moje drugie okno",1,2

W powyższej linii mamy wartość \$25, którą otrzymaliśmy z sumowania pierwszych pięciu pozycji z listy flag. Okno będzie teraz miało przycisk zmiany rozmiaru oraz standardowe pola w prawym górnym rogu. Ta sama zasada dotyczy innych wartości z listy. Mamy na niej również opcje pozwalające określić sposób traktowania ramki okna oraz wyświetlania okna na aktywnym ekranie.

Na końcu linii z poleceniem WINDOW dodajemy dwa dodatkowe argumenty, które ustawiają kolorystykę okna. Pierwszy argument określa kolor, za pomocą którego wyświetlany będzie tytuł okna. Druga liczba to numer koloru używanego do rysowania zewnętrznej ramki okna. Dzięki temu możemy dostosować wygląd okna do palety barw na używanym ekranie, a także specyficznych funkcji programu. Daje to dodatkowe możliwości projektowania interfejsu użytkownika.

Jeśli otworzymy kilka okien jednocześnie, przed wykonaniem operacji trzeba określić, z którego chcemy skorzystać. W przeciwnym razie można wywołać nieprzewidziane rezultaty lub program zostanie zatrzymany z powodu błędu. Trzeba pamiętać, że



program może być obsługiwany przez użytkownika w różny sposób i należy to przewidzieć.

Aby ustawić okno o określonym numerze jako aktualne - czyli to, z którego będziemy korzystać - wystarczy użyć polecenia USE WINDOW, podając obok numer. Niektóre okna powinny mieć przypisane swoje własne minimalne i maksymalne rozmiary. Ma to związek przede wszystkim w ich zawartością lub ułożeniem, gdy zastosujemy własny ekran o ściśle określonych parametrach. Aby ograniczyć możliwość zmiany wielkości należy skorzystać z kolejnego polecenia o nazwie SIZELIMITS. Oto jego sposób użycia:

SizeLimits 50,30,200,50

Najpierw podajemy minimalny rozmiar w poziomie i pionie, a następnie maksymalny – według tej samej kolejności. Linia ta powinna znaleźć się przed poleceniem powodującym otwarcie nowego okna. Dopóki nie wprowadzimy kolejnej linii modyfikującej limity, będą one obowiązywać. Można również umieścić słowo SIZELIMITS po

otwarceniu okna, ale wtedy każde nowe okno będzie zachowywało te same parametry. Oczywiście w każdej chwili można to zmienić i wszystko zależy od sposobu działania programu, umieszczonych w nim pętli, instrukcji warunkowych i innych elementów. Aby wykonać operację na konkretnym oknie, należy najpierw je aktywować. W przeciwnym wypadku bieżącym obiektem może być inne okno i program nie zadziała poprawnie. Na szczęście nie jest to skomplikowana operacja, wystarczy wpisać linię podobną do poniższej:

Activate 1

Jak łatwo się domyślić, oznacza to aktywację okna o podanym numerze. Polecenie to nie ma żadnych dodatkowych argumentów. Jeśli udostępniemy użytkownikowi te możliwości, okna nie powinny zmieniać swoich pozycji bez wyraźnej przyczyny. Z drugiej strony, gdy obiekty będą miały na stałe przypisane rozmiary, z pewnością natrafimy na konieczność modyfikacji parametrów. Okna można przesuwać po ekranie, a także

zmieniać ich rozmiar. Służą do tego polecenia WMOVE i WSIZE. Ich użycie podaję razem, bo oba wymagają podania tylko dwóch argumentów. W pierwszym przypadku będą one oznaczały pozycję na ekranie, w jakiej ma znaleźć się okno, a w drugim – żądaną szerokość i wysokość okna. W obu przypadkach liczby oznaczają punkty na ekranie. W związku z tym linie z wymienionymi poleceniami będą wyglądały tak:

WMove 150,50

lub

WSize 200,100

Oczywiście każde okno trzeba kiedyś zamknąć. Do tego z kolei służy polecenie FREE WINDOW, po którym podajemy po prostu numer okna. Dodajmy, że możliwość zamknięcia okna przez użytkownika powinna być przewidziana w programie w szczególny sposób, bowiem zmienia sposób obsługi wykonywanych czynności. Być może nie będziemy chcieli w ogóle udostępniać opcji zamknięcia okna, ale jeśli już tak się stanie, program musi zareagować na to w konkretny sposób. To już wymaga użycia Eventów, czyli zdarzeń.

Lista flag, które można przypisać do okna:

- \$0001 - dodaje przycisk zmiany rozmiaru w prawym dolnym rogu okna,
- \$0002 - pozwala na zmianę pozycji okna za pomocą myszki,
- \$0004 - okno będzie mogło być umieszczanie „z przodu” lub „z tyłu” innych okien,
- \$0008 - dodaje przycisk zamknięcia okna w lewym górnym rogu,
- \$1000 - aktywuje otwarte okno,
- \$0800 - powoduje, że okno nie będzie posiadało ramek,
- \$0100 - okno będzie otwarte „na spodzie” wszystkich innym aktualnie dostępnych okien.
- \$0400 - powoduje, że elementy umieszczane wewnątrz okna nie będą mogły zasłonić ramki; jest to wygodne, ale zajmuje większą ilość pamięci.

Tak to wygląda dla użytkownika, który uruchamia dyskietkę. Jednak cały proces rozruchu jest dużo bardziej rozbudowany. Przy czym najczęściej na ekranie widocznych jest minimum informacji.

Pamięć ROM w Amidze, czyli Kickstart, składa się z wielu modułów, bibliotek i sterowników urządzeń. Jednym z modułów ma nazwę Strap. Po rozpoznaniu i zainicjowaniu urządzeń wejścia/wyjścia oraz rozszerzeń, moduł ten przejmuje rozruch. Jego zadaniem jest uruchomienie środowiska AmigaDOS z dysku startowego o najwyższym priorytecie. Jeśli sterownik o nazwie trackdisk.device wykryje dyskietkę w napędzie o symbolu DF0, uzyska on domyślnie najwyższy priorytet, więc Amiga uruchomi się z dyskietki.

Uruchomienie dyskietki rozpoczyna się od przeszukania dwóch pierwszych sektorów dysku w poszukiwaniu wspomnianego już bootblocku. Standardowo wywołuje on bibliotekę dos.library, tak więc uruchamiane jest okno CLI/Shell. Odnaleziony może być również niestandardowy bootblock, wtedy jest do wczytywany do przydzielonej pamięci RAM i uruchamiany.

Warto zauważyć, że tylko za pomocą nietypowego kodu możemy zdecydować, czy użyty ma być sterownik systemu plików z Kickstartu, czy wyłącznie wywołać trackdisk.device i uruchomić kod przechowywany na dyskietce w oparciu o własny układ ścieżek i sektorów. W ten sposób dochodzimy do sytuacji, gdy dyskietki dla Amigi są zapisywane w nietypowych formatach, specjalnie stworzonych przez autorów gier i dem sceny komputerowej. Wśród tych produkcji

Bootblock inaczej niż zwykle

Sektor startowy na dyskietce nazywamy bootblockiem. Może on zawierać programy, które uruchamiane będą automatycznie po wystartowaniu Amigi za pomocą dyskietki. Może tam znaleźć się wiele przydatnych funkcji, jak na przykład menu wyboru, dzięki któremu będziemy mieli więcej możliwości podczas rozruchu.

są także niestandardowe bootblocki, które nie wywołują biblioteki dos.library, lecz uruchamiają autorski program ładujący.

Jeśli nie został zapisany autorski blok startowy, wtedy dos.library ma swoją własną metodę inicjalizacji i korzystania z systemu plikowego w celu dalszego rozruchu. Jest ona zgodna z poleceniami AmigaDOS podanymi w sekwencji startowej, która może mieć różną zawartość. Domyślnie wczytywana jest biblioteka workbench.library, tak więc użytkownik zyskuje dostęp do systemowego blatu.

Jeśli nie włożymy dyskietki do stacji, wtedy moduł Strap sprawdzi urządzenia zawierające możliwość rozruchu, wybierze urządzenie startowe o najwyższym priorytecie i wywoła je przeskakując do kodu dostarczonego przez kontroler tego urządzenia, zapisanego w pamięci ROM. W związku z tym proces ten zależy od zawartości układu ROM, ale ogólnie jest podobny do

procesu uruchamiania na dyskietkach.

Główna różnica polega na tym, że dyski twarde zawierają bloki, które mogą mieć zapisane sterowniki systemu plikowego. Dzięki temu dysk twardy również nie musi używać sterowników z Kickstartu. Może on najpierw załadować system plikowy z dysku, a następnie kontynuować proces rozruchu we własny sposób. Tak właśnie działają zaawansowane systemy plikowe, takie jak PFS czy SFS. Jeśli moduł Strap nie znajdzie dyskietki lub innego urządzenia startowego, wyświetli znany ekran z dyskietką, a następnie będzie kontynuował skanowanie w poszukiwaniu nośnika, który ma być włożony przez użytkownika.

Niestety czasem w bootblocku zapisywane są niepożądane programy w postaci różnego rodzaju wirusów, czyli programów zakłócających normalną pracę komputera. Może to prowadzić do problemów z dostępem do danych. Takiego niestandardowego



bootblocku nie zapiszemy, ani nie zmodyfikujemy za pomocą oprogramowania systemowego. W tym celu należy posłużyć się dodatkowymi programami, które jak zwykle znajdują się m.in. w zasobach Amineu. Większość umieszczonych jest w katalogu „disk/misc”.

Jednym z bardzo przydatnych narzędzi, a jednocześnie bardzo łatwym w obsłudze, jest polski program In Boot, który znajdziesz w pliku pod nazwą „2b_InBoot.lha”. Archiwum należy pobrać na dysk, a potem rozpakować przy użyciu okna „Wykonaj polecenie”. Aby uruchomić program wystarczy kliknąć dwukrotnie na jego ikonę. Po chwili pojawi się okno programu. Po lewej stronie widać listę dostępnych stacji dyskietyk.

Należy wskazać na niej nazwę urządzenia lub pozostawić symbol DF0 jeśli posiadamy tylko jedną stację.

Obok umieszczone jest są opcje przeznaczone do manipulowania bootblockiem. Możemy zapisać standardowy sektor startowy, taki sam jak za pomocą polecenia Install. W tym celu należy wybrać opcję „Boot”, a następnie skorzystać przycisku „Instaluj” w prawym dolnym rogu okna.

Gdy napotkamy nietypowy bootblock, którego chcemy się pozbyć trzeba wybrać funkcję opisaną jako „No Boot”. Potem znowu używamy przycisku „Instaluj”. Po chwili bootblock zostanie usunięty z dyskietyki. Zwróć uwagę, że w takiej sytuacji dyskietyka nie będzie mogła służyć do startowania Amigi, chyba że zapiszesz na niej ponownie zwykły sektor startowy. Dyskietykę bez bootblocku możesz

oczywiście stosować do innych celów.

Jeżeli chcesz wyświetlić zawartość bootblocku skorzystaj z funkcji „Bufor”. Aktywna stanie się wtedy opcja „Ładuj” po prawej stronie. Po chwili sektor startowy zostanie odczytany i możliwe do użycia staną się wszystkie opcje wyżej. Wskaż przycisk o nazwie „Pokaż...”. Na ekranie pojawi się okno z wczytanym bootblockiem. Taki sposób prezentacji nie jest zbyt czytelny, jednak sprawdzając większą ilość dyskietek można z czasem dość łatwo zorientować się czy zapisany sektor startowy jest typowy czy też może powodować problemy.

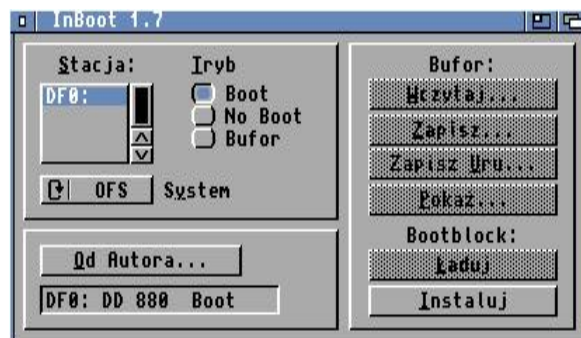
Ponadto program „In Boot” sprawdza treść pod kątem obecności w nim wirusów. Aby funkcja ta była aktywna w systemie musi być zainstalowana biblioteka o nazwie „xvs”. Znajdziesz ją na Aminecie w katalogu „util/virus” pod nazwą „xvslibrary.lha”. Instalacja jest bardzo prosta - wystarczy użyć typowego instalatora lub skopiować plik do katalogu systemowego „Libs”.

Bezpośrednio po dodaniu nowej biblioteki do systemu, „In Boot” będzie mógł sprawdzać czy odczytywane bootblocki nie zawierają wirusów. Może się to przydać w sytuacji, gdy korzystasz z

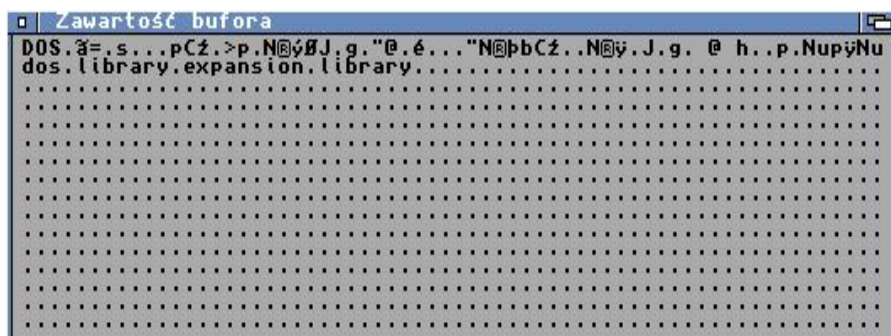
dużej ilości dyskietek lub jeśli wyczerpałeś już wszystkie możliwości usunięcia wirusa, a dyskietyka nadal go zawiera. Trzeba pamiętać, że standardowe programy antywirusowe nie zawsze sprawdzają sektor startowy dyskietyki.

Jeśli chcesz natomiast przenieść sektor startowy na inną dyskietykę, po odczytaniu bootblocku za pomocą funkcji „Ładuj”, wskaż przycisk „Zapisz...” znajdujący się wyżej. Na ekranie zobaczysz okno wyboru, w którym ustal katalog i nazwę pliku, pod którym zapisany zostanie bootblock. Potem użyj przycisku „Zapisz” w dolnym lewym rogu okna. Plik możesz teraz zapisać na innej dyskietyce jako bootblock. W tym celu włóż do stacji dyskietykę docelową, wybierz funkcję „Bufor” i „Ładuj”.

Gdy to zrobisz skorzystaj z przycisku „Wczytaj...”, który w innym przypadku nie byłby aktywny. W oknie wyboru wskaż zapisany wcześniej plik i użyj przycisku „Ok”. Na koniec wskaż jeszcze raz z opcję „Instaluj”. Bootblock zostanie zapisany na dyskietyce. Możesz go również zapisać w formie programu do uruchomienia na Workbenchu. Zamiast używać opcji „Zapisz...” trzeba wybrać widoczną niżej, opisaną jako „Zapisz Uru...”. Sposób obsługi jest identyczny jak w przypadku zwykłego zapisywania pliku.



Program InBoot posiada polską wersję językową, jest łatwy w obsłudze i daje spore możliwości kontroli nad bootblockiem.



Zawartość sektora startowego dyskietki wyświetlona jako tekst.

Weź pod uwagę, że sektor startowy jest specyficznym rodzajem programu związanym ściśle z dyskietką, ponadto jest wykonywany zaraz po wystartowaniu komputera, a zatem w nieco innym środowisku pracy niż Workbench. Dlatego oddzielna próba uruchomienia bootblocku może zakończyć się niepowodzeniem, zawieszeniem systemu lub innym trudnym do przewidzenia rezultatem.

Dla Amigi istnieje wiele innych programów przeznaczonych do przeprowadzania operacji na sektorze startowym dyskietki. Wiele z nich umożliwia uzyskiwanie dodatkowych funkcji lub efektów podczas uruchamiania. Zapisywanie nietypowych sektorów startowych może powodować problemy podczas korzystania z dyskietki. Wiele z nich nie bierze pod uwagę systemu operacyjnego, dlatego przy używaniu omawianych opcji należy zachować szczególną ostrożność, a najlepiej wszystkie operacje wykonywać na kopiach dyskietek. W przeciwnym razie możemy utracić dane lub spowodować, że staną się one niemożliwe do odczytania. Natomiast wszystkich fanów systemu Windows może zainteresować informacja, że jest dostępny

program "Amiga Bootblock Reader", który służy do wykrywania, analizowania i zapisywania amigowych bootblocków z obrazów ADF w zewnętrznych plikach. Program rozpoznaje ok. 1700 różnych bootblocków i może je automatycznie uruchamiać poprzez obrazy dyskietek w WinUAE lub FS-UAE. Jest on rozwijany od wielu lat i posiada ciekawe możliwości, między innymi testowanie bootblocku bez ryzyka uszkodzenia lub zainfekowania dyskietki wirusem, jak również obsługę kompresji ZIP. Bootblock to

specyficzny element w systemie Amigi i wiele osób analizuje ich możliwości lub pokazuje, jak można napisać własny sektor startowy. Najlepiej zacząć od najprostszego przykładu, który można znaleźć pod następującym adresem:

<https://rdrr.io/cran/adfExplorer/man/boot.block.code.html>

Serwis Aminet.net zawiera przynajmniej kilka innych, ciekawych programów służących do obsługi bootblocku. Służą między innymi do konfiguracji parametrów okna CLI, automatycznej aktywacji nowych sterowników urządzeń, wyświetlania komunikatów tekstowych, tworzenia własnego menu z wyborem programów do załadowania czy też zmiany trybów graficznych. Wystarczy wyszukać frazę "bootblock" lub zajrzeć do "disk/misc". Na początek polecam program o nazwie MegaBoot (tym razem w katalogu "util/boot"). Życzę powodzenia w samodzielnych eksperymentach!



Przykład menu wczytywanego bezpośrednio z bootblocku dyskietki. Zwykle posiada kilka prostych opcji, ale można je modyfikować, między innymi za pomocą programów dostępnych na Aminecie.

Użycie ARexxa w programie Music-X

Otóż, możliwe jest ustawienie parametrów bez limitów narzuconych poprzez interfejs użytkownika lub powiązanie funkcji w inny sposób niż przewidzieli to autorzy. Jednym z takich programów jest Music-X w wersji 2.0, a także Deluxe Music, który może być bardzo dobrym uzupełnieniem.

Piszę o tej konkretnej edycji, bowiem wcześniejsze nie obsługiwały ARexxa. Co można dzięki temu osiągnąć bardziej praktycznie? Na przykład, skrypty mogą zliczać liczbę nut w poszczególnych sekwencjach, transponować muzykę na różne sposoby lub tworzyć utwory w sposób automatyczny.

Można nawet sprawić, aby Music-X komunikował się z innymi programami zgodnymi z ARexxem, aby wysyłać dane z Music-X bezpośrednio do Deluxe Music w celu utworzenia zapisu muzycznego naszego utworu. To tylko kilka przykładów, teraz chciałbym pokazać jak napisać działające skrypty ARexxa.

Zacznijmy od prostego skryptu, który transponuje wszystkie nuty w górę o 7 półtonów. Oto listing:

```
/* Program nr 1 */
OPTIONS RESULTS
BeginScan ALL
NextEvent
DO WHILE RESULT = 1
  IF Event.Type = "NOTE" THEN DO
    Event.Num = Event.Num + 7
    ReplEvent
  END
  NextEvent
END
EndScan
```

Choć język ARexx pochodzi ze świata pecetów, jest zintegrowany z systemem Amigi. Pozwala nie tylko na kontrolę funkcji systemowych, lecz także każdego większego programu korzystającego z zasobów systemu. Za pomocą ARexxa można pisać oddzielne skrypty, programy lub próbować automatyzować funkcję zewnętrznego oprogramowania. W ramach tych możliwości często możemy uzyskać efekty dużo szersze niż wskazują na to opcje umieszczone w tych programach. Dlaczego?

Spróbujmy wyjaśnić, jak działa powyższy skrypt.

Pierwsza linia to komentarz, który musi znaleźć się na początku. Następnie nakazujemy ARexxowi przekazywanie argumentów do poleceń, a także pobieranie od nich informacji. Służy do tego zmienna o nazwie RESULT, która jest wykorzystana w linii ze słowami DO WHILE. Następnie sprawiamy, że Music-X będzie analizował cały zapis w bieżącej sekwencji i tu korzystamy już w poleceniu programu obsługiwanego przez ARexxa.

Dalej odczytywany jest pierwszy element i jego dane są automatycznie zapisywane w specjalnej zmiennej o nazwie Event. Mamy pętlę, która jest główną częścią skryptu. Podczas odczytu każdego elementu sprawdzany jest jego rodzaj. Jeżeli jest to NOTE (czyli nuta), dodajemy do niego wartość 7 i korzystamy z polecenia

ReplEvent, aby zmiana była trwała. Czynności te są powtarzane dla zakończy działanie. Ostatnie polecenia to zakończenie "skanowania".

Powyższy skrypt należy zapisać na dysku, następnie wczytać gotową sekwencję do edytora i wybrać opcję "Rexx..." z menu Modules. W oknie wyboru wskazujemy nasz nowy plik i klikamy go dwukrotnie. Skrypt zostanie uruchomiony automatycznie, tak więc sekwencja zostanie modyfikowana bez potwierdzania.

Obsługę ARexxa w Music-X zapewnia moduł o nazwie RexxEdit. Jest on automatycznie instalowany podczas korzystania z programu instalacyjnego. Jednak posiada on błąd polegający na tym, że domyślnie wpisane mogą być nieprawidłowe ścieżki dostępu. Jeśli tak jest należy wyjść z programu, a następnie uruchomić program "Install Modules". Dalej klikamy na

nazwę każdego modułu po kolei i usuwamy z jego ścieżki dwukropek. Na koniec należy wybrać opcję "Save" i ponownie uruchomić Music-X.

Ogólnie rzecz biorąc, RexxEdit pozwala na wykonanie czterech rodzajów operacji:

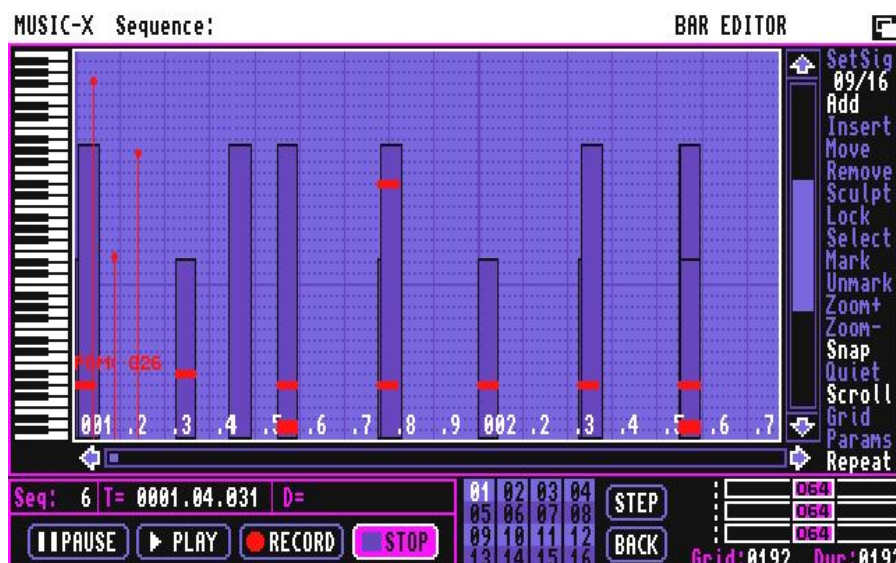
- analiza, dodawanie, usuwanie elementów i zmiana danych w sekwencjach,
- wyświetlanie okien wyboru z przyciskami pozwalającymi wprowadzać dane,
- odczytywanie informacji o stanie edytora takich jak na przykład rozmiar statki,
- komunikowanie się z innymi programami obsługującymi ARexxa.

Za pomocą skryptów można też wykonywać bardziej szczegółowe czynności związane z sekwencjami. Listing zwykle może wyglądać podobnie jak wyżej. Po pierwsze, musi być wywołane polecenie BeginScan w celu analizy sekwencji. Dalej następuje pętla, która sprawdza każdy element przy użyciu słowa NextEvent i przetwarza je w określony sposób.

Gdy polecenie NextEvent zwróci wartość zerową do zmiennej RESULT, pętla jest przerywana i wywołujemy EndScan. Jeśli planujemy dodać elementy do sekwencji za pomocą AddEvent, można to zrobić za pomocą zmiennej Event.

Event. = ""

Musi ona zostać umieszczona przed każdym wywołaniem AddEvent. Trzeba pamiętać, że po nazwie zmiennej trzeba wpisać kropkę. Nie



Edycja muzyki w programie Music-X. Po prawej podstawowe narzędzia, na dole kontrola odtwarzania i nagrywania. Główny obszar roboczy jest widoczny na środku, a widoczne elementy są nazywane zdarzeniami (ang. Event). Nad nimi właśnie możemy mieć kontrolę za pomocą języka skryptowego, jakim jest ARexx.

ma znaczenia, jaką wartość przypiszemy, na przykład może to być:

Event. = 1023012

Nasz poprzedni listing można łatwo zmodyfikować tak, aby zamiast przenosić nuty, dodawał nowe - 7 półtonów nad oryginalnymi. Wymaga to dwóch zmian: zainicjowania zmiennej Event (w początkowych liniach) i zmiany ReplEvent (umieszczonej w pętli) na AddEvent. Później wystarczy zapisać nowy skrypt i uruchomić go - powinien działać bez problemów.

Możemy też napisać skrypt, który doda funkcję Pitch Bend do zaznaczonego obszaru w edytorze. Będzie dodawany w regularnych odstępach czasu - jeden po drugim. Użytkownik będzie mógł natomiast zdefiniować, jak daleko od siebie będą wstawione efekty Pitch Bend.

Jak to zrobić? Informacje o edytorze są sprawdzane za pomocą polecenia GetBarData. Powoduje ono przekazanie do zmiennej BarData informacji, takich jak: bieżący rozmiar siatki, wielkość obszaru roboczego, początek i koniec zaznaczonego regionu i inne. Należy go używać w następujący sposób:

```
/* Program nr 2 */
OPTIONS RESULTS
GetBarData

IF RESULT = 0 THEN DO
  MXReport "Nie mogę odczytać danych"
  EXIT
END
```

Następnie wykrywamy, czy w edytorze został zaznaczony fragment utworu:



Opcje programu podczas nagrywania tak zwanej sekwencji (ang. sequence), czyli wydzielonego fragmentu utworu.

```
IF BarData.Select = "NONE" THEN DO
MXReport "Obszar nie jest zaznaczony"
EXIT
END
```

Oba fragmenty należy wprowadzić razem - w takiej kolejności jak podałem - i zapisać w osobnym pliku. Jest to gotowy, działający skrypt, który wystarczy uruchomić. Jeśli żaden obszar nie został zaznaczony, wyświetlona zostanie informacja o błędzie - wpisana po słowie MXReport.

Teraz umożliwimy użytkownikowi wybór odstępu pomiędzy efektami Pitch Bend. Będzie mógł to określić za pomocą wygodnego przycisku, a także ustawić kanał MIDI za pomocą suwaka. Dokonane wybory zapiszemy w zmiennych o nazwach Spacing oraz MIDI-Channel.

Oto jak wygląda dalsza część skryptu realizująca wspomniane funkcje:

```
MXNoteSize "Ustaw odstęp:"
MXSlider "Kanał MIDI:,1,16,1"
MXRequest "Dodaj efekty Pitch Bend"
```

```
IF RESULT = 0 THEN
EXIT
```

```
MXValue 1
```

```
Spacing = RESULT
MXValue 2
MIDIChannel = RESULT - 1
```

Linie te trzeba wpisać jako kolejne w tym samym skrypcie. Zauważmy, że od wartość zmiennej RESULT odejmujemy wartość 1. Jest to związane z faktem, że Music-X przechowuje wewnętrznie numery kanałów MIDI w zakresie od 0 do 15 (zamiast od 1 do 16).

```
CurrentTime = BarData.Start.Bars * BarData.Measure + BarData.Start.Clocks
EndTime = BarData.Stop.Bars * BarData.Measure + BarData.Stop.Clocks
```

Następnie wyszukamy początek i koniec regionu. Czas w programie jest mierzony w taktach (ang. bars) i

zegarach (ang. clocks), czyli upływie czasu w ramach każdego pomiaru. Na przykład nuta może być wstawiona w takcie o numerze 17 i wartości zegara 2200. Informacje te są przechowywane w zmiennych o następujących nazwach:

- **BarData.Start.Bars** - początek obszaru,
- **BarData.Start.Clocks** - wartość zegara na początku obszaru,
- **BarData.Stop.Bars** - koniec obszaru,
- **BarData.Stop.Clocks** - wartość zegara na końcu obszaru.
- **BarData.Measure** - ogólna liczba zegarów.

Będziemy tu używać własnej zmiennej CurrentTime, która będzie reprezentować bieżącą pozycję w obszarze podczas dodawania elementów. W miarę postępu działania skryptu do aktualnego czasu trzeba wiele razy dodać określoną liczbę zegarów zapisanych w zmiennej Spacing.

Przy dodawaniu elementów musimy jednak znać wartości taktów i zegarów, co będzie wymagało wykorzystania prostego wzoru matematycznego:

(Numer pomiaru) * (Ilość zegarów) + (Numer zegara)

Odnosząc to do naszej skryptu, czas początku i końca naszego obszaru obliczamy tak jak poniżej:

I takie dwie linie należy wpisać jako kolejne do naszego skryptu.



Teraz trzeba wykonać skanowanie obszaru. Niektóre parametry zmiennej Event pozostaną stałe, na przykład kanał MIDI. Musimy jednak ustawić wartość 8192, co będzie oznaczało neutralne ustawienie wirtualnego pokrętki Pitch Bend. W związku z tym następne linie skryptu będą wyglądały tak:

```
Event. = ""
BeginScan
```

```
Event.Type = "PBEN"
Event.Channel = MIDIChannel
Event.Num = 8192
```

Możemy wreszcie dodać konkretne efekty w edytorze. Aby dodać je w odpowiednich miejscach, musimy obliczyć numer taktu i wartość zegara - tam zostanie dodana funkcja Pitch Bend. Numer taktu otrzymujemy dzieląc go przez liczbę całkowitą (operator "%"), a numer zegara za pomocą funkcji Mod lub Remainder (operator "//"). Gotowy listing wygląda następująco:

```
DO WHILE currentTime <= endTime
    Event.Start.Bars = currentTime % BarData.Measure
    Event.Start.Clocks = currentTime // BarData.Measure
    AddEvent
    currentTime = currentTime + spacing
END
```

Kiedy pętla jest zakończona, należy zakończyć skanowanie za pomocą prostej linii, jak poniżej:

```
EndScan
```

MUSIC-X Sequence:
EVENT EDITOR

Event	Time	Ch	Type	Data	
00000	001.01.000	10	NOTE	key=044(G#1) vel=063,064 dur=0000.00.023	SetSig 09/16
00001	01.000	10	PGM	pgm=026	
00002	01.004	10	NOTE	key=035(B0) vel=104,064 dur=0000.00.024	Dup Remove
00003	01.020	10	CTL	ctl=007 val=127	
00004	01.042	10	CTL	ctl=010 val=064	
00005	02.020	10	CTL	ctl=091 val=101	Select Unmark
00006	03.012	10	NOTE	key=046(A#1) vel=063,064 dur=0000.00.023	
00007	04.022	10	NOTE	key=035(B0) vel=104,064 dur=0000.00.024	Undo
00008	05.024	10	NOTE	key=044(G#1) vel=063,064 dur=0000.00.023	Quiet Scroll
00009	05.024	10	NOTE	key=038(D1) vel=104,064 dur=0000.00.023	
00010	05.024	10	NOTE	key=037(C#1) vel=104,064 dur=0000.00.023	
00011	07.036	10	NOTE	key=075(D#4) vel=058,064 dur=0000.00.024	Grid
00012	07.036	10	NOTE	key=044(G#1) vel=063,064 dur=0000.00.023	Params Repeat
00013	07.040	10	NOTE	key=035(B0) vel=104,064 dur=0000.00.024	
00014	0002.01.000	10	NOTE	key=044(G#1) vel=063,064 dur=0000.00.023	
00015	03.012	10	NOTE	key=044(G#1) vel=063,064 dur=0000.00.023	

Event Type: NOTE CAT PAT CTL PGM PBEN SYSX
 PSEQ MTRK STRK MSEQ SSEQ REPT END KMAP TEMPO TSIG STOP STEP BACK
 Channel: 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16
 Seq:006 PAUSE PLAY RECORD STOP A.Velocity: 063 R.Velocity: 064

Edytor zdarzenia, czyli elementu składającego się na gotowy utwór, w większości oczywiście nuty (ang. Note).

Wszystkie linie po kolei trzeba zapisać w jednym pliku. Później, aby korzystać ze skryptu, należy zaznaczyć obszar lub wybrać zapisane elementy. Dalej można już uruchomić skrypt. Kierunek i siłę efektów Pitch Bend zmieniamy za pomocą narzędzia Sculpt Tool. To wszystko.

Warto zwrócić uwagę jeszcze na dwie rzeczy. RexxEdit ma błąd związany z zaznaczaniem obszaru polegający na tym, że wygląda on na zaznaczony, a mimo to pojawia się komunikat "Brak zaznaczonego

time", ponieważ pierwszy z nich nie jest mierzony w zegarach. Dlatego trzeba uważać na te specyficzne i nietypowe ustawienia.

ARexx nie jest skomplikowanym językiem, a można za jego pomocą nie tylko automatyzować czynności, lecz także dodawać nowe funkcje do wielu programów. W następnych numerach naszego pisma napiszę o innych narzędziach, gdzie można wykorzystać praktycznie te możliwości.

W następnym numerze naszego pisma napiszę o kolejnym, tym razem mniej znanym, programie muzycznym, w którym warto użyć języka ARexx. Jego nazwa niech na razie pozostanie tajemnicą.

Szerzej o możliwościach ARexxa piszę w książkach z serii AMIGA OS 3.1, które zostały wydane kilka lat temu. Zapraszam do zakupu wszystkich trzech tomów - łączna objętość to aż 1500 stron!

Zdalne sterowanie

X10 to nazwa standardu systemu automatyki budynkowej. Mówiąc bardziej technicznie, jest to protokół komunikacji pomiędzy urządzeniami elektronicznymi stosowanymi do automatyki w domu, tak zwanej domotyki. Został opracowany w 1975 r. przez Pico Electronics z Glenrothes w Szkocji, w celu umożliwienia zdalnego sterowania urządzeniami i sprzętem domowym. Była to pierwsza technologia sieci domotycznej ogólnego przeznaczenia i nadal pozostaje najszerzej dostępna. Czy nasza poczciwa Amiga jest w stanie współpracować z tym standardem? Wbrew pozorom, tak - istnieje oprogramowanie obsługujące X10.

Protokół ten umożliwia efektywne sterowanie oświetleniem, ogrzewaniem, wentylacją czy klimatyzacją bez konieczności instalowania dodatkowego okablowania. Dlatego bardzo szybko stał się standardem w USA. Jest też stosowany na wszystkich innych kontynentach, choć w ograniczonym zakresie.

Rozróżnia się trzy rodzaje urządzeń X10: nadawcze, odbiorcze i dwukierunkowe. Jeśli domowa instalacja elektryczna jest jednofazowa, nie ma znaczenia, czy urządzenie funkcjonuje w Polsce (50 Hz) czy też w USA (60 Hz) - transmisja ramek trwa krócej niż połowa sinusoidy. Podczas rozszerzania protokołu X10 na trójfazowe instalacje elektroenergetyczne musiał zostać uwzględniony fakt, że w połowie sinusoidy występują trzy przejścia przez zero.

Proste urządzenia X10, jak na przykład sterowniki oświetlenia, instaluje się bezpośrednio do

standardowego gniazda elektrycznego albo montuje się wyłącznik światła przystosowany do tego standardu. Inne, bardziej złożone urządzenia są wyposażane w specjalne moduły X10. Pakiety danych są transmitowane przez układ nadawczo-odbiorczy z przepływnością ok. 10 kb/s, tak więc, nie wymaga to ogromnej przepustowości. System X10 może być bardzo łatwo kontrolowany przez komputer. Trzeba tylko zainstalować oprogramowanie, która pozwala włączać i wyłączać różne elementy w domu. Można również monitorować systemy, które są wyłączone, a także ich status. System automatyki może być też użyty do uzbrojenia lub rozbrojenia systemu alarmowego.

I tutaj dochodzimy do najważniejszego - kilku programów pozwalających na obsługę protokołu X10, które znajdziemy na Aminecie. Jednym z nich jest CyberX10, który można pobrać z katalogu "driver/other". Został

napisany w 1992 roku, a szczerze mówiąc do tej pory nie znalazłem nigdzie opracowania dla naszego komputera. Trzeba to zmienić, co niniejszym czynię.

CyberX10 komunikuje się z urządzeniami takimi jak CP-290, które w oryginalnej dokumentacji wymienia nie tylko "profesjonalny" standard IBM PC, ale także Commodore 64. Posiada proste gniazdo typu DIN-5 i bardzo proste podłączenie tylko 3 przewodów:

- Pin 1 - Niepodłączony
- Pin 2 - Input (wejście)
- Pin 3 - Ground (masa)
- Pin 4 - Output (wyjście)
- Pin 5 - Niepodłączony

Jak z tego wynika, sygnały wejściowe z komputera (wejście danych) są podłączone między pinami 3 i 2, natomiast sygnały wyjściowe do komputera (wysyłanie danych) są podłączone między pinami 3 i 4. CP290 ma dostęp do wszystkich 256 adresów i może wysyłać natychmiastowe zdarzenia (na przykład: przyciemnić światła), a także być zaprogramowany na 128 zdarzeń czasowych, nie wymagając tym samym stałego połączenia z komputerem. Szczegółowe informacje na temat CP-290 znajdziemy w oryginalnej instrukcji, która jest dostępna w sieci:

<https://cdn.shopify.com/s/files/1/2279/4329/files/CP290.pdf>

Interfejs jest zaprogramowany tak, aby rozpoznawać 8 różnych typów instrukcji wysłanych z komputera. Mają one numery identyfikacyjne od 0 do 7. Każda instrukcja z komputera posiada sygnał synchronizacji wzór 16 bajtów w formie szesnastkowej. Numer

identyfikacyjny określa, jakiego rodzaju danych należy się spodziewać i jaka jest prawidłowa suma kontrolna, która jest porównywana z ostatnim bajtem danych polecenia. Jeżeli sumy kontrolne się zgadzają, interfejs potwierdzi przekazanie sterowania do komputera, a następnie wykona przesłaną instrukcję. Producent sugeruje, że jeśli brak jest odpowiedzi na instrukcję w ciągu 10 sekund, komputer powinien poinformować użytkownika o potencjalnym problemie z połączeniem.

Wracając do naszego amigowego programu, CyberX10 można uruchomić zarówno z poziomu okna Shell, jak i na blacie Workbenchu. Posiada kilka argumentów pozwalających na sterowanie interfejsem CP290. Możliwe jest również korzystanie z szeregu parametrów ikon (tooltypes). Wszystkie opcje są jasno opisane w dołączonej dokumentacji w formie Amiga Guide.

Warto zwrócić uwagę, że autor programu zastosował domyślne wartości podstawowych argumentów, takich jak DEVICE, UNIT i ATTEMPT. Są one domyślnie ustawione odpowiednio na: urządzenie serial.device, wartość 0 oraz argument NO. Oznacza to oczywiście, że sterowanie będzie przeprowadzane za pomocą wbudowanego portu szeregowego. Aby skorzystać z tych opcji w ramach ikony na Workbenchu wystarczy, że każdy parametr będzie miał zwykłą postać, czyli OPCJA=wartość.

Przykładowo:

DEVICE=uart.device, UNIT=1

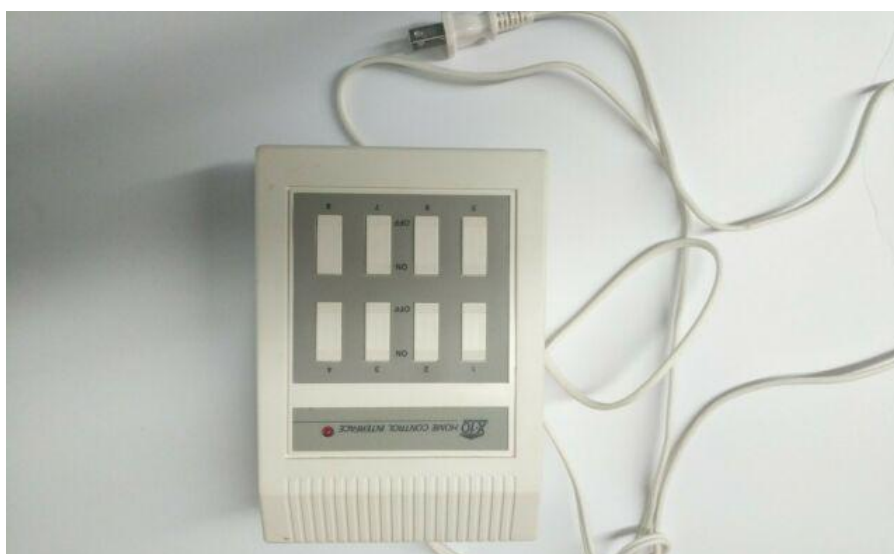
Powiedzmy pokrótce jakie argumenty mamy do dyspozycji. Wszystkie polecenia są wydawane w trybie bezpośrednim, co oznacza, że są one wysyłane natychmiast. Pierwszą ważną opcją to ON, która pozwala na włączanie wielu urządzeń. Jej przeciwieństwem jest OFF, która wyłącza urządzenia. Opcja DIM pozwala na ustawienie poziomu jasności oświetlenia. Wartość może tutaj zawierać się w przedziale od 1 (najjaśniejsza) do 16 (najciemniejsza).

Kolejną sprawą to obiekty docelowe, które chcemy obsługiwać. Określamy je za pomocą opcji o nazwie TARGETS. Obiekty mają formę jednoznakowego kodu, po którym natychmiast następuje 1- lub 2-cyfrowy numer jednostki. Na przykład, A1 i P16, natomiast wpis "A 1" nie zadziała, ponieważ zawiera spację.

Funkcja BASEHC umożliwia ustawienie podstawowego kodu sterownika CP290. Przyjmuje on argument jednoznakowy. Należy pamiętać, że ustawienie bazowego kodu spowoduje skasowanie dowolnego programu w pamięci

interfejsu CP290, dlatego najpierw należy przestać zdarzenia do pliku, a dopiero później używać tej opcji. Kod podstawowy, które urządzenia są dostępne za pomocą przycisków na klawiaturze CP290. Słowo SETCLOCK ustawia natomiast zegar wewnętrzny interfejsu w taki sposób, aby był synchronizowany z zegarem systemowym Amigi.

Funkcja DIAG pozwala przeprowadzić test diagnostyczny sterownika CP290. Wymaga to dłuższej chwili. Tak jak poprzednio, należy pamiętać, że uruchomienie diagnostyki spowoduje skasowanie dowolnego programu w pamięci CP290. Jeśli urządzenie przejdzie pomyślnie test, program CyberX10 wyświetli ustawienia aktualnego kodu bazowego i zegara wewnętrznego sterownika. Jeśli CP290 nie przejdzie testu diagnostycznego, zostanie wykonana jedna z dwóch czynności. Wystąpi błąd Timeout (reakcja urządzenia trwała zbyt długo i nie uzyskano odpowiedzi), albo pojawi się błąd Status Flag Error (uzyskano odpowiedź w planowanym czasie, ale została przekazana informacja o błędnym działaniu).



Interfejs CP-290 to małe, proste pudełko o wielu ciekawych możliwościach.



Idźmy dalej. Opcja DOWNLOAD powoduje odczytanie zdarzeń czasowych i danych graficznych z określonego pliku i przesłanie ich do pamięci sterownika CP290. Wykonanie tej funkcji może zająć nawet ok. 4-5 minut. Jak tworzyć pliki opisuje szczegółowo dokumentacja CyberX10, pod odnośnikiem "Programming". Ogólnie rzecz biorąc należy stworzyć zwykły plik tekstowy. Każdy jego wiersz będzie traktowany jako pojedyncze zdarzenie czasowe lub element danych graficznych. Linie zaczynające się od znaku "#" są traktowane jako komentarze.

Można też wstawiać puste linie. Każda linia musi zaczynać się od słowa "timer" lub "graphics". W ten sposób przekazujemy informację, z jakim rodzajem zdarzenia mamy do czynienia i interfejs CP290 będzie wiedział jak je przetworzyć. Przykładowo, po słowie kluczowym "graphics" powinny znaleźć się dwie liczby, obie w zakresie od 0 do 254 (ale nie 255), czyli:

graphics 1 2 graphics 3 15

Więcej informacji znajdziemy w pliku "CyberX10.guide" dołączonym do archiwum z programem. Pobranie pliku z komputera całkowicie zastępuje wszelkie inne programy, które mogą znajdować się w pamięci modułu CP290. W związku z tym, jeśli chcemy zmodyfikować, usunąć lub dodać zdarzenie, trzeba załadować do pliku aktualne oprogramowanie sterownika CP290, edytować je, a dopiero później wysłać plik do urządzenia. Z kolei funkcja UPLOAD spowoduje przesłanie zdarzeń czasowych i danych graficznych z modułu

CP290 i zapisanie ich w pliku na dysku komputera. Plik ten można następnie

Warto dodać także kilka słów na temat słowa ATTEMPT. Bez jego użycia interfejs CP290 będzie na stałe przypisany do transmisji szeregowej. Zazwyczaj jest to pożądane, ponieważ pozwala na wysyłanie wielu poleceń w tym samym czasie i żadne nie zostanie pominięte. Jednak w pewnych sytuacjach może nastąpić przerwa w przesyłaniu danych i interfejs CP290 zostanie zablokowany, bo wciąż będzie oczekiwał na odpowiedź ze strony interfejsu. Po wykorzystaniu argumentu ATTEMPT (ustawiamy go na wartość YES), program zadziała bez "zamrożenia". Będzie czekał do pięciu sekund, po których interfejs CP290 powinien odpowiedzieć. Jeśli się to nie stanie, program przerwie próby wysyłania informacji i polecenia po prostu nie zostaną wykonane. To nie wszystkie opcje programu, jednak podstawowe, pozwalające na obustronną komunikację z CP290. Oczywiście jego dokumentacja mówi wyłącznie o obsłudze systemu Windows, a dłuższy opis możliwości oraz dokonanych zmian można znaleźć pod poniższym adresem:

<http://www.hifi-remote.com/huras/x10wc.htm>

Więcej informacji na temat poleceń standardu X10 można też znaleźć tutaj (w języku polskim):

[https://pl.qwe.wiki/wiki/X10_\(industry_standard\)#List_of_X10_commands](https://pl.qwe.wiki/wiki/X10_(industry_standard)#List_of_X10_commands)

Jak oceniać tego typu oprogramowanie? CyberX10 może być zaskoczeniem, nawet dla długoletnich użytkowników Amigi.

W moim mieście, czyli Tomaszowie Kujawskim, mało kto interesował się domotyką i nawet teraz nie jest to temat do dyskusji. Tym bardziej cieszy mnie, że Amiga może być wykorzystana w celu obsługi automatyki domowej. Program jest prosty w użyciu, chociaż wymaga przeczytania instrukcji z uwagą. Na pewno nie możemy go po prostu uruchomić i używać - najpierw trzeba poświęcić chwilę na zapoznanie się z podstawowymi informacjami. Jeśli jednak znamy interfejs CP290 oraz sposób jego programowania, nie zajmie to zbyt długo.

Zdaję sobie sprawę, że mój artykuł będzie traktowany głównie jako ciekawostka lub archiwum historyczne. Jednak chcę pokazać, że Amiga od samego początku była traktowana jako komputer użytkowy. Dowodem są podobne programy, a Aminet zawiera więcej oprogramowania obsługującego standard X10, na przykład "3DHome.lha" (katalog "misc/misc"). Otwiera on własny ekran i pokazuje pomieszczenie z 4 lampami i oświetleniem. Po kliknięciu na którekolwiek z urządzeń, włącza je lub wyłącza, pokazując to, jak również zmienia stan urządzeń w rzeczywistym świecie za pomocą protokołu X10.

Można to nawet połączyć z programem VoiceShell (katalog "util/misc"), który umożliwia sterowanie programami po prostu mówiąc, zamiast wydając polecenia z klawiatury. Możliwość jest naprawdę dużo i możemy z nich korzystać także dzisiaj. Amiga, czyli w ogóle temat retro, nie musi kojarzyć się tylko z gramami.

Nowe narzędzia sieciowe

Mówiąc o oprogramowaniu związanym z siecią zwykle mamy na myśli duże pozycje jak przeglądarka internetowa czy serwer FTP. Są to rzeczy bardzo potrzebne, ale raczej zarezerwowane dla systemów NG. Amiga klasyczna nie ma tak dużej pamięci i szybkości, aby obsługiwać filmy w rozdzielczości FullHD na YouTube czy też szyfrować dane z dużą szybkością podczas transmisji do "chmury". Nie znaczy to jednak, że nasz komputer jest bezużyteczny. Wbrew pozorom wciąż pojawia się sporo programów do obsługi sieci, które działają na procesorach serii 68k oraz Workbenchu 3.0 lub 3.1.

Gdzie szukać tych nowych programów? Oczywiście, między innymi, na niezastąpionym Aminecie. Ostatnio znalazłem tam kilka bardzo ciekawych rzeczy i warto je przybliżyć naszym Czytelnikom. Na pierwszy ogień idzie program do testowania szybkości połączenia o nazwie "AmiSpeedTest". Działa nawet na Kickstarcie 2.0, ale - jak można łatwo sprawdzić - będzie działał również w systemach MorphOS i AmigaOS 4. Jest to program bardzo świeży, bo pojawił się na Aminecie w połowie marca tego roku.

Jak píše autor AmiSpeedTest, program działa na zasadzie analizy listy pobliskich serwerów, w stosunku do naszej lokalizacji. Dla każdego z nich otwierane są połączenia TCP w celu określenia serwera o najmniejszym opóźnieniu. Testy pobierania i wysyłania są kierowane do tego serwera, natomiast domyślne ustawienia to Speedtest.net oraz Fast.com.

Program może również działać jako klient lub serwer LAN, dzięki czemu testy prędkości mogą być przeprowadzane w sieci lokalnej. Rozmiar bufora dla danych może być ustawiony na 8, 16 lub 32 KB. W zależności od testu, ta wielkość może mieć wpływ na wydajność. Wyniki pokazywane są domyślnie w kilobitach na sekundę, ale można to zmienić, przelączając ustawienia na kilobajty na sekundę. Rezultat jest więc łatwy do zrozumienia i analizy.

Kolejny program o jakim warto powiedzieć to nowa wersja sterownika Prism do bezprzewodowych kart sieciowych. Archiwum ma nazwę "prism2v2.lha" i pojawiło się w styczniu. Urządzenie systemowe to "prism2.device" i obsługuje chipset Prism II i jego warianty - stąd nazwa. Obsługuje karty PCMCIA i CompactFlash w A1200/A600 oraz karty PCI we wszystkich systemach AmigaOS 4 oraz w komputerach PC z działającym Amithlonem.

Nowa wersja obsługuje szyfrowanie WPA i WPA2, w tym oba tryby szyfrowania TKIP i CCMP/AES, co wcześniej było problematyczne. Jednak nie wszystkie karty sieciowe obsługują wszystkie tryby szyfrowania, to trzeba już sprawdzać indywidualnie. Z mojego doświadczenia szyfrowanie działa znośnie na procesorze 68030 z zegarem 40 MHz lub szybszym.

Inne zmiany w nowej wersji Prism o numerze 2.7 to rozpoznawanie wielu nowych kart PCMCIA jak również poprawiono obsługę 64-bitowego szyfrowania typu WEP. Dodano obsługę starszych wersji firmware dla kart Intersil, a do archiwum dołączono kompletny kod źródłowy sterownika. Ta ostatnia wiadomość nie będzie przydatna każdemu Amigowcowi, ale każdy teraz może dokładnie sprawdzić, jak działa "prism2.device" i ewentualnie poprawić sterownik, jeśli tylko potrafi programować.

W 2019 roku na Aminecie pojawił się też port Netio - programu służącego do testowania szybkości transmisji danych, który dostępny jest dla systemów takich jak MS-DOS, Windows czy Linux. Netio był udostępniony już w 2012 roku, ale wymagał biblioteki "ixemul.library", poza tym jest bardziej zoptymalizowana, zajmuje mniej miejsca w pamięci i posiada wiele oryginalnych funkcji, które wcześniej nie były dostępne w wersji na Amigę.

Nowa wersja obsługuje testowanie szybkości sieci zarówno jako klient i serwer. Można więc jej użyć na dowolnych dwóch komputerach, na których działać mogą różne systemy operacyjne. Jako ciekawostkę



podam, że Netio jest dostępny nawet w wersji dla OS/2 Warp, czyli systemu firmy IBM, który został wyparty później na rzecz Windows. Program wymaga procesora 68020 lub lepszego, choć archiwum zawiera plik "netio68000" działający nawet na Amidze 500. Zajmuje tylko ok. 30 kilobajtów.

W katalogu "comm/tcp" znajdziemy plik o nazwie "netinfo.lha". Jest to pakiet pozwalający wyświetlać wiele informacji o naszej sieci. Integruje on funkcjonalność dobrze znanych, samodzielnych programów takich jak TRACEROUTE, PING, WHOIS i FINGER w jedną spójną całość z wygodnym interfejsem użytkownika.

Wśród wielu ciekawych możliwości warto wymienić opcję wysyłania serii pakietów danych w szybkim tempie, graficzne wyświetlanie czasów odpowiedzi dla funkcji PING, wyświetlanie graficzne czasów uzyskanych poprzez TRACED (minimalnych, maksymalnych oraz bieżących), jak również rozpoznawanie graficznego położenia serwerów. Zintegrowano inteligencją funkcję WHOIS, która pokazuje szczegółowe informacje dla obiektów. Program posiada też port ARexxa, co nie jest typowe dla tego typu pozycji.

Ponadto, zaciekało mnie archiwum "AmiNBD-0.1.lha", znajdujące się w tym samym katalogu "comm/tcp". Za jego pomocą możemy obsługiwać na Amidze serwery typu Network Block Device, które pozwalają uzyskać zdalny dostęp do pamięci masowej innego komputera, czyli zwykłego dysku twardego. Dzięki temu możemy wykonać kopię dysku za pośrednictwem sieci.

Technicznie usługa realizowana jest przez trzy elementy: część serwerową, część kliencką i sieć pomiędzy nimi. Na komputerze kliencie (ang. client), na której znajduje się punkt sieciowy, sterownik steruje urządzeniem. Za każdym razem, kiedy program próbuje uzyskać dostęp do dysku, sterownik przekazuje to żądanie do serwera, gdzie fizycznie znajdują się dane.

Jak pisze autor wersji na Amigę, AmiNBD nie jest wersją ukończoną, ale działającą. Wymaga dysku SCSI, ponieważ używa poleceń tego typu kontrolera, co niestety od razu zmniejsza ilość potencjalnie zainteresowanych Amigowców. Nie sposób jednak pominąć ten program, bo jest bardzo ciekawy i praktycznie niespotykany na naszym systemie.

AmiNBD obsługuje zdalnie tylko system Linuks i można go użyć zarówno do wykonania kopii naszego dysku, jak również przywrócenia danych, a ściślej mówiąc całego obrazu dysku. Mamy tu jednak sporo ograniczeń, bowiem niewidoczny jest postęp pracy, jak również program nie sprawdza ograniczeń sprzętowych Amigi w zakresie obsługi maksymalnego rozmiaru dysku. Trzeba tego pilnować samodzielnie.

Do odczytywania lub zapisywania danych można użyć partycji amigowej, ponieważ można ją zamontować poprzez sieć. Jednak jest to niebezpieczne, bo nasz ulubiony system nie będzie miał szansy na sprawdzenie zmian na dysku. W związku z tym nie pojawiają się żadne komunikaty o ewentualnych błędach i bardzo łatwo możemy uszkodzić system

plikowy. Dlatego najlepszym rozwiązaniem jest używanie dysku niezamontowanego, przynajmniej jeżeli mówimy o zapisywaniu danych.

W załączonej dokumentacji autor dość dokładnie wyjaśnia sposób użycia AmiNBD wraz z przykładowymi komunikatami uzyskiwanymi poprzez sieć. W związku z tym raczej nikt nie powinien mieć problemów z obsługą. Trzeba tylko pamiętać, że jest to bardzo wczesna wersja beta, która nie musi działać poprawnie w każdej sytuacji i na każdej konfiguracji Amigi. Z pewnością jest to jednak jeden z najciekawszych programów do obsługi sieci, jakie pojawiły się niedawno dla klasycznego Workbencha.

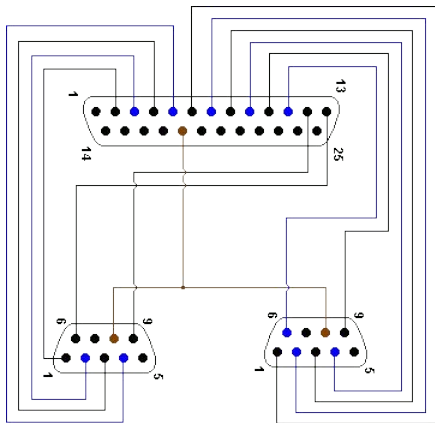
Powyższe programy to tylko mój wąski wybór archiwów, które można znaleźć na Aminecie. Nie musimy się do niego ograniczać, ale już ten krótki przegląd pokazuje, że nawet na Amigę klasyczną wciąż powstaje ciekawe oprogramowanie obsługujące różne usługi sieciowe. Nie są to może projekty na miarę nowego Firefoxa, ale nie zapominajmy o ograniczeniach pamięciowych czy wydajnościowych.

Bardzo dobrze bawię się na mojej Amidze 1200 z procesorem 68030 i życzę Wam tego samego. A wielu przypadkach ta zabawa przekształca się w dłuższą pracę, dzięki której ciągle dowiaduję się czegoś nowego o komputerach. Dodatkowo, podłączenie Amigi do sieci to rzecz niesłychana dla wielu osób spoza naszego środowiska, tak więc można zaskoczyć i zaciekać znajomych, na przykład, z pracy.

Gry dla czterech graczy

Na początek powiedzmy, w jaki sposób można grać w cztery osoby. Otóż, trzeba przygotować specjalny adapter podłączany do portu równoległego (Parallel). Joystick numer 3 i 4 będzie do niego podłączony, co pozwoli bawić się dodatkowym graczom. Z technicznego punktu widzenia, wspomniany adapter to tylko odpowiednie połączenie przewodów, nie trzeba lutować żadnych płytek elektronicznych ani używać układów programowalnych. Sposób połączenia przewodów przedstawiam w ramce. Wszystko można wykonać łatwo i szybko, nawet jeśli ktoś nie ma specjalnych zdolności w posługiwaniu się lutownicą.

Gdy już wszystko będzie gotowe, zastanówmy się jakie gry warto uruchomić? Decyzja nie będzie prosta, bo do wyboru mamy przynajmniej ok. 100 gier, w tym wiele zupełnie darmowych. Od razu mała uwaga: Nie oznacza to, że będą to same "krapy" - po prostu niektórzy autorzy dodają do kolekcji Aminetu swoje zbiory, które dawniej publikowali w kategorii Shareware. Niektóre gry powstały też po roku 2000, a wtedy mało kto zastanawiał się, czy na Amidze można jeszcze cokolwiek zarobić. Wiadomo było,



Schemat połączenia adaptera 4 joysticków. Korzystamy tutaj z portu równoległego Amigi (Parallel Port).

Chyba każdy z nas pamięta zabawę w gry takie jak Mortal Kombat czy Moonstone. Można w nie grać na dwie osoby, co przynosi niezapomniane wrażenia. Takie tytuły katowałem dawniej niemalże "dniami i nocami", gdy tylko znalazł się kolega dzielący moją pasję. Było to czas, gdy poznawałem bardzo dużą ilość gier, ale nie myślałem o tym, aby grać w większym gronie. Wyjątkiem były gry typu Pinball, ale tam gracze bawią się po kolei, a nie jest to walka bezpośrednia. Okazuje się jednak, że na Amigę istnieją gry pozwalające grać w grupie od 4 to nawet 12 osób. Na dodatek, wiele z tych tytułów można znaleźć na Aminecie i pobrać je całkowicie za darmo. Zaciekało mnie to nie na żarty.

że programy piszemy dla zabawy lub własnej satysfakcji.

Przejdźmy do tego, co najważniejsze, czyli do konkretnych tytułów. Wybrałem kilka, które wydały mi się najciekawsze lub zapewniły mi zabawę na wiele jesiennych wieczorów.

Parallel	JOY3	Parallel	JOY4
2	1	6	1
3	2	7	2
4	3	8	3
5	4	9	4
13	6	11	6
18	8	18	8
12	9	10	9

BASE JUMPERS

Jest to gra wydawana komercyjnie w 1995 roku. Znam ją z tamtego okresu i choć grałem głównie sam, to od razu zauważyłem, że można w nią grać maksymalnie na cztery osoby. Skąd taki wniosek? Jest to platformówka ze scrollingiem pionowym, zresztą bardzo dynamiczna i ze świetną grafiką. Jednak co drugi poziom jest nietypowy, bowiem zamiast "wspinać" się na górę labiryntu, który okazuje się wnętrzem budynku, skaczemy (oczywiście nielegalnie) na spadochronach w grupie czterech postaci.

Jedną z nich sterujemy joystickiem. Lecimy w dół nie widząc przeszkód, które pojawiają się na tyle późno,



Base Jumpers

trzeba mieć niezły refleks, aby nie wpaść na zbliżające się balustrady. Nic nie stoi na przeszkodzie, aby zepchnąć innego gracza w bok, ale trzeba pamiętać, że kolega może próbować się nam zrewanżować. Oczywiście cała sztuka polega na tym, aby nie zginąć. Jednocześnie musimy otworzyć spadochron na tyle późno, aby uzyskać dużą ilość punktów, ale też na tyle wcześnie, aby nie roztrzaskać się o podłoże.

W poziomach platformowych mamy dość skomplikowany labirynt, system zapadek, blokad i drzwi. Ekran przewijany jest do góry i musimy zdążyć przechodzić przez kolejne etapy tak, aby nasz bohater nie wypadł poza ramę ekranu - wtedy traci życie. W grze zbieramy też litery i inne przedmioty przynoszące punkty lub różne bonusy. Gracza ścigają policjanci, którzy nie chcą doprowadzić do nielegalnego skoku z budynku. Brzmi ciekawie? Bardzo polecam spróbować.

ale z wieloma zmianami. Po uruchomieniu od razu dostępne jest menu, w którym możemy wybrać liczbę graczy (od 1 do 4) oraz tryb rozgrywki. Może to być bitwa, czyli zwyczajna plansza na jedną lub dwie osoby, ale możemy też wybrać grę "po kablu". Trzeba wtedy połączyć dwie Amigi za pomocą przewodu null-model, dzięki czemu można się bawić nawet w osiem osób.

Niektórzy twierdzą, że w ten sposób można prowadzić rozgrywkę nawet w 16 osób jednocześnie, ale nigdy czegoś takiego nie widziałem. Ciekawym trybem są również



Blitz Bombers

BLITZ BOMBERS

Kolejna gra została wydana w 1996 roku i można ją dzisiaj znaleźć na Aminecie. Obsługiwany jest tryb dla nawet 8 graczy, więc musi być to coś ciekawego. W zasadzie, jest to kolejny klon słynnego Dyna Blaster,

mistrzostwa, które polegają na tym, że gramy większą ilość rund i wygrywa zwycięzca większości z nich. W menu można skonfigurować, czy chcemy korzystać z ulepszeń, czy zachować bardziej "klasyczną" rozgrywkę.

Wybieramy też mapę, których jest ok. 3 tysięcy (sic!), a następnie rodzaj uzbrojenia naszego bohatera. Może on dysponować różną liczbę bomb o różnej mocy oraz używać różnej ilości jednocześnie. W ustawieniach zmieniać można nawet czas przeznaczony dla jednej planszy oraz korzystać z opcji zapisywania i wczytywania stanu rozgrywki.

Na każdym poziomie mamy za zadanie dotrzeć do wyjścia, zabić potwory, zebrać klucze i wiele innych przedmiotów. Grafika jest bardzo dobra, chociaż bohater jest mniejszy niż w Dyna Blaster. Za to całość jest bardziej kolorowa (gra wymaga układów AGA - przyp. red.), mamy różne tajne tunele czy teleporty. Przedmioty jakie odnajdujemy niekoniecznie działają pozytywnie, bo można znaleźć też szkodliwe.

Przypomnę, że na wykonanie zadania mamy określony czas, ale jest go na tyle dużo, że w praktyce nie ma to większego znaczenia. Po ukończeniu każdego poziomu uzyskujemy hasło, więc nie trzeba od nowa "męczyć" tych samych planszy. Niektóre poziomy są bardziej rozbudowane i mają ukryte przejścia powodujące, na przykład, przełączenie trybu graficznego na Hires, czyli rozdzielczość 640x256 pikseli. Wydaje mi się to na tyle ciekawe, że warto poświęcić trochę czasu na Blitz Bombers.

Z drugiej strony, gra ma też kilka wad. Widać, że zabrakło porządnego przetestowania całości, bo zdarza się, że wpisanie hasła nie przenosi do właściwego poziomu. Jeśli trafimy na taki problem, przejście wszystkich poziomów jest praktycznie

niewykonalne. Następnym problemem to zawodność bomb. Po prostu niektóre w ogóle nie wybuchają, po prostu leżą w labiryncie. Być może jest to celowe urozmaicenie rozgrywki, ale potrafi być denerwujące.

Jeśli więc interesuje Was ładniejszy klon Dyna Blaster, z ulepszoną grafiką i mnóstwem nowych pomysłów, bonusów, potworów i nowych poziomów - zdecydowanie warto zagrać. Zwróćcie tylko uwagę, że na Aminecie dostępna jest zarówno wersja demonstracyjna, jak i pełna, która zawiera poprawki.

GAUNTLET 2

Gra pod tytułem Gauntlet była prawdziwym objawieniem w 1985 roku. Można było zanurzyć się w rozbudowany świat labiryntów i potworów, choć prawdę mówiąc nigdy nie byłem w stanie docenić tej ascetycznej grafiki oraz dziwnego widoku. Dopiero później zrozumiałem na czym polega

fenomen tej gry.

Po kilku latach jej autorzy postanowili zebrać nowe pomysły i stworzyć kontynuację. Wygląda podobnie jak oryginał i oferuje podobny poziom rozgrywki. Mamy sporo nowych elementów takich jak ruchome ściany, tajne przejścia czy fałszywe wyjścia. Wprowadzono także nowego wroga, który wygląda jak wielka ognista kula. Jeśli nasz bohater zostanie nią trafiony, będzie następnie przyciągał do siebie wszystkie inne potwory. Ma to znaczenie głównie podczas gry wieloosobowej, bo normalnie i tak wszyscy wrogowie są skupieni na jednym graczu.

Ciekawy jest też potwór Green Acid, do którego nie można strzelać, a magia wywołuje tylko krótkie oślepienie. Ma on bardzo nieregularne ruchy i choć nie ściga gracza to może narobić niezłych szkód teoretycznie przypadkowo.

Gauntlet 2 jest przykładem gry, w którą bardzo trudno grać samotnie.



Gauntlet 2



Dużo łatwiej zagrać przynamniej w dwie osoby, inaczej można się zniechęcić. Gdy jedna z postaci umiera, druga może korzystać z zasobów pierwszej, więc jest to niezły sposób na przedłużenie rozgrywki. Z pewnością jest to jeden z najlepszych przykładów gry zespołowej, która wymaga naprawdę ścisłej współpracy. Oczywiście jeśli zależy nam na przejściu gry, a nie tylko niedzielnej zabawie. Gracze mogą mieć nawet te same postacie, będą one wtedy różnić się kolorem. To odróżnia drugą część od pierwszej, gdzie każdy gracz musiał być inny.

HIRED GUNS

Ostatni tytuł jest bardzo znany i zastanawiałem się, czy go w ogóle opisywać. Jednak pamiętam, że w latach '90-tych Hired Guns zrobiło na mnie kolosalne wrażenie. Czym? Jakością grafiki, klimatem i rozmachem. Była to jedna z nielicznych produkcji RPG na Amigę 500 jakie miałem, pracująca częściowo w rozdzielczości 640x512 pikseli. Pojawiające się płynnie komunikaty wręcz elektryzowały, a ciężka atmosfera sprawiała, że chciało się zobaczyć więcej i więcej.

Jest to produkcja dla maksymalnie czterech graczy, ale wszyscy grają na jednym komputerze. Ekran jest podzielony na trzy lub cztery niezależne okna, po jednym dla każdej postaci. W trybie dla czterech graczy, dwie osoby sterują grą za pomocą dwóch myszek, jedna za pomocą joysticka, a jedna za pomocą klawiatury. Moim zdaniem najwygodniejszy jest tryb dla dwóch graczy, w którym każdy kontroluje dwóch bohaterów.



Gra legenda: Hired Guns. Oprócz bardzo ciekawego trybu rozgrywki posiada wspaniałą opracę audio-wizualną. Możemy zobaczyć między innymi grafikę w trybie 640x512 pikseli, na zwykłej Amidze 500. Przy większej ilości pamięci aktywują się dodatkowe efekty dźwiękowe.



W grze Base Jumpers możemy ścigać się nawet do czterech graczy jednocześnie. To może być świetna zabawa na imprezie retro.

Na początku do wyboru mamy 3 lub 4 postacie (wszystkich jest 12), które różnią się głównie siłą i wyposażeniem. Sterowanie jest bardzo podobne do każdego innego RPG-a, ale zamiast zwykle spotykanych mieczy, łuków i zaklęć mamy pistolety, karabiny maszynowe, lasery, granaty i inną nowoczesną broń.

Naszym zadaniem jest przejść prawie 20 dużych poziomów, pełnych zwoźniczych potworów, rozwiązać dziesiątki labiryntów, znaleźć jednostki nuklearne, używać ogromnej ilości zagubionych kluczy i kart magnetycznych. Oczywiście przy okazji zabijamy setki, jeśli nie tysiące wrogów. Oprócz broni można używać specjalnych urządzeń, zwanych wzmacniaczami psionicznymi (ang. psionic amplifiers), które wywołują różne ciekawe efekty. Pozwalają unikać zatruc, teleportować się, oddychać pod wodą albo zmieniać kształt i charakter przedmiotów.

Powodzenie misji w dużej mierze zależy od właściwego wykorzystania konkretnych elementów jakie odnajdujemy. Labirynty są pełne skrzynek zawierających broń,

amunicję, żywność i wiele innych przedmiotów, które można zbierać i używać w odpowiednich momentach. Dostępne są nawet misje szkoleniowe i krótkie gry zręcznościowe, dzięki którym można sprawdzić swoje umiejętności.

Gra jest bardzo wciągająca. Atmosfera jest mroczna, co wzmacniają świetne efekty dźwiękowe. Niektóre potwory są naprawdę straszne, szczególnie jeśli zaczynają nas gonić akurat wtedy, gdy kończy się amunicja. Nie jest to łatwe. Wydaje mi się, że główną siłą Hired Guns jest właśnie jej atmosfera, bowiem zachęca ona do zabawy, a później przez wiele godzin nie daje odejść od komputera.

Poziomy są bardzo dobrze zaprojektowane i wymagają współpracy między członkami drużyny. Najlepiej posiadać Amigę z pamięcią przynajmniej 2 MB, bo aktywują się wtedy dodatkowe efekty audio. Całość zajmuje pięć dyskietek i można ją zainstalować na twardym dysku bez pakietu WHDLoad, który osobiście średnio lubię. Jedyne, co mi się nie podoba to fakt, że podczas gry z dyskietek

trzeba je często przekładać i to niekoniecznie po kolei. Dzisiaj nie ma to jednak specjalnego znaczenia. Nie wiem czy mój wybór gier przypadł Wam do gustu. W ramach prezentuję jeszcze kilka innych tytułów, które są warte uwagi. Na pewno rozgrywka z żywymi graczami to niezapomniane wrażenie. Chcę też zwrócić Waszą uwagę na usługę AmigaLive, która jest dostępna na poniższej stronie:

www.amigalive.com

Dzięki odpowiednio skonfigurowanego emulatorowi FS-UAE można grać z innymi osobami przez sieć - nawet, gdy gra oryginalnie tego nie przewiduje. W dobie ograniczeń związanych z koronawirusem jest to ważne, bo przecież w ten sposób można nawet zorganizować turniej na wirtualnej imprezie scenowej, co dzisiaj może mieć szczególne znaczenie dla wielu osób pozostających w domu. Minęły czasy, gdy trzeba było przynosić drugi komputer do kolegi i łączyć go za pomocą przewodu null-modem. Moim zdaniem nic nie zastąpi prawdziwego kontaktu "twarzą w twarz", ale AmigaLive pozwala na zabawę z osobami na drugim końcu świata. W tym sensie emulacja ma większy sens, niż trwanie przy prawdziwym sprzęcie sprzed lat.

Pamiętać trzeba tylko, że AmigaLive ma też swoje wady, na przykład pojawiają się problemy z połączeniem lub synchronizacją gier na dwóch komputerach. Zwykle jednak da się to łatwo ominąć i cieszyć się ulubionymi grami. Polecam gry takie Chaos Engine lub Banshee - tu co prawda zagramy tylko w dwie osoby, ale ich przejście zajmie sporo czasu. A przecież to



Usługa Amiga Live to sposób na grę przez sieć w klasyczne amigowe gry. Nie trzeba przy tym modyfikować oryginalnych wersji gier - wszystko załatwia emulator.

A-Train w detalach

O tej grze pisaliśmy w poprzednim numerach Amigazynu, jednak mój artykuł przedstawia dużo więcej technicznych szczegółów. Mam nadzieję, że będzie to dla Was interesujące. A-Train jest grą o budowie imperium finansowego opartym na budownictwie kolejowym, planowaniu pociągów, obrocie gruntami, zarządzaniu nieruchomościami, planowaniu przestrzennym i handlu akcjami. Aby ująć to w odpowiedniej kategorii, najlepiej opisać tę grę jako skrzyżowanie Sim City z Railroad Tycoon.

Zacznijmy od uwag technicznych. Gra może być uruchomiona w trybie Hires Interlace (czyli 640x400 pikseli) lub standardowym Lores (320x200 pikseli), natomiast menu ma zawsze rozdzielczość Hires (640x200 pikseli). Jeśli dysponujemy monitorem VGA możemy mieć cztery razy większą powierzchnię roboczą, a całość działa wystarczająco szybko na procesorze 68030.

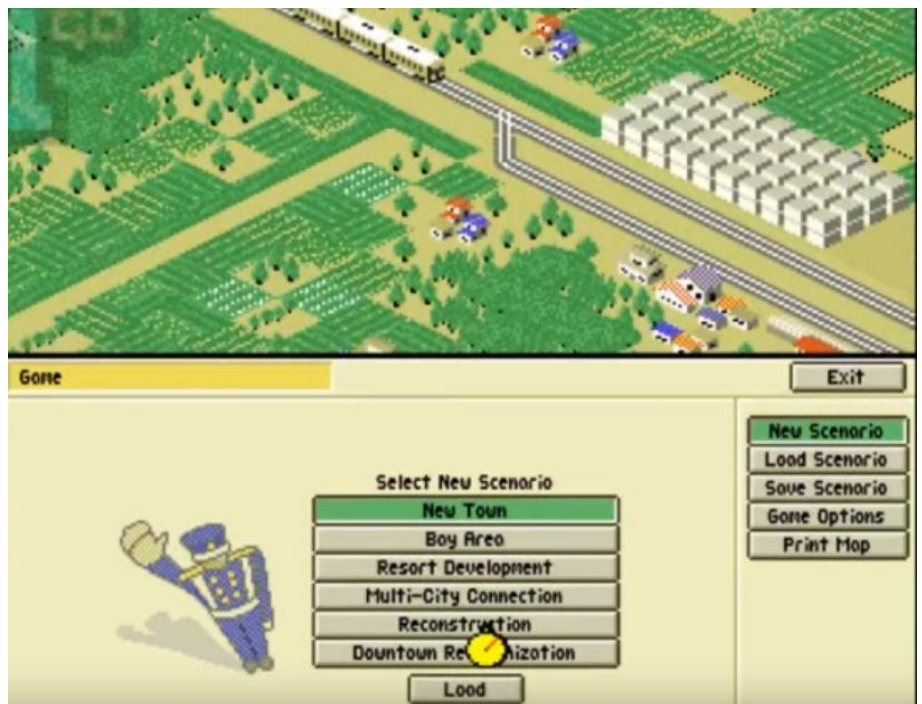
Pierwszą rzeczą, którą można zauważyć jest naprawdę świetna grafika. Wszystko jest przedstawione w ujęciu izometrycznym, z drobnymi szczegółami, takimi jak drzewa, a także wiele różnych rodzajów domów i gospodarstw. Występuje tu cykl nocny i dzienny (można go wyłączyć), który stopniowo powoduje przyciemnienie krajobrazu w miarę zbliżania się wieczoru.

Po zachodzie słońca, gdy wszystko zmieni się w rdzawą czerwień, zapalają się światła budynkach i pociągach. Szary wczesny świt zamienia się w jasne światło

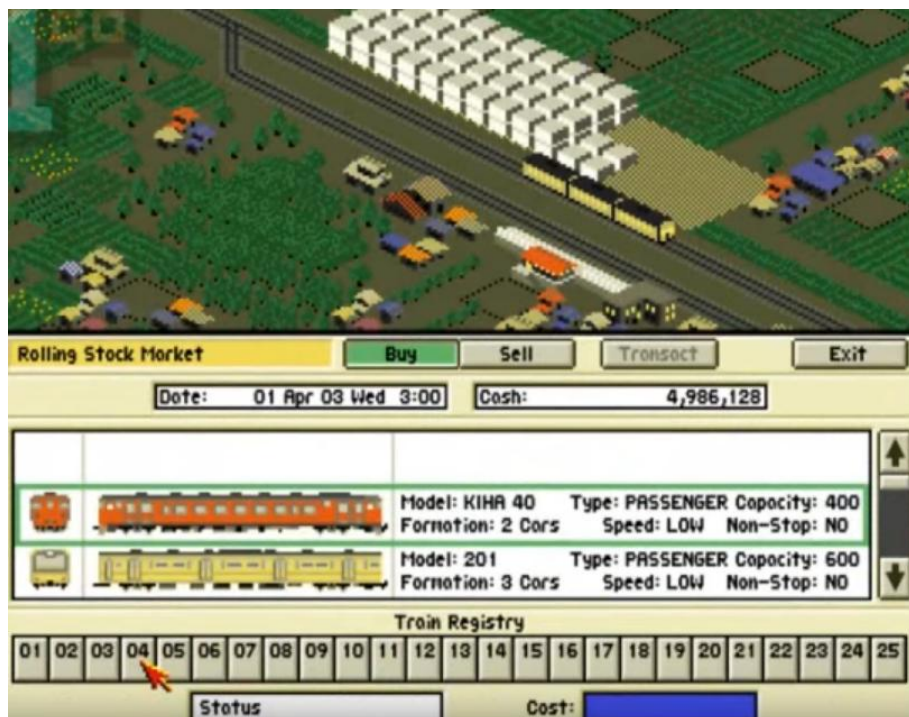
słoneczne dnia około 7 rano. Zimą pada śnieg, a od czasu do czasu pojawiają się zabawne drobiazgi, jak na przykład Święty Mikołaj przelatujący na saniach i renifery o północy 24 grudnia.

Są tu nawet obiekty UFO (choć sporadycznie), a w parkach rozrywki, w piątkowe wieczory w lecie, nie raz wybuchają fajerwerki. Wieżowce w budowie mają na szczycie dźwigi (czerwono-białe), a ulice są oświetlone latarniami ulicznymi. Budynki handlowe mają neony. Dbałość o szczegóły jest niesamowita.

A-Train to jednak o wiele więcej niż tylko dobry wygląd. W grze mamy sześć scenariuszy, które dotyczą konkretnych problemów z wykorzystaniem różnych kwot pieniędzy. Najbardziej otwartym jest "Nowe Miasto" (New Town), które przedstawia matką społecność z



Wybór jednego z przygotowanych scenariuszy rozgrywki.



Inspekcja linii kolejowych.

linią kolejową i już działającą stacją. Pociągi pasażerskie i towarowe (nad nimi gracz nie ma kontroli) przyjeżdżają i odjeżdżają 24 godziny na dobę. Z tych pociągów czerpiemy zyski.

Wyzwaniem jest zbudowanie linii dostarczającej pasażerów do tego miasta, co zasadniczo powoduje powstawanie innych nowych miast. Pozostała część mapy pokryta jest rozproszonymi gospodarstwami rolnymi, bez faktycznych skupisk ludności.

Na drugim końcu skali znajduje się scenariusz zatytułowany "Reorganizacja miasta" (Downtown Reorganization), w którym należy przekształcić słabo prosperującą miejską linię dojazdową w dużym mieście w bardziej dynamicznie działającą jednostkę. Pozostałe cztery scenariusze mieszczą się pomiędzy tymi dwoma przypadkami. Ciekawym pomysłem w tej grze jest fakt, że mamy do

czynienia z miejskimi liniami komunikacyjnymi, a nie z liniami głównymi. Jedno spojrzenie na tabor pokaże nam, że to wszystko są pociągi podmiejskie. Po drugie, jeśli jakaś linia biegnie poza mapą, zakłada się, że łączy się z dużym miastem lekko poza mapą. Pociągi kursujące z pominięciem mapy wracają z materiałami eksploatacyjnymi oraz pasażerami.

Towary wyglądają jak małe, białe pudełka, które pociągi przewożą i wyładowują obok naszej stacji, ale tylko w sytuacji, gdy posiadamy własny teren, na którym materiały mogą być przechowywane. Pusty pociąg towarowy zatrzymujący się na stacji zabierze towary, jeśli są w pobliżu i przeniesie je na następny przystanek.

Surowce są wykorzystywane do budowy budynków. W zależności od tego, co chcemy zrobić, potrzebujemy więcej lub mniej materiałów dostępnych lokalnie.

Domy również wymagają odpowiednich surowców. Jeśli nie są dostępne żadne materiały, zatrzymuje się rozwój, co może nas szybko zrujnować. Pociągi są jedynym sposobem na przewiezienie materiałów i pasażerów, a także dostarczenie ich w konkretne miejsca, gdzie są potrzebne.

To prowadzi nas do tematu samych pociągów. Istnieje pięć różnych pociągów towarowych, o różnych prędkościach i różnej pojemności (2 lub 4 pudeł z towarami). Mamy około 20 różnych pociągów pasażerskich, 2- lub 3-wagonowych, o dużej lub małej prędkości. Pociągi te mogą lub nie być w stanie przejechać przez stacje bez zatrzymywania się, co może mieć duże znaczenie. Ponadto kupno i przejazd tych pociągów kosztuje różne kwoty.

Każdy pociąg musi być zaplanowany. Na małej mapie pokazane są dostępne tory, gdzie decydujemy, jaką orientację mają mieć zwrotnice (punkty lub rozjazdy), gdy pociąg przez nie przejeżdża. Następnie można przeprowadzić jazdę próbną.

Należy również ustalić godziny odjazdów pociągów. Domyślnie pociągi kursują wahadłowo, zatrzymując się na każdej stacji na trasie. Jednak dla pociągów pasażerskich bardziej logiczne jest opuszczenie przedmieścia o 8:00 rano i powrót z węzła kolejowego w mieście o 6:00 wieczorem - tak, aby przewieźć wszystkich dojeżdżających pasażerów. Każda stacja może mieć 2 tory, a pociągi można ustawić tak, aby można było prowadzić dwa składy na jednym torze.



Po ułożeniu torów i zbudowaniu stacji, można zwrócić uwagę na budowanie bloków mieszkalnych, aby zachęcić do transportu osoby dojeżdżające do pracy. Później można budować domy (od 5 do 40 pięter), budynki handlowe, hotele, fabryki (produkują one materiały), parki rozrywki, pola golfowe, ośrodki narciarskie i tym podobne - aby zwiększyć zyski. Inni deweloperzy również budują takie budynki, a od czasu do czasu pojawiają się na rynku i można je kupić jako gotowe obiekty. Zapewniają one zatrudnienie, zakwaterowanie i stwarzają więcej możliwości dla naszych linii kolejowych.

Na rynku akcji dostępne są 24 różne akcje, a także bank, który udziela kredytów (do 10% wartości netto). Oba są jednak otwarte tylko w niektórych godzinach w tygodniu. Oprocentowanie pożyczek zmienia się (dostępne są 3 rodzaje

pożyczek), więc najlepiej jest zaciągać je przy niskim oprocentowaniu. Chociaż może się to wydawać zaskakującą różnorodnością opcji, wszystko jest łatwo dostępne za pomocą zestawu czytelnego menu.

Model ekonomiczny w tej grze jest bardzo złożony, mamy tu wiele składników, jak podatek dochodowy od osób prawnych, opodatkowanie zysków kapitałowych, zmienne ceny gruntów, ceny budowlane i ceny akcji, które rosną i spadają w zależności od tego, co robimy. Jeśli dużo budujemy, możemy spodziewać się wzrostu kursów akcji firm z branży stalowej i budowlanej.

W miarę pojawiania się coraz większej liczby budynków wynajmowanych, akcje spółek zarządzających rosną. W żadnym momencie nie wydaje się to jednak

zbyt trudne, ponieważ wiele z wymienionych rzeczy jest obsługiwanych automatycznie przez komputer.

Do tej pory nie spotkałem się z grą strategiczną na Amidze, która dawałaby takie poczucie autentyczności w sposobie działania. Bilanse są dostępne od razu, aby dać pogląd na to, co się dzieje - godzina po godzinie. Dostępny jest również wykres radarowy, który pokazuje, w jakich sektorach obserwuje się największy wzrost. Stymulowanie tego wzrostu bez bankructwa jest prawdziwym wyzwaniem w tej grze.

Podsumowując, jest to niezwykle skomplikowana i złożona gra. Jest tu wiele do zrobienia i nie jest łatwo znaleźć odpowiednią formułę, aby wszystko dobrze funkcjonowało. Nie oznacza to jednak, że nie można się przy tym świetnie bawić. Warto jeszcze raz podkreślić przejrzysty system menu, który jest wielką zaletą.

Oczywiście, gra ma też kilka wad. Nie można powiedzieć nic dobrego o dźwięku, który najlepiej od razu wyłączyć. Muzyka jest powtarzalna, a dźwięk pociągu brzmi bardziej jak karabin maszynowy. Poza tym nie widzę tutaj niedoskonałości, chyba że będą to wymagania. A-Train co prawda działa na pamięci 1 MB i nawet Kickstarcie 1.3, ale po pełni szczęścia przydaje się szybszy procesor. Podobnie, problematyczny może być tryb Interlace, na szczęście można go wyłączyć. Jest to z pewnością bardzo złożona, precyzyjna, ale i zabawna gra.



Gra pomaga przy ocenie aktualnej lokalizacji gracza lub planów rozmieszczenia jego linii kolejowej.

Future Bike

Gra wygląda podobnie jak Lotus czy Jagura, ale widocznych jest większa ilość wysokich obiektów, po obu stronach drogi. Animacja nie jest niestety tak płynna, co każe przypuszczać, że w wersji na Amigę dokonana bezpośredniego portu gry z Atari ST. Pod tym względem wersja na C64 jest bardziej dynamiczna, choć oczywiście mamy do czynienia z grafiką o niższej rozdzielczości. Nasi przeciwnicy wyglądają praktycznie tak samo jak my, odróżnia ich tylko kolor kasku. Widać, że autorzy używają tych samych lub tylko lekko zmienionych sprajtów, być może dla zaoszczędzenia pamięci. Podczas rozgrywki jeździmy w typowych miejscach typu miasto, pustynia czy góry lodowe, a przeszkody zmieniają się w zależności od tematyki, jednak brakuje tu większej różnorodności. Co jakiś czas możemy dokupić wyposażenie do naszego pojazdu, na przykład osłony, radar czy pociski eliminujące wrogów. To już wygląda zdecydowanie lepiej i daje więcej frajdy. Future Bike daleko jednak pod tym względem do takich tytułów jak Road Kill czy nawet Micro Machines.

Tak, tak - mam na myśli grywalność. Wyścig powinien wywoływać dreszcz emocji, co niestety nie ma miejsca. Może dlatego, że nie mamy hamulców, więc droga jest przewijana jednostajnie, a gracz musi tylko unikać przeszkód. Przypomina mi to słynną grę Stealth na C64, ale tam trzeba dojechać do celu w określonym czasie - tutaj tego nie ma. Bardzo łatwo też uderzyć w dowolną przeszkodę, co sprawia, że gra jest dużo trudniejsza i może zniechęcać. Radar, który możemy kupić w sklepie często jest bezużyteczny, bo potrafi pokazywać

Podczas przeglądania mniej znanych (dla mnie) gier, natrafiłem na pozycję pod tytułem "Future Bike". Nie jest to najlepsza gra, ale mimo to chciałbym ją przedstawić czytelnikom naszego magazynu. Jest to wyścigówka, w której jeździmy futurystycznym motocyklem. Na samym początku gracz może przeczytać krótką historię, w której mowa o tym, że światowe rządy zakazały używania motocykli, więc miliarder Finnius Houlder zaczął tworzyć legalny zamiennik. Niestety jest to tylko tekst, bez animacji czy choćby ilustracji.

nieprawdopodobnie cele. Nie wiem czy to celowy zabieg autorów, ale takie urozmaicenie do mnie nie przemawia. Strzelanie do innych zawodników szybko pokazuje, że mają oni worki z pieniędzmi, które zyskujemy. Można je oczywiście wydać w sklepie. Niezbyt dobrze zrealizowane jest także strzelanie. Ciężko trafić w cel, do tego nie mam w ogóle poczucia zwiększonego poziomu trudności czy zwiększania mocy wraz z tym, co kupuję w sklepie. Można mianowicie dokupić takie rzeczy jak "Extra Energy" czy "Power-Up", ale niewiele to daje. W pewnym momencie trafiamy jednak na trudniejszy wyścig, ale z kolei poziom trudności jest od razu zwiększony maksymalnie. Nie wypada to dobrze.

Dlaczego więc piszę o Future Bike? Z kilku powodów. Być może moje

wrażenie odnośnie słabej grywalności jest odosobnione i gra się komuś spodoba. Grafika nie jest wybitna, ale obiekty narysowane są schludnie, szkoda tylko, że nie wykorzystują możliwości palety barw Amigi. Poza tym gra wygląda jak ewolucja 8-bitowych wyścigówek przepisana na Atari ST, a następnie przeniesiona na Amigę 500 bez większych zmian. Jest to część ciekawej historii.

Nie jest to moja ulubiona gra, ale jak na rok wydania 1990, nie wypada najgorzej. Wydano ją również w wersji na ZX Spectrum i chyba na tej platformie jest bardziej dopracowana. Na Amidze można ją uruchomić głównie po to, aby porównać z wydaniem na Atari, albo sprawdzić swój refleks, bo przechodzenie przez dalsze poziomy nie jest łatwe.



Future Bike to typowy przykład portu gry z Atari ST na Amigę 500.

BECOME A SMUGGLER - SHOW THEM HOW TO FLY!



TM

VEPGE WORLD

ICARUS RISING



RETRO
BONES

JOIN THE REVOLT!!! visit www.RETROBONESGAMES.com
..... WE NEED YOU TO FIGHT THE TYRANNY!