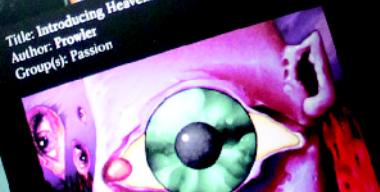




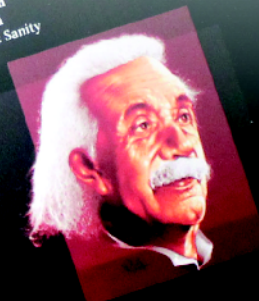
Polskie Pismo Amigowe

Numer 1/2014 (11)

cena 20 zł



Title: Clown
Author: Ra
Group(s): Sanity



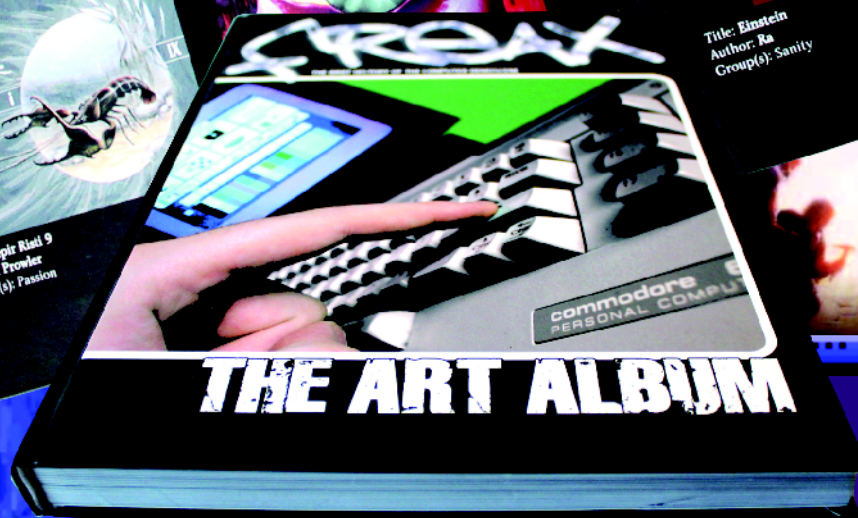
Title: Einstein
Author: Ra
Group(s): Sanity

Title: Eagle
Author: Ra
Group(s): Sanity



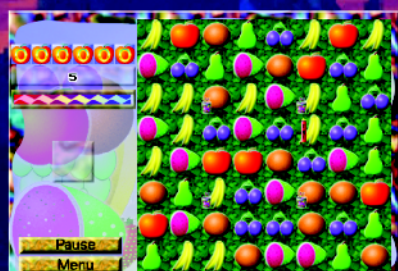
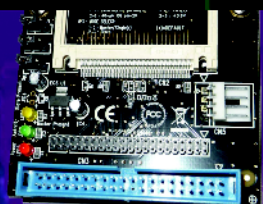
Title: Pepsi Top
Author: Ramjet
Group(s): Pure Metal Coders

Year: 1991



Recenzja gry THE DIG

POS	TRACK #1
28	D-12
29	D-12
30	D-12
31	D-12
32	D-12
33	D-12
34	D-12
35	D-12
36	D-12
37	D-12
38	D-12
39	D-12
40	D-12



	Retro Game 2013	HW	Pkt. (0-10)
1	Montezuma	Amiga	
2	Solomon's Key	Amiga	
3	Deadly Labyrinth	ZX	



Dobry Hosting w PPA.pl

Już za 100 PLN rocznie* możesz mieć:

Konto 1000 MB, 10 GB transferu miesięcznie, unikalny adres w domenie *.ppa.pl, 10 baz danych (MySQL i PostgreSQL), FTP, WWW, PHP, CGI, ColdFusion, Tomcat, CRON. Własne strony błędów, kopie bezpieczeństwa, pomoc techniczna. Do tego wiele nielimitowanych opcji: konta e-mail z ochroną antywirusową i antyspamem, z dostępem przez www, aliasy/przekierowania e-mail, parkowanie własnych domen, subdomeny. Duży wybór preinstalowanych aplikacji www (CMS-y, galerie, blogi, fora i inne), a wszystko jest zarządzane poprzez WWW za pomocą wygodnego panelu administracyjnego PLESK.

... a za dodatkowe 100 PLN Twoje konto zwiększy się do:

2000 MB pojemności
20 GB transferu miesięcznie
20 baz danych

A wszystko w domenie www.ppa.pl
- największego w Polsce portalu poświęconego Amidze i tematom z nią związanym.

W ofercie dostępne także inne warianty, również konta darmowe.

Sprawdź na www.hosting.ppa.pl

* możliwa płatność jednorazowa lub w ratach

APEL O ARTYKUŁY!

Drodzy Czytelnicy,

Czekamy na Wasze artykuły, które zasilają bazę artykułów do kolejnych numerów. Przed nadesłaniem artykułu prosimy o skonsultowanie tematu z redakcją. Artykuły prosimy nadsyłać w postaci plików tekstowych (ASCII) wraz z dołączonymi obrazkami lub zdjęciami (format PNG lub JPG). Propozycje oraz sugestie należy nadsyłać na adres podany w stopce redakcyjnej.

Konkurs na artykuł

Redakcja PPA ogłasza konkurs na napisanie artykułu do „Polskiego Pisma Amigowego”. Temat artykułu powinien pasować do profilu pisma i powinien być wstępnie uzgodniony z redakcją (np. e-mailowo na adres kontaktowy redakcja@ppa.pl). Artykuł zgłoszony do konkursu nie może być krótką notką, orientacyjne minimum to 10 000 znaków oraz ilustracje, co pozwoli wypełnić dwie strony pisma.

Redakcja zastrzega sobie prawo do wydrukowania każdego ze zgłoszonych artykułów w „Polskim Piśmie Amigowym”. Artykuł wybrany do druku nie może być nigdzie opublikowany ani przed, ani rok po wydaniu go drukiem w PPA.

W miarę możliwości będziemy starać się opublikowane artykuły nagradzać albo w drodze plebiscytu na najlepszy tekst numeru lub w postaci wierszówek wypłacanych za wypełnioną stronę pisma.

Oprócz tego, tradycyjnie, każdy autor artykułu zakwalifikowanego do druku otrzyma bezpłatny egzemplarz pisma.

Termin nadsyłania artykułów upływa:

15 marca 2014 roku.

Zapraszamy do udziału!



Polskie Pismo Amigowe

Redaktor naczelny: Sebastian Rosa.

Zespół redakcyjny: Konrad Czuba, Grzegorz Murdzek, Piotr Sadowski, Krzysztof Żegleń.

Korekta:

Skład: Sebastian Rosa

Grafika: Sławomir Woźniak

Kontakt: redakcja@ppa.pl, <http://ppa.pl>.

Skład pisma wykonywany jest w programie OpenOffice 3.

Druk: Drukarnia cyfrowa DCTF w Tarnowie.

Wydawca: Adam Zalepa – AIBB.pl



Polskie Pismo Amigowe jest wydawane w wolnym czasie członków redakcji i autorów. Redakcja nie gwarantuje regularnego ukazywania się kolejnych numerów. Cena jaką płacisz za pismo pokrywa jedynie koszty jego wydawania, pismo nie przynosi zysków.

Poglądy wyrażane w artykułach są poglądami ich autorów i niekoniecznie odpowiadają stanowisku redakcji.

Prawa autorskie artykułów należą do ich autorów. Przedruk i publikacja w formie elektronicznej wyłącznie za zgodą redakcji. Pojawiające się w artykułach słowne i graficzne znaki towarowe firm trzecich są użyte wyłącznie w celach informacyjnych i pozostają własnością tych firm. PPA nie jest gospodarczo związane z żadną z tych firm.

© Polski Portal Amigowy 2010–2014.
Tux (maskotka Linuksa) © Larry Ewing,
Simon Budig i Anja Gerwinski.

<http://www.ppa.pl>

Rozwiązanie konkursów

W odpowiedzi na nasz konkurs wpłynęły artykuły, które pozwoliły wypełnić numer, który właśnie trzymacie w rękach. Podobnie jak poprzednio zdecydowaliśmy się docenić trud każdego z autorów za wkład w powstanie magazynu przyznając tzw. wierszówki. Na chwilę obecną może jest to skromna kwota, lecz niech to będzie pewnego rodzaju sposób na wyrażenie wdzięczności. Być może w przyszłości autorzy za swoją publikację będą nagradzani nieco hojniej. Dziękujemy za nadesłane artykuły.

W numerze

	AmiWest 2013 – podsumowanie najważniejszych informacji	4
	Amigujemy na Androidzie	5
	Protracker – część 2	10
	Changelt	12
	Metal Blob Solid: Blob Wars	13
	The Dig	14
	Nasza pierwsza gra – kurs programowania pod AmigaOS – część 6	16
	Adaper CF-IDE	18
	Retro Game Compo	20
	The Incredible Adventures of Moebius Goatlizard	23
	Freax – The Art Album	24
	Z listów starego amigowca	26
	Scenowe graffiti	27

Od redakcji

Już po raz jedenasty mam przyjemność pisać te słowa w tym właśnie miejscu, lecz tym razem czuję się tak, jakby to był pierwszy raz. Dlaczego? Powód jest prosty – jest to pierwszy wstępniak pisany do pierwszego numeru w pełni będącego pod skrzydłami wydawcy. Poprzedni numer został wydrukowany i przygotowany do wysyłki na długo przed jego oficjalną premierą właśnie dlatego, że chcieliśmy sformalizować raz na zawsze tę jakże istotną kwestię. To jest również przyczyna dlaczego numer dziesiąty zaliczył takie opóźnienie i dlaczego nasz magazyn pojawił się w poprzednim roku tylko raz. Na szczęście to wszystko mamy już za sobą i możemy działać dalej – pełną parą.

W tym numerze wiele znajdą dla siebie gracze – raczej Ci, żądni klasycznej zabawy na klasycznych Amigach. Po pierwsze mamy przegląd wszystkich tytułów (niektórych bardzo obiecujących), które zadebiutowały podczas Retro Game Compo. Po drugie mamy recenzje aż czterech gier: „Changelt”, „The Dig”, „The Incredible Adventures of Moebius Goatlizard” oraz „Metal Blob Solid: Blob Wars”. Nie samymi grami stoi jednak Amiga. W numerze zamieściliśmy podsumowanie najważniejszych informacji przekazanych na AmiWest 2013, obszerny artykuł dotyczący amigowania na systemie Android, a także rozpoczynamy nowy cykl przeznaczonych dla pasjonatów amigowej sceny, gdzie mamy nadzieję systematycznie pojawiać się będzie przegląd produkcji z ostatnich party. Dodatkowo znalazło się miejsce na krótką polemikę z ostatnim bardzo niepokojącym zjawiskiem „definicji Amigi” oraz „wyższości świat”, które niestety musimy często czytać na naszym forum. Liczymy, że tego typu artykuły zastąpią bezcelowe dyskusje i przerodzą się w coś bardziej produktywnego.

Co więcej w numerze znajdziecie zerkając kilka centymetrów wyżej – do spisu treści lub po prostu przewracając kolejne strony. Pamiętajcie, że magazyn istnieje dzięki Wam – piszcie artykuły o tym, co obecnie robicie na Waszych nowych, starych Amigach i wszystkim co z nimi związane. Postaramy się, aby to zostało zamieszczone i aby nasze pismo mogło doczekać się jeszcze kolejnych stu numerów...

Sebastian Rosa

P.S. Poniżej zdjęcie z obecności członków redakcji naszego portalu na listopadowej edycji „Dawne Komputery i Gry”.



AmiWest 2013 – podsumowanie najważniejszych informacji

W dniach 18-21 października odbyła się kolejna edycja spotkania AmiWest, które w amigowym świecie uzyskało status największego amigowego wydarzenia roku. Oglądając zdjęcia, niekoniecznie można się zgodzić z kwestią wypowiedzianą w ostatnich słowach poprzedniego zdania, jednakże bądź co bądź, tegoroczna odsłona AmiWest, od strony przekazanych nowych informacji, stanowiła największy w 2013 roku zbiór nowinek skierowana głównie do użytkowników i zwolenników systemu AmigaOS 4.x i powiązanego z nim sprzętu. AmiWest 2013 zostało zdominowane przez firmę A-EON, która przygotowała wiele atrakcji i niespodzianek mających szansę zrealizować się w najbliższej przyszłości.



Projekt Cyrus Plus

Firma A-EON ogłosiła rozpoczęcie naboru na liczbę 50 betatesterów płyt głównych *Cyrus Plus*. O projekcie Cyrus po raz pierwszy poinformowano w lipcu 2013. Zakładano dwie prototypowe wersje (Micro ATX i Full ATX) nowej płyty głównej, działającej w oparciu o procesory Freescale. Cyrus Plus to płyta główna stworzona przez Ultra Varisys. Posiada procesor Freescale QorIQ P5 64-bit e5500 dual-core PowerPC 2.0 GHz wykorzystywany w instalacjach wojskowo-lotniczych i ma zapewniony co najmniej 10-letni okres dostępności. Płyta będzie w stanie obsłużyć do 16 GB pamięci DDR3 i mieć możliwość uruchamiania firmware z kart MicroSD. Ponadto jest wyposażona w dwa porty SATA 2.0, sloty PCI-e x16, x4 (po jednym) i x1 (2 sztuki), x1/Xorro; PCI (2 sloty); USB 2.0 (6 zewnętrznych, 2 wewnętrzne), karta sieciowa, port szeregowy, układ Xena/Xorro. Pierwszeństwo w programie testowym będą mieć programiści, obecnie czynni betatesterzy AmigaOne X1000 i AmigaOS 4, jak również posiadacze AmigaOne X1000 "First Contact". Zainteresowani powinni nadysłać swoje zgłoszenia na adres mailowy contact@a-eon.com z tematem „Cyrus Plus Beta Test Programme”. Wymagane będzie wniesienie kaucji. Dalsze szczegóły będą udostępnione w odrębnej informacji, a wysyłka płyt planowana jest na koniec 2013 roku.

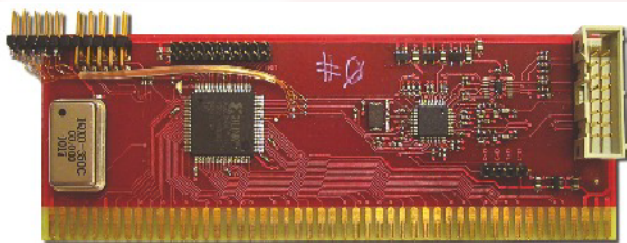
Płyty główne dostępne będą w trzech wariantach różniące się zamontowanym procesorem: Freescale QorIQ 32-bit P3041 1,5 GHz, 64-bit P5020 2 GHz, 64-bit P5040 2,4 GHz. Tutaj firma A-EON zwróciła się do amigowej społeczności o stworzenie nazwy dla nowych modeli komputerów, które powstaną na bazie wspomnianej płyty. Wyodrębniono 12 grup nazewnictwa, a w każdej z nich trzy warianty, zgodnie z zastosowanymi odmianami procesorów. Każdy mógł mieć swój udział w nadaniu nazwy dla nowych produktów. Na początku stycznia 2014 roku ujawniono, że komputery będą się nazywać **AmigaONE X3500**, **AmigaONE X5000/20** i **AmigaONE X5000/40** (w zależności od procesora).

Nemo

Rozpoczęcie prac nad nową płytą dla komputerów serii AmigaOne mogło oznaczać zarzucenie rozwoju płyty Nemo, będącej niejako poprzedniczką Cyrus Plus. Czy tak się stało – czas pokaże. Na chwilę obecną firma A-EON rozpoczęła osobną sprzedaż płyt głównych Nemo wyposażonych w procesor PA Semi PA6T 1682M, a 64-bit dual core PowerPC, umożliwiając tym samym samodzielne złożenie komputera na ich bazie. Płyty główne są do nabycia u oficjalnych dystrybutorów, czyli w sklepach firm Amiga-Kit oraz Alinea Computer (nowy dystrybutor) i dostarczane są wraz z systemem AmigaOS 4.1.

Prisma Megamix

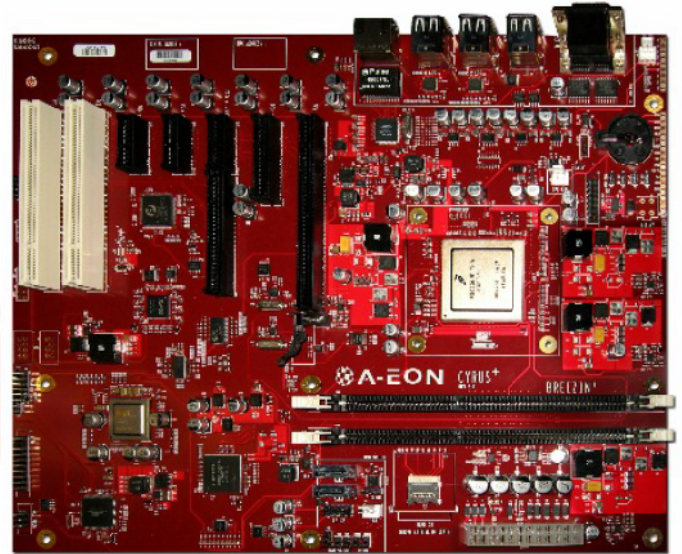
Firma A-EON nie przestaje jednak zaskakiwać i zaznacza chęć wejścia również na rynek Amig klasycznych. Wspólnie z Michałem Boehmerem (możemy go kojarzyć z firmą E3B), który jest odpowiedzialny za stronę sprzętową oraz Ianem Gledhilliem (strona software'owa – sterowniki) pracują nad nową kartą dźwiękową dla Amig 1200, 2000, 3000 oraz 4000. *Prisma Megamix* wyposażona będzie w układ VS1063 DSP zapewniający obsługę m. in. formatów MP2, MP3, WMA, OGG, LC-AAC, HE-AAC, FLAC, IMA, WAV PCM. Kartę będzie można wpiąć zarówno do portu zegara (stream 8-bitowy), jak i w slot Zorro II (stream 16-bitowy). Planowana jest również wersja dla slotu Xorro PCI-E.



Prisma Megamix

Hyperion Entertainment

Na tegorocznym AmiWest firma Hyperion Entertainment była nieco w cieniu mając na względzie to, co zaprezentował ich partner w interesach. Użytkownicy i pasjonaci AmigaOS 4.x niczym nie zostali zaskoczeni. Steven Solie przedstawił podsumowanie tego, co udało się dokonać od ostatniego AmiWest oraz jakie są najbliższe plany.



Cyrus Plus

- oficjalnie potwierdzono zawieszenie projektu netbooka działającego w oparciu o AmigaOS 4.x. W przytoczonym fragmencie korespondencji Bena Hermansa, problemem okazał się brak porozumienia w kwestiach finansowych z chińskim dostawcą (próg cenowy, warunki płatności, wartość zaliczki). Ostatnie zdanie korespondencji pozwala jednak twierdzić, że całość nie została ostatecznie przekreślona,
- w temacie Gallium3D ukończona została akceleracja programowa i obecnie trwają prace nad implementacją części WinSys. Zmierzono się z problemem wymagań sprzętowych stawianych systemowi graficznemu, jego wydajności oraz możliwości uruchamiania sterowników spoza spektrum Mesa i Gallium,
- prace na kernelu obsługującym wiele rdzeni procesorów postępują. W ślad za zmianą nazewnictwa na X-Kernel nastąpiło m. in. przepisanie Task Schedulera (menadżer zadań) oraz uniezależnienie od siebie zadań wykonywanych na różnych rdzeniach procesora,
- wydanie AmigaOS 4.2 jest uzależnione od wydania systemu graficznego Gallium 3D oraz prawdopodobnie także od dodania pełnej obsługi wielu rdzeni,
- siódme uaktualnienie AmigaOS 4.1 nastąpi wraz z dostępnością płyt głównych Cyrus Plus. Skonsoliduje ono wszystkie do tej pory wydane uaktualnienia sprawiając, że będzie pełnym i aktualnym na dany moment systemem.

Steven Solie wspomniał również o wydaniu nowego SDK oraz przynajmniej 58 mniejszych uaktualnień systemu.

Sebastian Rosa



Amigujemy na Androidzie

Jakiś czas temu napisałem recenzję emulatora Amigi „UAE4Droid” przeznaczonego dla systemu Android. Program testowałem na jedynym dostępnym mi wtedy urządzeniu z tym systemem (Sony Ericsson Xperia Arc S). W międzyczasie kolekcja moich urządzeń przenośnych zdążyła się powiększyć o tablet – ot nic wielkiego, zwykły Asus Transformer TF101G pracujący na Androidzie. Tablet ten nie posiada oszałamiających parametrów w porównaniu z tabletami dostępnymi dziś w sprzedaży (pojawił się w sklepach w 2011 r.), jednakże mając już jakieś takie doświadczenie z emulacją Amigi na telefonie i mając do dyspozycji dużo większy ekran, postanowiłem sprawdzić, jak Asus poradzi sobie w roli przenośnej Amigi. Od czasu recenzji „UAE4Droid” wody w Wiśle trochę upłynęło, sklep z aplikacjami dla Androida również się rozrasta, nabrałem więc podejrzeń, że być może pojawiło się w nim o wiele więcej aplikacji związanych z Amigą niż miało to miejsce przy pisaniu recenzji powyższego emulatora.

Amitablet na zakupach

Zacząć by wypadało od tego, że moje oczekiwania rozbudziły parametry samego tabletu: przede wszystkim wielkość ekranu – 10,1” kontra 4,2” robi wrażenie, podobnie jak i pozostałe parametry tabletu w porównaniu z telefonem. Szybszy procesor (dwa rdzenie 1 GHz), grafika w rozdzielczości 1280x800 w technologii LED i pamięć 1 GB – to musi dobrze wróżyć na przyszłość. O ile na telefonie nie zauważyłem żadnych problemów z emulacją, nie było zwolnień, przekłamań grafiki i tym podobnych fajerwerków, o tyle mógł denerwować mały ekran – obsługa Workbench przy rozdzielczości 480x854 piksele na niewiele ponad czterech calach stanowi nie lada wyzwanie – wręcz męczący.



Lion King na Uae4All pod Androidem



Zool 2 na Uae4All pod Androidem

Mamy już zapas mocy, duży ekran a więc... pora odwiedzić Sklep Play i sprawdzić, co też oferuje on nam – amigowcom. Szybkie wpisanie słowa „Amiga” w polu „Szukaj” zwraca nam całkiem sporo odnośników w ten czy inny sposób powiązanych z wyszukiwaną frazą. Niestety, żeby znaleźć coś ciekawego trzeba przebrnąć również przez gąszcz mniej przydatnych dla nas aplikacji (celowo nie używam słowa „bezużytecznych”, zdając sobie sprawę z tego, że są i amatorzy takowych). Słowo „Amiga” bowiem występuje tu w różnych kontekstach i czasem trafiamy na emulator, a czasem tylko na grę, która, jak pisze autor, przypomni Ci stare, dobre czasy spędzone nad „przyjaciółką”. Czasem pojawi się – z wiadomych przyczyn – w hiszpańskim opisie programu, a czasem nie ma z Amigą nic wspólnego. Przykła-

dem tego ostatniego są takie programy, jak np. „Avril Lavigne Puzzle Jigsaw” czy też „Radios Ecuador” – jak się to ma do zadanego zapytania wie chyba tylko Google.

Myliliby się jednak każdy, kto twierdzi, że nie da się znaleźć kompletnie nic z interesującej nas dziedziny. Da się, ale jak to zwykle bywa potrzeba trochę wysiłku i czasu. Na potrzeby tej recenzji wybrałem kilka programów, które najbardziej mnie zainteresowały, starając się przy okazji nie skupiać wyłącznie na emulatorach, ale również na wszystkim, co może się jakoś wiązać z Amigą. Nie mogło by jednak być inaczej, niż sprawdzić na początek czy faktycznie uda się zamienić tablet w coś, na czym uruchomi się parę plików ADF, pobawi Workbenchem a wszystko to, gdy za oknami pociągu mijają nam kolejne stacje.

UAE4Droid

Mając już doświadczenie z tym emulatorem, postanowiłem puścić go na „pierwszą linię ognia”. Program pobrałem w wersji 0.998.2, a więc wydaje się, że w nowszej niż recenzowana już przez mnie wersja 0.99. Rzut oka na zakładkę „Co nowego” pozwala się w tym przekonaniu utwierdzić – wygląda na to, że poprawiono działanie programu dla Androida w wersji 4.2 (tzw. fix, ale czego, tego nie udało mi się doszukać, niestety. Na pewno poprawiono problemy z ładowaniem stanów gry w niektórych przypadkach i obsługę niektórych urządzeń). Reszta zasadniczo się nie zmieniła – mamy do dyspozycji cztery stacje wirtualnych dyskiek, możliwość określenia lokalizacji naszego Kickstartu itp. Opcje są bardzo przejrzyste i w zasadzie dostępne dla użytkownika są tylko te, które są wymagane do poprawnej emulacji. Niestety nadal nie jest obsługiwany ROM w wersji 3.x (najwyższy jaki udało mi się uruchomić to 2.05), a także zmiana Kickstartów w locie, ale w zamian za to



AmigaOS 3.1 z pliku ADF na ROM-ie 2.05 uruchomił się bez problemu. O komforcie obsługi na ekranie takiej wielkości nawet nie warto wspominać – rewelacja. Kursor myszy obsługuje się albo specjalnymi przyciskami po lewej stronie ekranu, albo palcem po ekranie – jak na dotykowe urządzenie przystało. Wciśnięcie przycisku fire joysticka uzyskamy po dotknięciu małego kółeczka na ekranie po prawej. Zwrócić należy jednak uwagę, że z racji tego, iż kursory sterujące są w szarym kolorze – podobnie jak domyślne tło Workbench, można ich zwyczajnie nie zauważyć. Szczególnie odczuwalne jest to po wybraniu opcji wyświetlania obrazu jako „Stretched”, czyli rozciągnięty na cały ekran. Nie ma też problemu ze zdefiniowaniem własnych przycisków, w tym właśnie obu kliknięć myszką. Czasem pojawiają się różne dziwne artefakty (czarne kwadraty), ale nie jest to jakoś szczególnie uciążliwe i pojawia się zazwyczaj na głównym ekranie Workbench. Wczytywanie danych z wirtualnych stacji dyskiety jest dość powolne.

Przetestowałem również sporo gier, m. in. „Blues Brothers” (nie działała), „Commando” (reset po głównym ekranie), „Ikari Warriors” (zawieszka po uruchomieniu się pierwszego etapu). Co działało? Równie sporo – wymienić wystarczy takie tytuły, jak „Jumping Jack’son”, „Pac-Mania” czy „Rick Dangerous 2”. Z ciekawości sprawdziłem czy zadziała „Colorado” testowane wcześniej na telefonie – niestety tylko do logo Silmarils i do tego z przekłamaniami grafiki. Uwaga dla zainteresowanych – tak, „superzabka” działa i mam wrażenie, że nawet za szybko. Prawdą jednak jest, że nie wszystko się da uruchomić, dodatkowo chipset AGA wciąż nie jest obsługiwany, nad czym pozostaje tylko ubolewać (przy mocy tabletu jest to co najmniej dziwne). Dobrym źródłem wiedzy odnośnie wymagań poszczególnych gier jest baza HOL (odnośnik w ramce). Pamiętajmy też, że niektóre gry wymagają wciśnięcia klawisza 1 lub 2, w zależności ilu graczy ma brać udział w zabawie, a wyboru dokonuje się po wejściu do menu głównego i wybraniu opcji „Switch to joystick mode/Switch to keyboard mode”. Ogólnie – uczciwa czwórka z plusem w szkolnej skali 1–6.

Omega 500 (UAE for Android)

W sumie dość miłe zaskoczenie, bo autorem tego emulatora jest nasz rodak, Michał Galiński, ale reszta już nie napawa takim optymizmem. Pierwsze starcie z aplikacją nie wróży niczego dobrego – plik z Kickstartem (w wersji 1.3), przemianowany na Kick.rom bezwzględnie musi być umieszczony w katalogu root pamięci wewnętrznej urządzenia. Jeśli już przebrniemy przez proces uruchomienia programu, to... zaskoczy nas ilość dostępnych opcji konfiguracyjnych. A raczej ich brak! Wyboru możemy dokonać jedynie między plikiem ADF, który chcemy urucho-



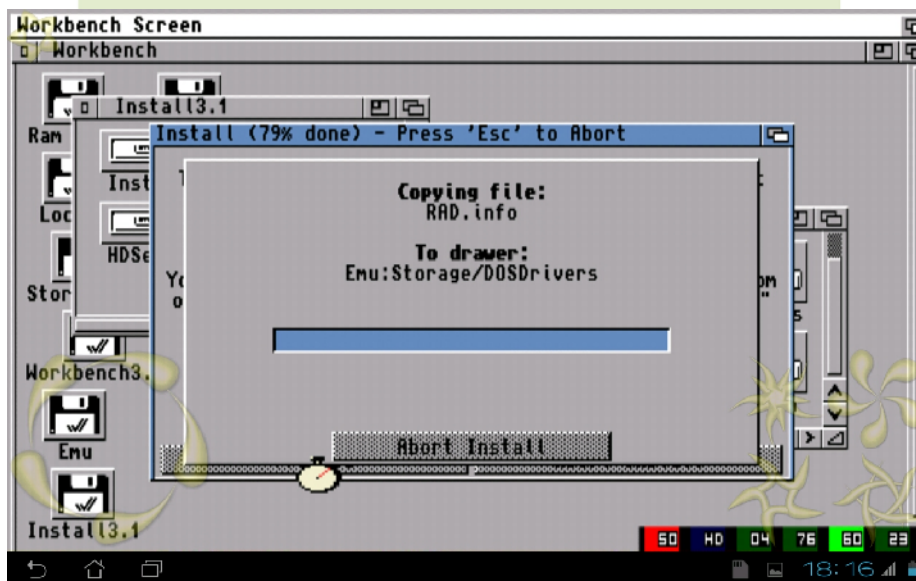
Superfrog na UAE4Droid pod Androidem

mić i wyborem trybu pracy joysticka: QuickJoy do gier platformowych i GridJoy do shooterów i gier sportowych. Opcja ustawień zaawansowanych umożliwia nam jedynie zmianę przezroczystości przycisków na ekranie i zmianę ustawień tajemniczego „CPU Emulation Core”. „Soft Reset” i „Power Off” dostępne pod ikonką „zakręconej strzałki” również nie dodają splendoru aplikacji. Dodatkowe zamieszanie wprowadzają dwa małe przyciski, umieszczone u góry po obu stronach ekranu. Ten z prawej powoduje przełączanie między klawiaturą, myszką i joystickiem, a ten z lewej – chyba nikt nie wie (u mnie wciśnięcie go nie wywołuje żadnej reakcji). Wspomniane już powyżej gry nie uruchomiły się i nie podejrzewam, żeby kompatybilność tego emulatora z Amigą była jakaś specjalnie wysoka. Aż chce się powiedzieć – jaka szkoda... Testowana wersja to 0.2.1 – może to jakieś wstępne stadium, tym bardziej, że w tytule jest UAE – implementacja tegoż na Androida? Hmm... Autorowi życzę inwencji twórczej – potencjał jakiś w tej aplikacji na pewno drzemie, a co z tego wyjdzie – okaże się pewnie z wydaniem kolejnej wersji (o ile w ogóle). Ocena ogólna: słabutkie 3 i to z małym minusikiem.

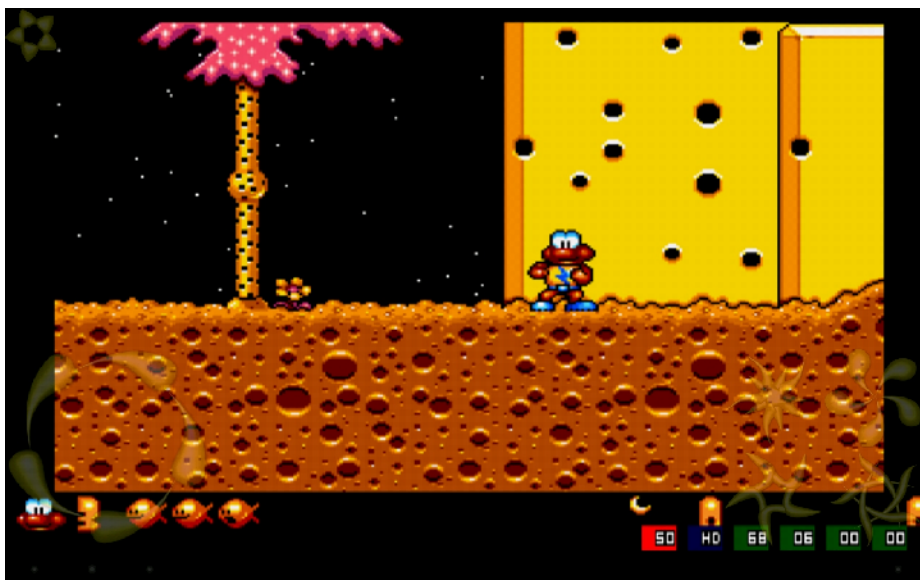
Uae4all2-SDL

Uruchamiając na tablecie powyższą aplikację nie byłem nastawiony jakoś specjalnie optymistycznie, bo i niby dlaczego? Mając w pamięci poprzednie emulatory z pewną dozą rezerwy „tapnąłem” w ikonkę programu. Domyślając się wcześniej, że potrzebne będą pliki Kickstartu, zawnazę skopiowałem do pamięci tabletu odpowiedni plik (u mnie był to folder /sd-card/android/data/pandora.uae4all.sdl/files/kickstarts) i... moje spotkanie z tym programem zakończyło się błyskawicznie komunikatem o braku pliku kick.rom. Po wielu nieudanych próbach rozwiązanie przyniosła strona producenta, a konkretnie zawarte na niej komentarze, choć nie do końca. Okazuje się bowiem, że nie wystarczy odpowiednio nazwać pliku i skopiować go do folderu „kickstarts”. Ścieżkę tę należy również podać w opcjach dostępnych przy uruchamianiu aplikacji i opcje te nie są tożsame z tymi dostępnymi w głównym programie! Po wielu próbach w końcu dotarłem do tego, że żeby w ogóle móc uruchomić Uae4all2 należy, po ukazaniu się okna z logo SDL, stuknąć u góry na przycisku „Change device configuration” i wybrać opcję „Data installation location”. Następnie wybieramy „Specify directory” i na końcu ścieżki dopisujemy folder „kickstarts”. Teraz już tylko „OK”, akceptujemy okienko, które się pojawi bez podawania parametrów, ponownie „OK” i po chwili naszym oczom ukazuje się główne menu programu.

Przyznam uczciwie, że serce zabiło mi mocniej na widok opcji umożliwiającej wybór typu Amigi. Amiga 1200? Czy to możliwe? I na do kładkę obsługi twardego dysku?



Instalacja systemu na UAE4All pod Androidem



James Pond 3 na UAE4All pod Androidem

Przeglądając na szybko dostępne opcje byłem zdumiony. Jest opcja wyboru procesora, typu i ilości różnych pamięci (Chip/Slow/Fast), chipsetu (OCS/ECS/AGA), startu system z pliku HDF lub z katalogu, rozdzielczości grafiki itp. Słowem – mnóstwo opcji, zbyt dużo by je tu wszystkie opisać. Skonfigurowałem więc sobie na szybko skromny zestaw A1200 z 8 MB pamięci Fast i „wrzuciłem” do stacji wirtualną dyskietkę z Workbenchem. Zanim wybrałem z menu opcję „Reset” ustawiłem sobie jeszcze tylko rozdzielczość ekranu na 640 x 256 i „Pandora CPU-Speed” na 1000 MHz. Jeśli ktoś lubi, może sobie jeszcze ustawić „Status In” na „On”, dzięki czemu będzie miał podgląd na to, co się dzieje ze stacjami dysków. Po zabootowaniu systemu mogłem już podziwiać blat Workbenchu w całej krasie, ale niestety pojawił się nowy problem – kursor myszki kompletnie nie reagował na dotknięcia ekranu. Z pomocą przychodzi ponownie znane nam już menu „Change device configuration”, opcja „Mouse emulation”, a następnie „Mouse emulation mode”. U mnie pomogło wybranie metodą prób i błędów opcji „Small, touchpad mode”. Od tej pory kursor działał już jak należy.

Śmieszna sprawa wydaje mi się sterowanie myszką i joystickiem na ekranie tabletu. O ile jest ono naprawdę wygodne, to w życiu nie widziałem tak zabawnej grafiki przycisków – zamiast zwykłych i szarych „guzików” mamy tu przedziwny pokaz fantazji grafika, a jedyne z czym one mi się kojarzą stylistycznie, to z... Chinami! Swoje zadanie spełniają jednak znakomicie.

Nie miałem akurat pod ręką żadnego pliku HDF z systemem, ale postanowiłem utworzyć dysk w katalogu i tam spróbować zainstalować system z dyskietek. Jak można było się domyślić – system zainstalował się poprawnie, start z dysku to czysta przyjemność. Przeglądając stronę producenta można się również dowiedzieć, że możliwa jest instalacja pakietu WHDLoad, a co za tym idzie gotowych paczek z grami! Czegoż chcieć więcej?

W końcu przyszła więc i pora przetestować gry a tym, co najbardziej mnie interesowało, była oczywiście obsługa chipsetu AGA. Wybrałem więc kilka losowych gier w postaci plików ADF i przystąpiłem do dzieła. Muszę przyznać, że poziom zgodności jest bardzo wysoki i nie za-

notowałem jakichś większych problemów. Prawie wszystkie uruchomione przeze mnie gry na chipset AGA działały bardzo dobrze i były grywalne, a były to m.in. „James Pond 3”, „Lion King” czy chociażby „Zool 2”. Nie uruchomiły się „Aladdin” i „Banshee”. Z gier przeznaczonych dla starszych układów spora większość nie działała, np. „Pang” („Colorado” tradycyjnie nie zadziałało na żadnej konfiguracji). W tej więc dziedzinie tablet z Uae4all2 śmiało może zastąpić konsolę do gier w podróży czy na nudnym wykładzie.

Podsumowując ostatecznie ten program, ocenić można go w ten sposób – masa opcji i dobra zgodność kontra skomplikowana konfiguracja. Jak widać, dla każdego coś miłego i za to właśnie, między innymi, należy mu się jedyna słuszną w tym wypadku ocena – brzuchate 6.

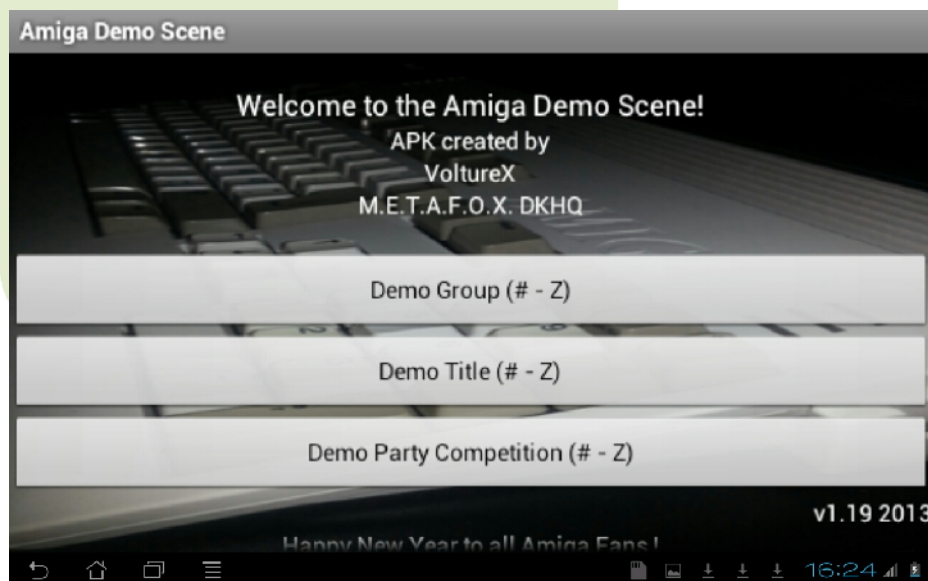
Amiga Demo Scene

Przedziwna aplikacja. Z założenia gromadzi w sobie demo dostępne na Amigę, ale... głównie te, które są do obejrzenia na YouTube. Jaki w

tym sens, skoro każdy może sobie znaleźć demo, które go interesuje i obejrzeć w domowym zaciszu na wspomnianym kanale? Szczerze? Nie wiem – zawsze sam wybieram sobie, co chcę zobaczyć. Domyślam się jednak, że dla osób, które chcą dowiedzieć się czegoś o scenie Amigi, aplikacja może być przydatna. Mnóstwo materiału do przetrzeźwienia, brak za to jakichkolwiek opcji poza zakończeniem aplikacji. Są za to dane o twórcach demo, roku wydania, wykorzystywanym chipsecie, nagrodach i minimalnych wymaganiach. Brakuje mi za to jakiegokolwiek interakcji z użytkownikiem – to po prostu taka zwykła przeglądarka dem. Pochwalić można ideę, a zganić wykonanie, tym bardziej, że rozdzielczość, w której wykonano aplikację w zachwyty nie wprawia. Jak pisze sam autor, program nie zawiera jeszcze całej bazy dostępnych dem i ma problemy z wyświetlaniem stron YouTube zawierających w nazwie znaki „&_”. Aktualna, testowana wersja to 1.19 z 2013 roku. No cóż, przed autorem jeszcze długa droga, głównie odnośnie prezentacji programu, a innym można życzyć tylko miłych chwil sam na sam z demosceną Amigi. Ocena – 3 z plusikiem.

Amiga Mod Player Lite

Była aplikacja do pooglądania scenowych utworów, to teraz przyszedł czas na coś związanego z muzyką. Aplikacja z założenia ma służyć odtwarzaniu plików w formacie MOD, XM, SM, MMD1 i innych. Jakich? Nie bardzo chyba wiadomo. Nie znalazłem niestety opcji odtwarzania własnych modków, ale jakoś mnie to nie odstraszyło, skoro baza utworów dostępna dla programu jest tak szeroka (tak, tak, potrzebne jest połączenie z siecią)... Opcje są wręcz spartańskie – można, co prawda, ustawić sobie opcję „offline”, ale czy to zadziała bez uprzedniego pobrania utworów – jeszcze wątpię. Program potrafi się również wręcz koncertowo zawiesić, np. po wejściu w opcję (jedną z nielicznych) „Greetings”. Szata graficzna – no cóż, za bardzo nie ma co opisywać. Pobrane/odtwarzane dźwięki faktycznie, tak przynajmniej pokazuje się to w bazie utworów, pochodzą z Fasttrackera, Protrackera czy innego Screamtrackera. Za to, jako człowiek wesoły z natury, setnie się ubawiłem po kliknięciu na „About”. Śmiało się z dobre dziesięć minut. Dlaczego? Mam wrażenie, że autor



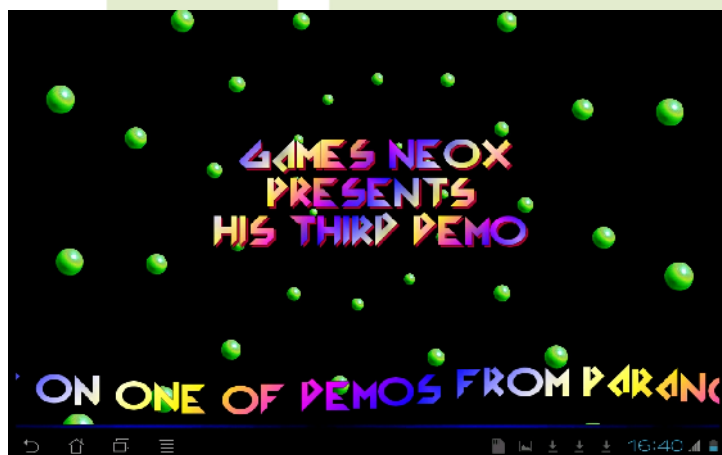
Amiga Demo Scene



aplikacji większość swojego czasu poświęcił na dopieszczenie tejże opcji, zapominając o samym programie. Czegoś takiego jeszcze nie widziałem – wspomniane wyżej okienko zapisane jest drobnym maczkiem (dokładnie tak – przysłowiowym maczkiem) opisami licencji, na której program jest udostępniony, możliwymi uszkodzeniami i ich następstwami itp. Dosłownie jakby był to program ze stajni któregoś z wielkich koncernów wart krocie i mający spory udział w rynku oprogramowania. Abstrahując już od tych dziwności dodam, że spotkałem się z opinią, jakoby program po pobraniu i odtworzeniu pięciu utworów narzucał się użytkownikowi z zakupem pełnej wersji. W moim przypadku nic takiego nie miało miejsca. Pobrałem więcej modków i jakoś nikt mnie nie namawiał do niczego. Reasumując, ocena może być chyba tylko jedna – solidna dwójka. Natomiast gdybym miał oceniać program pod względem poprawy nastroju przeciętnego Kowalskiego – zdecydowanie szóstka.

NXDEMO 00A

Jeden z przykładów, o których była mowa wcześniej, tj. Demo, w którym specjalnie nie widzę niczego związanego z Amigą. No może po za tym, że autorzy piszą sobie w markecie, iż scena Amigi powróciła, że to trzecie demo w starym, dobrym amigowym stylu itp. No i do tego piękna ikonka boinga. Demo przypomina raczej ubogie cracktro przed trzecioligową grą. Jak to leciało w pewnym programie telewizyjnym? A, już wiem – jestem na nie. Ocena – 2 z plusem i natychmiastowa kasacja. Poprzednimi odsłonami nawet nie zamierzam sobie zawracać głowy.



NXDEMO 00A

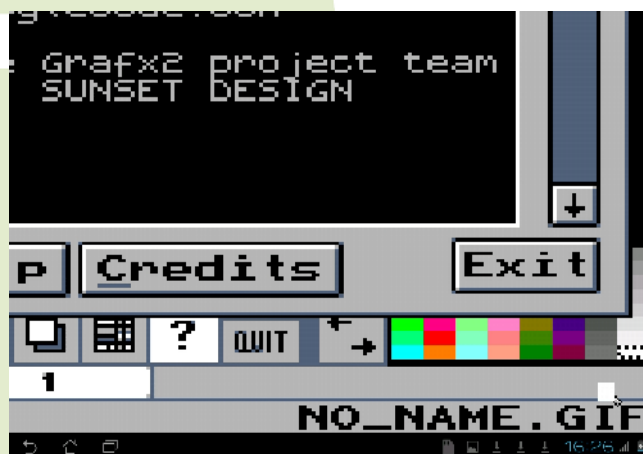
Speedball 2 Evolution Free

Zaraz, zaraz. Jest ktoś, kto tego tytułu nie zna? Tu w odmłodzonej wersji. Grafika wg mnie traci leciutko jakimś flashem, ale co z tego? Nawet piłka błyszczy tak, jak dawniej na Amidze. Są powerupy, przeciwnicy, można sobie wybrać skład naszej drużyny. Błyszczy jak dawniej logo Bitmap Brothers, choć producentem jest Vivid Games. Niestety, aplikacja posiada w nazwie słowo „Free”, czyli można się spodziewać jakichś ograniczeń. I rzeczywiście tak jest – dostępna jest tylko opcja „Career Mode”, a ponadto mecze są ograniczone tylko i wyłącznie do dwóch i pół w lidze (jakkolwiek dziwnie by to nie brzmiało). Pobawić się jednak można, a dla spragnionych gry z większą ilością opcji pozostaje tylko zakup „Speedball



Speedball 2 Evolution Free

2 Evolution” w cenie 9,99 PLN. No i kupujący wie, co kupuje, bo jest to jeden z chlubnych wyjątków, gdzie potencjalny nabywca gry wie, za co chce zapłacić i ma możliwość pobawić się programem na żywo w okrojonej, ale mimo wszystko w miarę funkcjonalnej



Grafx2

wersji. Czekamy teraz na „Chaos Engine” i „Gods” dla Androida, a program oceniam na 5 minus (za obcięcie funkcjonalności).

czaj jest po wejściu w menu „Change device configuration”, pojawiające się na belce u góry po uruchomieniu Grafx2. Można tam zmienić ustawienia myszki, obrazu, przemapować klawiaturę czy włączyć akcelerometr. Niestety skutkuje to tylko jeszcze gorszym efektem a i sam „cursor-kwadrat” dziwnie się rozjeżdża i nie trafia w odpowiednie przyciski. Udało mi się pobawić w zmiany kolorów, narysować parę punktów. Jest tu sporo różnych opcji, ale brak mi jakiegokolwiek instrukcji tłumaczącej co i jak krok po kroku. Niemniej jednak osoby znające ten program poczuć się, jak u siebie w domu, widać również spory potencjał w samej aplikacji. Dodać należy również, że istnieje pełnoprawna aplikacja Grafx2 dla wielu innych systemów m. in. AmigaOS 3.x i 4.x, MorphOS, Haiku, Linux, Windows i wiele innych. Za chęci przeprogramowania być może ciekawego programu graficznego z Amigi na Androida – 4 z plusem, za wykonanie – niestety tylko trójka. No, dla upartych plusik do całko-

Grafx2

Bardzo ciężko będzie mi ocenić tę aplikację. Z jednej strony program (a dokładniej wszystkie ikonki, menu itp.) daje wrażenie pracy na prawdziwej Amidze, z drugiej wydaje mi się jeszcze niedokończony (artefakty w postaci kwadratów na ekranie są wszędzieobecne), a z trzeciej żaden ze mnie grafik, żeby wypowiadać się w tej dziedzinie. Odnosnie artefaktów – domyślałem się, że na ekranie powinien pojawić się kursor – być może winny jest tu układ graficzny mojego tabletu. Uruchamiając aplikację pyta ona zresztą jak ma być wyświetlany i obsługiwany ekran – opcja dla tabletu powoduje dodatkowe „efekty specjalne”. Nie ina-

Strona najlepszego emulatora Amigi dla Androida –
http://anddev.at.ua/load/emulators/uae4all2_sdl_pandora_edition/2-1-0-20

„Żywa” tapeta z Boing Ballem –
<https://play.google.com/store/apps/details?id=com.globus.boingballwlp&hl=pl>

Recenzja programu UAE4Droid na smartphony –
<http://www.droidandboing.net/?p=202>

witej oceny można od biedy dodać.

AmigaBoingBall

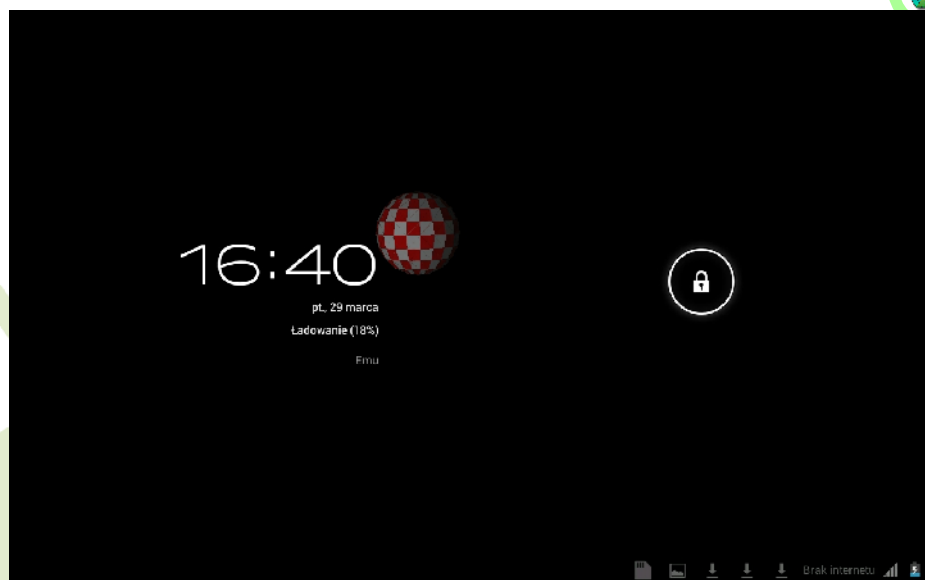
Wszyscy pamiętają demo „Boing Ball”? Nie? To jest okazja sobie przypomnieć historię przy pomocy naszego tabletu. W sumie program niczemu nie służy – zwykła drobnostka, która cieszy. Animacja bardzo płynna, dodatkowo można sobie zerknąć na obciążenie procesora naszego urządzenia. Niestety u mnie nie działał przycisk opcji. Wizualnie – na pięć, za pomysł przywrócenia do życia ikony – leciutka szósteczka.

Boing Ball LWP

Rzecz jeszcze drobniejsza niż opisywane wyżej demo. Wszyscy lubią mieć fajną tapetkę na swoim smartfonie/tablecie, prawda? No to czemu nie zainstalować sobie latającej, czerwono-białej piłeczki? Skrót LWP wszystko tłumaczy – Live Wallpaper, czyli tzw. „żywa tapetka”. Świetnie, bo i tak też ona działa. Uwaga – poprawia humor! Ustawień jakichkolwiek brak, bo i po co? Piłeczka lata sobie w tą i w tamtą po ekranie tabletu i to wystarczy, żeby załapać klimat. Dodatkowo jeśli mamy na desktopie widżet z pogodynką to nasz piłeczka przepięknie się chowa za chmurki (o ile są). A przy włączonej blokadzie ekranu – bez ikonek na pulpicie – mamy już tylko czarne tło i naszą bohaterkę. Na proste rzeczy najtrudniej wpaść – 6 bez wahania.

Skromna garść uwag technicznych

Pamiętajmy o jednym – wybór aplikacji będzie większy, jeśli... zdecydujemy się zapłacić parę groszy. Zazwyczaj nie są to wielkie kwoty – rzędu kilku lub kilkunastu złotych. Najczęściej wersje darmowe są oznaczane w androidowym markecie jako „Lite”, różnią się od płatnej wersji ilością poziomów, ograniczonym czasem lub czymkolwiek innym, co może nam uprzykrzyć życie. Szkoda, że w większości przypadków nie ma możliwości pobrania wersji demo danej aplikacji. Z tego też powodu np. nie umieściłem w tym pobieżnym zestawieniu takich aplikacji, jak „guru Meditation Aminet



Boing Ball LWP

RSS” czy „AnUAE4All” – pierwsza to czytnik nowości z Aminetu (2,99 PLN), a druga to emulator A500 (9,90 PLN). Kwoty nie są wielkie, ale czy warto inwestować w te programy bez możliwości sprawdzenia ich na naszym urządzeniu? Druga rzecz – trzeba wziąć pod uwagę rozmiary pobieranych aplikacji, gdyż niektóre z nich mogą sporo ważyć, o czym zresztą lojalnie poinformuje nas komunikat ze sklepu Play. Najlepiej oczywiście pobierać aplikacje przy pomocy połączenia Wi-Fi, względnie być posiadaczem abonamentu w dowolnej sieci z nielimitowanym dostępem do globalnej sieci.

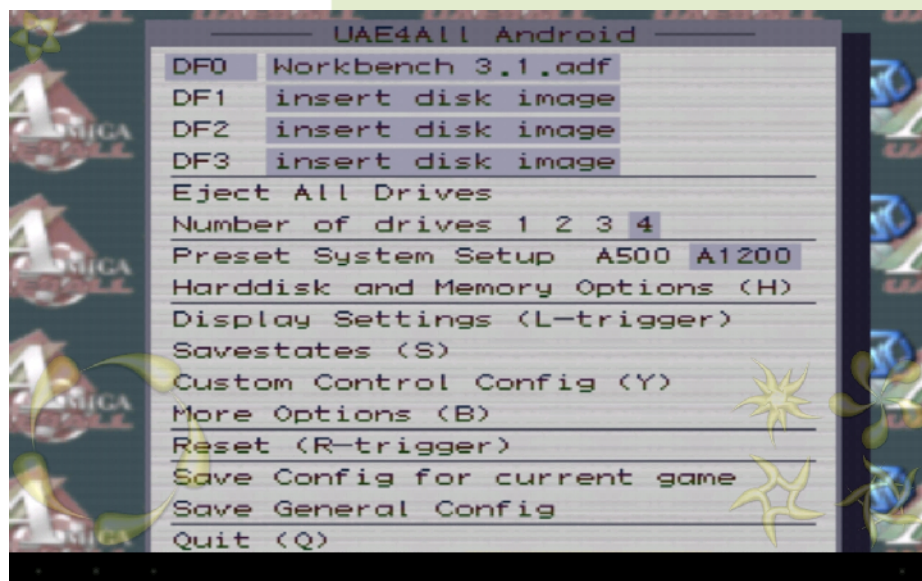
Komu w drogę, temu...

Jak widać z powyższej prezentacji programów dla Androida, nie jest ani źle, ani dobrze. Są wartościowe aplikacje, ale i trafiają się średniaki czy wręcz aplikacje całkowicie nieudane. Całe szczęście, że jeszcze na razie autorzy oprogramowania z tej ostatniej kategorii nie życzą sobie opłat za wytwory swojej wyobraźni (możliwe, że w tej kwestii się mylę, oby nie). Rozczarowujące na pewno może być to, że

posiadając tablet spodziewać by się można czegoś ciekawego – w końcu parametry takiej przeciętnej maszynki są raczej wystarczające do emulacji Amigi. Aż wierzyć się nie chce, że w środowisku związanym z Amigą nie ma nikogo, kto napisałby coś sensownego dla Androida.

Jeśli jesteś fanem Amigi, masz przy sobie tablet i jesteś w drodze, zapewne Ci się nudzi. Nie czytaj wiadomości o polityce i pogodzie – włącz sobie emulator Amigi, uruchom „War Zone”, pobaw się zegarem w Workbenchu. Ale pamiętaj o jednym – zanim wybierzesz się w podróż, dobrać sobie najpierw odpowiedni zestaw aplikacji i przetestuj je w domu. Zapewniam, że mimo wszystko warto.

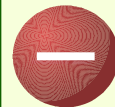
Marcin „Emu” Skawiński



Ekran ustawień UAE4All



- łatwe pobieranie z marketu Google
- każdy znajdzie coś dla siebie
- działa, gdy jesteś w drodze lub na nudnym spotkaniu



- spora część aplikacji jest płatna lub posiada ograniczenia
- w niektórych przypadkach potrzebne jest dobre połączenie z siecią (przy pobieraniu aplikacji)
- brak możliwości przetestowania wersji demo z posiadanym urządzeniem
- niektóre aplikacje pozostawiają wiele do życzenia pod względem funkcjonalności...
- ... a niektóre nie mają większego sensu



Protracker – część 2

Poprzedni odcinek kursu zakończyliśmy na opisie pierwszej z siedmiu zakładek, które określiliśmy mianem opcji. W tym odcinku zajmiemy się pozostałymi sześcioma.

SAMPLE EDITOR ciężko nazwać jakimiś opcjami. To wbudowane narzędzie do wszelkiego rodzaju operacji na samplu lub instrumencie. Składają się na niego:

- **TRACK/PATTERN/SAMPLE** (przycisk cykliczny) – przełącza zakres, w obrębie którego mają być dokonywane operacje DELETE, EXCH(AN)GE i COPY. W przypadku TRACK operacje zostaną wykonane w obrębie aktualnie wybranej ścieżki, PATTERN – w obrębie patternu, SAMPLE – na samplach.
- **FROM nn/TO nn** – określa parametry dla operacji EXCH(AN)GE i COPY.
- **DELETE** – kasuje aktualnie wybrany sampel (patrz SAMPLE nn) ze ścieżki (ustawienie: TRACK) lub całego patternu (ustawienie: PATTERN). Wykonanie tej operacji nie powoduje usunięcia sampla z pamięci komputera.
- **EXCHGE (EXCHANGE)** – przy ustawieniu TRACK/PATTERN zamienia miejscami sample w obrębie ścieżki lub patternu. Wszystkie sample o numerze określonym w „FROM nn” w ścieżce lub patternie zostaną zamienione miejscami z samplami określonymi przez „TO nn”. Przy ustawieniu SAMPLE, zostaną zamienione miejscami sample (instrumenty).
- **COPY** – przy ustawieniu TRACK/PATTERN sample o numerze określonym w polu „FROM nn” w ścieżce lub patternie zostaną zastąpione samplami określonymi w polu „TO nn”. Przy ustawieniu SAMPLE zostanie skopiowany sampel (instrument) o numerze „FROM nn” do „TO nn”.

Poniższe opcje mają zastosowanie wyłącznie na aktualnie wybranym samplu i działają niezależnie od wymienionego wyżej ustawienia TRACK/PATTERN/SAMPLE:

- **KILL** – usuwa aktualnie wybrany sampel z pamięci komputera. Wykonanie tej operacji nie powoduje jednak usunięcia go ze ścieżek/patternów.
- **MIX** – miksuje dwa sample tworząc trzeci, wedle wzoru: SAMPLE XX + SAMPLE YY = SAMPLE ZZ. Po wciśnięciu przycisku wprowadzamy numery sampli, które chcemy zmiksować oraz numer sampla, w którym ma się znaleźć rezultat końcowy.

1	SAMPLE EDITOR									
2	TRACK									CLIP
3	DELETE	KILL	MIX	ECHO	POS	0000	↑	↓		
4	EXCHGE	COPY	BOOST	FILTER	MOD	000	↑	↓		
5	FROM	00	↑	↓	X-FADE	BACKWD	CUTBEG	FU	FD	
6	TO	00	↑	↓	UPSAMP	DNSAMP	VOL	100%	↑	↓
7										

Edytor sampli

- **ECHO** – prosty efekt echa. W polu POS nnnn podajemy pozycję (bajt), stanowiący o czasie kolejnych powtórzeń echa. Parametr ten można ustawiać za pomocą myszki w oknie SAMPLERA (na ten temat będzie mowa później).
- **BOOST** – wzmocnienie wysokich tonów sampla (treble).
- **FILTER** – filtruje wysokie częstotliwości, pomocny np. przy usuwaniu szumów w samplach basowych.
- **MOD nnn (MODULATION)** – nadaje dzwiękowi efekt modulacji (częstotliwości). Parametrem nnn ustawiamy częstotliwość w zakresie od 0 do 127. Przytrzymanie lewego przycisku na tej liczbie, a następnie wciśnięcie prawego powoduje jej wyzerowanie.
- **X-FADE** – miksuje aktualnie wybrany sampel z jego odwróconą wersją (odgrywaną od tyłu).
- **BACKWD (BACKWARDS)** – odwraca sampel, który po tej operacji będzie odgrywany od tyłu.
- **CUTBEG (CUT BEGINNING)** – odcina od początku sampla liczbę bajtów określoną w polu POS nnnn.
- **FU (FADE UP)** – zastosuje efekt stopniowego narastania amplitudy sampla od początku do pozycji określonej w polu POS nnnn.
- **FD (FADE DOWN)** – zastosuje efekt stopniowego opadania amplitudy sampla od pozycji określonej w polu POS nnnn do końca.
- **UPSAMP (UPSAMPLE)** – zwiększa tonację sampla o jedną oktawę do góry. Obniża częstotliwość oraz długość sampla o połowę.
- **DNSAMP (DOWNSAMPLE)** – obniża tonację sampla o jedną oktawę w dół. Zwiększa częstotliwość oraz długość sampla dwukrotnie.
- **VOL nnn% (VOLUME)** – umożliwia zwiększenie lub zmniejszenie poziomu głośności sampla w zakresie od 0 do 999%.
- **CLIP/HALVE (przycisk cykliczny)** – i tu uwaga! W wersji 3.61/3.62 jest błąd w kodzie programu. W założeniu opcja ta ma wpływać na sposób działania funkcji miksovania dwóch sampli (patrz: MIX). CLIP – miksovanie odbywa się z pełną oryginalną głośnością obu sampli, co w pewnych przypadkach może powodować docelowo przesterysty (ang. clipping). HALVE – miksovanie odbywa się z 50% głośnością sampli, co pozwala z kolei na uniknięcie przesterowania sampla wynikowego. Błąd w używanej przeze mnie wersji Protrackera polega na tym, że co prawda technicznie wszystko działa z tym, że ustawienie CLIP powodowało efekt, jak przy ustawieniu HALVE i na odwrót.

CHORD EDITOR to edytor akordów, które możemy sobie tworzyć z sampli załadowanych do programu. Jego działanie polega na kilkukrotnym nałożeniu na siebie źródłowego sampla w różnych częstotliwościach odpowiadających nutom (od 2 do maksymalnie 7).

1	CHORD EDITOR									
2	FROM SAMPLE		NOTES:		HARD		HARD			
3	0001		↑ ↓		3		↑ ↓			
4	TO SAMPLE		ADJUST							
5	0002		↑ ↓		MAKE CHORD					
6	NOTE1	NOTE2	NOTES	NOTE4	NOTES	NOTES	NOTE			
7	C-1	C-1	C-1							

Edytor akordów

- **FROM SAMPLE nnnn** – numer instrumentu źródłowego, z którego chcemy utworzyć akord.
- **TO SAMPLE nnnn** – numer instrumentu, w którym pojawi się wynik naszych działań.
- **NOTES n** – określa z ilu nut będzie składał się akord.
- **ADJUST/NOADJUST** (przycisk cykliczny) – określa czy miksovanie akordu ma się odbywać ze zmniejszoną (NOADJUST) lub zwiększoną głośnością sampla źródłowego (ADJUST). Działanie tej opcji jest analogiczne do wyżej opisanego CLIP/HALVE z tą uwagą, że w tym przypadku działa prawidłowo.
- **MAKE CHORD** – tworzy akord na podstawie ustawionych parametrów i zapisuje jego rezultat w samplu o numerze nnnn ustawionym w polu „TO SAMPLE”.
- **NOTE1 do NOTE7** – wartości nut składających się na akord, np. C-3, D#-3, G-3 utworzy akord C-moll (minorowy), a C-3, E-3, G-3 utworzy akord C-dur (majorowy) itd.

Panel czwarty nazywany jest **FILTER EDITOR**, lecz jest on nieukończony w wersji 3.62 programu. Z kolei panel piąty jest w ogóle niewykorzystany, a więc od razu przechodzi do szóstego, który nazywa się **MOD2SAMP OPTIONS**. Jest to panel wbudowanego w Protrackera programiku *Mod2Smp*, służącego do miksovania fragmentów lub całych patternów do pojedynczego sampla. Jak wspomniałem przy omawianiu instalacji, do prawidłowego działania tego komponentu wymagana jest biblioteka *mod2smp.library* w systemowym katalogu *LIBS*. Jej brak zaowocuje zawieszeniem się Protrackera z chwilą rozpoczęcia miksovania.

- **NOTE** – do jakiej nuty (i odpowiadającej jej częstotliwości) ma być zmiksowany fragment patternu.
- **PATTERN nn** – numer patternu, który będziemy miksować.
- **STARTP. nn (START POSITION)** – pozycja początkowa fragmentu mikowanego patternu. Przyjmuje wartość od 00 do 63. UWAGA! Przy ustawianiu powyższego parametru należy brać pod uwagę, że długość sampla w Protrackerze to maksymalnie 64 kB. W przypadku ustawień powodujących wygenerowanie sampla o długości większej niż maksymalna, nasz miks zostanie obcięty.
- **ENDPOS. nn (END POSITION)** – pozycja końcowa fragmentu mikowanego patternu. Przyjmuje wartość od 00 do 63. Uwaga jak wyżej.
- **SAMPLE nn** – numer sampla wynikowego, w którym znajdzie się miks.



- **SPEED nnn** – szybkość użyta przy mikrowaniu. Działa tylko w przypadku, gdy w patternie nie ma wpisanej żadnej komendy sterującej szybkością odtwarzania.

Z pozostałych funkcji widocznych w panelu **Mod2Smp** działają jedynie **FILTER** i **BOOST** (tak samo jak w **SAMPLE EDITOR**). **BACKWARDS**, **MAXIMIZE**, **NORMALIZE**, **INTERPOL** oraz **GRAB** są nieaktywne.

Ostatni, siódmy panel to **SONG OPTIONS**. Podobnie jak kilka poprzednich jest on nieukończony w wersji 3.62 programu.

1	COMMENT	SONG OPTIONS
2	SONGNAME	-----
3	AUTHOR	UNNAMED AUTHOR-----
4	TIME	03-May-12 14:17 00:07
5	TRACKER	PT3.61 VERS. 0000
6	PATTERNS	00 TUNING CIA COMMENT
7	TRACKS	04 SPEED 0125 CREATE

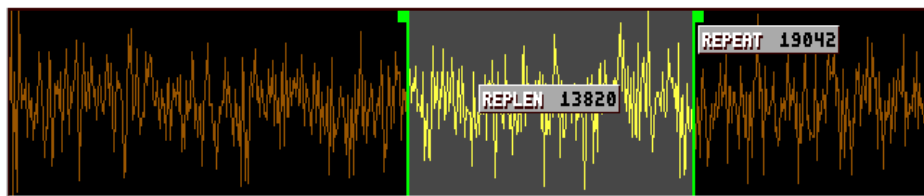
Podczas przygotowywania brzmień do wykorzystania w Protrackerze za pomocą innego programu czy komputera zwykle chcemy, aby każdy dźwięk nagrany np. na klawiaturze syntezatora odpowiadał takiej samej nucie wpisywanej w ścieżkę naszego modułu. Poniższa tabela przedstawia częstotliwości nut w formacie Protrackera, które przydadzą się przy konwersji i pomogą zachować zgodność oryginalnej tonacji.

mccnex/Wanted Team

dedykuję żonie Karolinie

Nuta	PAL	NTSC
C-1	4143 Hz	4181 Hz
C#1	4389 Hz	4430 Hz
D-1	4654 Hz	4697 Hz
D#1	4926 Hz	4971 Hz
E-1	5231 Hz	5279 Hz
F-1	5542 Hz	5593 Hz
F#1	5872 Hz	5926 Hz
G-1	6222 Hz	6279 Hz
G#1	6592 Hz	6653 Hz
A-1	6982 Hz	7046 Hz
A#1	7389 Hz	7457 Hz
B-1	7829 Hz	7901 Hz

Protracker w wersji 3.32



Nuta	PAL	NTSC
C-2	8287 Hz	8363 Hz
C#2	8779 Hz	8860 Hz
D-2	9309 Hz	9395 Hz
D#2	9852 Hz	9943 Hz
E-2	10462 Hz	10559 Hz
F-2	11084 Hz	11186 Hz
F#2	11744 Hz	11852 Hz
G-2	12445 Hz	12559 Hz
G#2	13185 Hz	13306 Hz
A-2	13964 Hz	14092 Hz
A#2	14778 Hz	14914 Hz
B-2	15694 Hz	15838 Hz

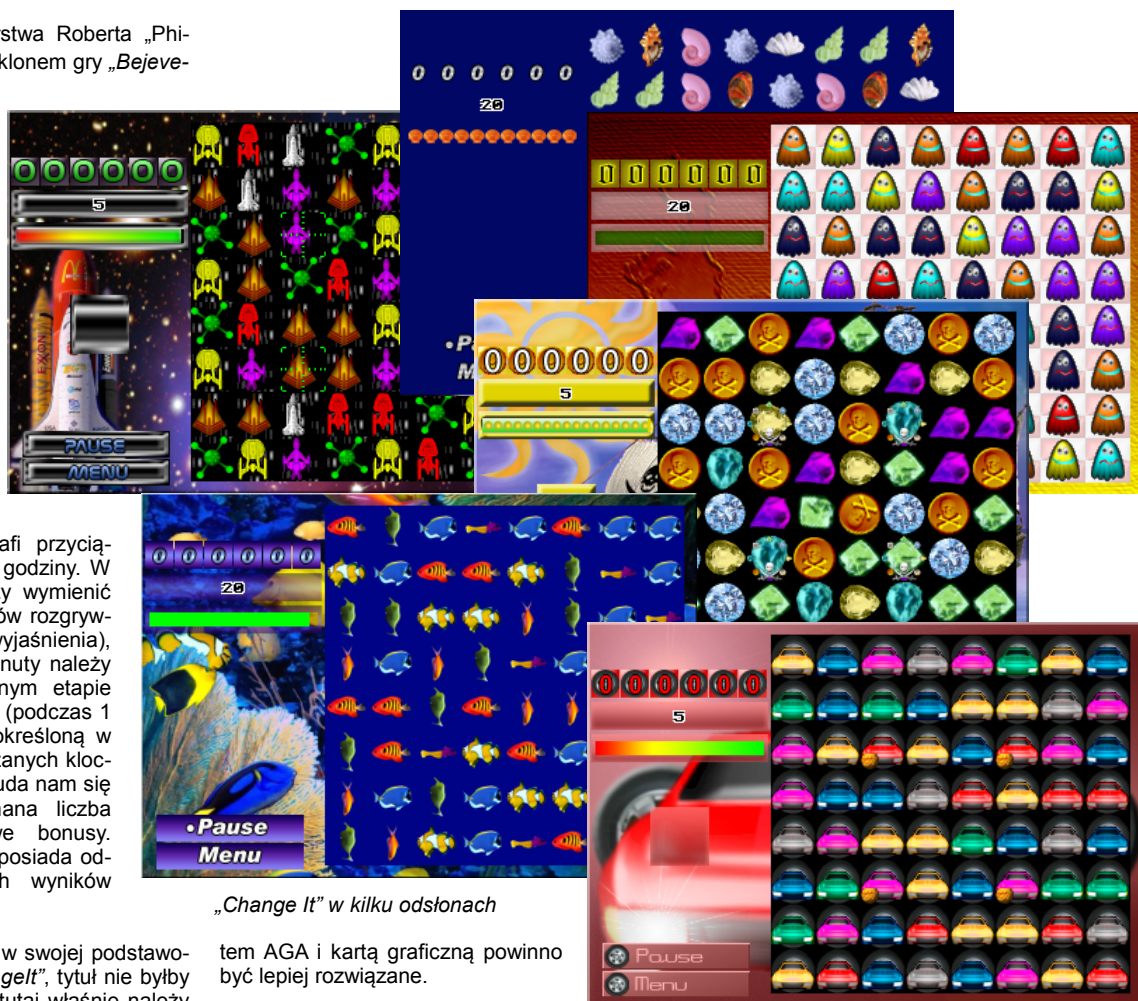
Nuta	PAL	NTSC
C-3	16574 Hz	16726 Hz
C#3	17558 Hz	17720 Hz
D-3	18667 Hz	18839 Hz
D#3	19704 Hz	19886 Hz
E-3	20864 Hz	21056 Hz
F-3	22168 Hz	22372 Hz
F#3	23489 Hz	23705 Hz
G-3	24803 Hz	25031 Hz
G#3	26273 Hz	26515 Hz
A-3	27928 Hz	28185 Hz
A#3	29557 Hz	29829 Hz
B-3	31388 Hz	31677 Hz

Tabela częstotliwości sampli odpowiadających nutom w Protrackerze



Changelit

„Changelit” to gra autorstwa Roberta „Phibrizzo” Krajcarza, będąca klonem gry „Bejeveled”. Na planszy rozrzucone są różnokolorowe elementy. Celem zabawy jest zamiana dwóch sąsiednich, aby uzyskać linię poziomą lub pionową składającą się z przynajmniej trzech takich samych klocków. Gra nie toczy się w nieskończoność – zadanie jakie przed nami postawiono, w zależności od trybu wybranej rozgrywki, musimy wykonać w określonym czasie. „Changelit” nie byłaby niczym nadzwyczajnym, gdyby nie pewne elementy sprawiające, że gra potrafi przyciągnąć do ekranu na długie godziny. W pierwszej kolejności należy wymienić już wspomniane kilka trybów rozgrywki: trening (nie wymaga wyjaśnienia), time attack (podczas 1 minuty należy zdobyć wymaganą w danym etapie liczbę punktów), brick hunt (podczas 1 minuty należy zniszczyć określoną w danym etapie liczbę wskazanych klocków). Od tego ile klocków uda nam się zniszczyć, zależy otrzymana liczba punktów oraz dodatkowe bonusy. Każdy z trybów rozgrywki posiada odrębną tabelę najlepszych wyników punktowych.



„Change It” w kilku odsłonach

Mając na względzie to, co w swojej podstawowej odsłonie oferuje „Changelit”, tytuł nie byłby niczym nadzwyczajnym. I tutaj właśnie należy wspomnieć o kolejnym elemencie, mającym wpływ na wydłużenie żywotności gry – skórki. Zabawę możemy toczyć na jednym z 16 zestawów tematycznych. Każdy z nich jest inny, wyróżnia się zarówno indywidualną grafiką elementów, ekranami menu oraz efektami dźwiękowymi. Niektóre skórki przygotowane są w taki sposób, że można odnieść wrażenie grania w całkiem inną grę. Możemy zamienić domyślnie stosowane w tego typu grach diamenty i kryształy na liście, wielkanocne jajka, duszki (to ukłon w stronę uczestników cyklicznych spotkań AmiZaduszki), kule, pirackie artefakty, kamienie, jabłka, muszle, samochody. Obiektywnie jednak patrząc, niektóre z tych skór w mojej ocenie należałoby poprawić, gdyż uważam, że ich kolorystyka nie została do końca korzystnie dobrana, sprawiając że gracz myli się lub nie zauważa grup kolorystycznych. Przyszedłoby się nieco więcej kontrastu lub różnicowania, co znacznie ułatwiłoby zabawę. Oczywiście jest to uwaga tylko do niektórych skór i ogólnie jakoś nie wpływa ujemnie na całokształt zabawy, jaką oferuje gra – zawsze skórkę można sobie zmienić lub narysować samemu, z czego autor na pewno bardzo by się ucieszył. Z rzeczy, które chciałbym, aby zostały bezwzględnie poprawione jest zaplanowanie nad chaosem w folderze z grą – rozróżnienie wersji do uruchomienia pod chipse-

tem AGA i kartą graficzną powinno być lepiej rozwiązane.

Gra została napisana w pełni z wytycznymi systemowymi. Uruchamia się na ekranie Workbenchu zarówno na Amigach z chipsetem AGA, jak również na kartach graficznych (oferując efekt przezroczystości czy też dodatkowe efekty graficzne). Wymagania są, można rzec, podstawowe jak na grę działającą „w okienku”: trochę pamięci na otwarcie okna i wczytanie niezbędnych grafik (datatyp PNG), AHI (do obsługi dźwięku). Autor równoległe przygotowuje kompilacje swojej gry dla AmigaOS 3.x, AmigaOS 4.x, AROS-a oraz MorphOS-a, dzięki czemu każdy może czerpać z niej przyjemność.

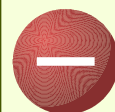
Zachęcam do zapoznania się z tym tytułem i doceniania autora oraz tych wszystkich osób, które mu pomogły, że możemy się z niej cieszyć. Phibrizzo to utalentowany programista, który doprowadza swoje projekty do końca i nie powinien w żadnym razie się ich wstydzić. „Changelit” jest przykładem właśnie takiego tytułu – prostej, nieskomplikowanej gry, należyście przygotowanej i oferującej dokładnie to, czego się po niej można spodziewać.

Sebastian Rosa

Autor: Robert „Phibrizzo” Krajcarz
Strona gry: <http://changelit.ppa.pl>
Ostatnia dostępna wersja: 0.9 (wg stanu na 31.10.2013)



- oklepany pomysł przygotowany z odrobiną świeżości,
- produkt w pełni systemowy,
- duża ilość i zróżnicowanie skór graficznych, co skutecznie wpływa na żywotność gry,
- kilka różnych trybów rozgrywki.



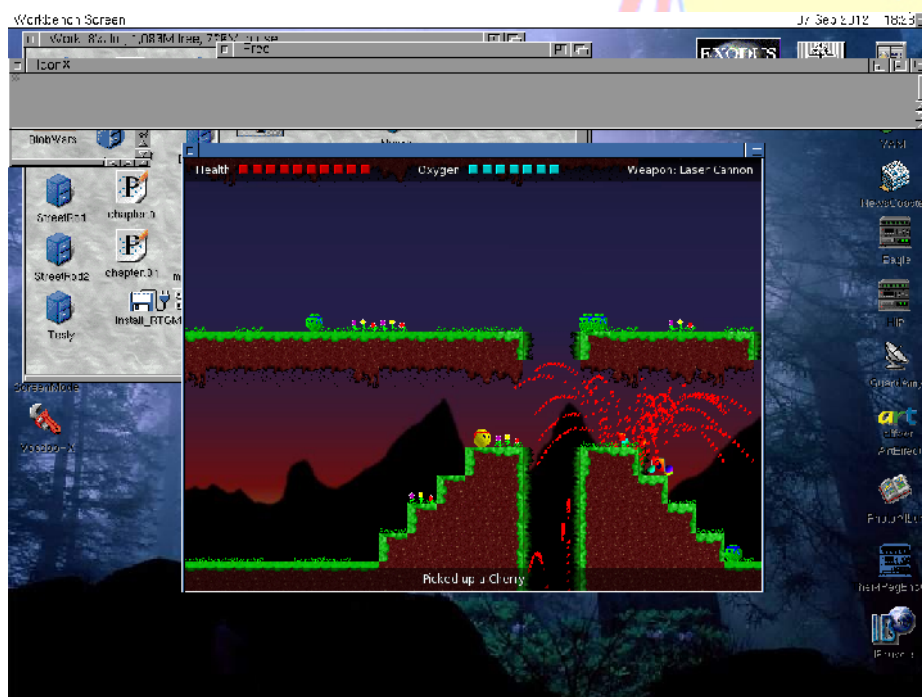
- nie do końca dobrany kontrast elementów gry na niektórych skórkach,
- trochę chaosu w folderze z grą (można było lepiej rozwiązać wybór uruchamianej wersji gry).



Metal Blob Solid: Blob Wars

„**Metal Blob Solid**” to pierwsza gra z serii „**Blob Wars**” przygotowana przez grupę Parallel Realities. Jest to platformówka połączona ze strzelaniną. Wcielamy się w niej w postać Boba – sprżystej, żółtej kulki, która musi uratować swoich pobratymców uwieczonych przez wrogie oddziały najeźdźców z kosmosu, którzy pod wodzą Metal Bloba Galdova opanowały świat. Bob do eksterminacji przeciwnika wykorzystuje broń, która pozostaje po niektórych uśmierconych wrogach. Wachlarz jest dosyć szeroki: od pistoletu, przez karabin maszynowy, karabin strzelający w trzech kierunkach, działo laserowe aż po granaty. Intrigującym elementem gry jest to, że nie polega ona wyłącznie na zabiciu określonej liczby przeciwników i uratowaniu określonej liczby blobów. Każda mapa posiada dodatkowe misje, które należy wykonać, aby móc zagrać w kolejną. Bardzo się od siebie różnią i polegają na odnajdywaniu zaginionych przedmiotów (kluczy, komunikatorów, dynamitu), niszczeniu generatorów, odnajdywaniu ukrytych przejść. Map do przejścia jest sporo i zgodnie z zapewnieniami autorów gra oferuje ponad 9 godzin zabawy (w trakcie gry liczony jest czas). Ciekawym pojęciem jest to, że map nie musimy pokonywać w ustalonej kolejności – możemy czynić to dowolnie, w miarę możliwości jak są udostępnione. Sprawia to, że w grze pojawia się pewien pierwiastek nieliniowości.

Wykonanie gry jest poprawne. Trochę mogą razić puste przestrzenie i monotonne podkłady tła występujące w trakcie rozgrywki, lecz prawdę powiedziawszy więcej wcale nie potrzeba, aby czerpać z gry przyjemność. Te niedobory są trochę rekompensowane przez zróżnicowanie grafiki między mapami (każda wyspa, a w związku z tym dostępne na niej mapy, charakteryzuje się innym wykonaniem).



Ekrazy przerywnikowe oraz menu czy też mapa terenu wykonane są dobrze i nie ma na co narzekać. Przyczepić się trzeba do małej różnorodności przeciwników – występują tylko zasymilowane na nasz wziderunek kulki oraz unoszące się w powietrzu droidy. To trochę mało. W trakcie rozgrywki towarzyszą nam dobrze wykonane efekty dźwiękowe oraz atmosferycznie przgrzająca muzyka.

Szczerze przyznam, że wykonaniem gry, jej

prędkością działania oraz możliwościami jestem pozytywnie zaskoczony. Mając w pamięci fakt, że gra jest napisana pod SDL zazwyczaj spodziewałem się kiepskiej i wolnej gry zręcznościowej lub kolejnej gry logicznej będącej nowym wcieleniem „Tetrisa”. Gra może rodzić skojarzenia z „Fury of the Furries”, przy czym większy nacisk położony jest na część strzelecką niż aspekt logiczny. Nie jest wolna od wad czy irytujących rozwiązań (brak możliwości zmiany broni, niezwykła precyzja przy niszczeniu fragmentów ścian, po których trzeba się wspinać), ale posiada również kilka elementów uprzyjemniających (krwawący przeciwnicy, na których możemy uskuteczyć combosy) i ułatwiających zabawę (np. mapa z lokalizatorem przeciwników, bloby do uratowania itp.). Ogólnie rzecz biorąc, gra prezentuje całkiem równy poziom, zarówno od strony wykonania, jak i oferowanej formy zabicia wolnego czasu. Dzięki mnogości i różnorodności misji potrafi przyciągnąć do ekranu monitora na dosyć długo. Czy od darmowej gry można chciał czego więcej?

Gra do działania wymaga karty graficznej i biblioteki SDL. Posiada możliwość uruchomienia zarówno w trybie pełnoekranowym, jak i w okienku.

Sebastian Rosa

Author: Parallel Realities
Strona gry: <http://sourceforge.net/projects/blobwars>
Wersja dla AmigaOS 3: Lorence Lombardo
AmigaOS 3:
http://aminet.net/game/jump/BlobWars_68k.lha
AmigaOS 4:
<http://os4depot.net/share/game/platform/blobwars.lha>
AROS:
<http://aminet.net/game/actio/blobwars.i386-aros.zip>
MorphOS:
http://aminet.net/game/actio/blobwars-1.05_mos.lha



The Dig

Nazywam się Low, Boston Low. Nie wiem co się stanie, gdy wejdę do portalu, więc zostawiam tę informację dla wszystkich, którzy dotrą tu po mnie. Kilka dni temu wyruszyliśmy z czterema innymi członkami załogi, aby ratować naszą planetę przed zagrożeniem z kosmosu – planetoidą Atylla, znajdującą się na kursie kolizyjnym z Ziemią. Nasz prom kosmiczny – Atlantis – w ciągu kilku godzin wszedł na orbitę okołozemską i osiągnął odległość operacyjną. Do momentu podłożenia dwóch ładunków nuklearnych wszystko szło jak po sznurku – dopiero potem okazało się, że Atylla to prawdziwy hun. Ale po kolei...

Fabula

„The Dig” (co można tłumaczyć jako „wykopalisko”) to kolejna przygodówka od LucasArts, którą bez problemu uruchomimy i ukończymy przy pomocy ScummVM. Tym razem mamy do czynienia z zaadaptowanym

scenariuszem filmowym – autorstwa podobno samego Stevena Spielberga, z czasów gdy kręcił swoje „Amazing Stories” („Niesamowite historie”). Dość powiedzieć, że po – przyznajcie sami – sztamkowym wstępie (ile już razy Ziemi zagrażały przeróżne asteroidy, planetoidy, meteoroidy i kometoidy?) nasze zuchy znajdują się niespodziewanie na nieznannej, obcej planecie, która jeśli nie jest, to z pewnością kiedyś była zamieszkała. Zdani tylko na siebie i



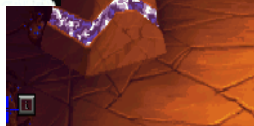
rozwickać piętrzące się przed nimi zagadki, uniknąć groźących niebezpieczeństw i powrócić – o ile to możliwe – w komplecie do domu.

Oprawa

W chronologii gier działających pod kontrolą interfejsu Scumm „The Dig” plasuje się na przedostatniej pozycji, pomiędzy „Full Throttle” a „The Curse of Monkey Island”. Niestety, nie mamy tu do dyspozycji ani wysokiej rozdziel-



swoje umiejętności (jedno to stary wyjadacz – astronauta i wojskowy na dokładkę, drugie – to dziennikarka z uzdolnieniami lingwistycznymi, a trzecie – to geolog-archeolog z przerostem ambicji) muszą



czości znanej z tej drugiej, ani też nowego interfejsu sterowania, który pojawił się już w pierwszej z nich. Prawdopodobnie jest to związane z faktem, że gra powstawała długo i przeszła po drodze sporą rekonstrukcję. W kwestii samej oprawy więcej podobieństw dostrzegamy z „Full Throttle” – już w pierwszych lokacjach widzimy połączenie rysowanej grafiki postaci oraz prerenderowanych fragmentów scenarii i przerywników filmowych. Szkoda tylko, że podobieństwa na tym się kończą i bohaterowie gry w przerywnikach filmowych narysowani już są w sposób „uproszczony” – zamiast pełnokrwistych, świetnie animowanych postaci mamy trochę dziwne i uproszczone, sprawiające wrażenie wyciętych z kartonu, płaskich plansz. Trochę rekompensuje to animacja w trakcie gry – choć w niskiej rozdzielczości, to jednak wszystkie ruchy są płynne i bardzo dobrze dopracowane. Muzyka z kolei jest bardzo przyjemna i dobrze wprowadza w

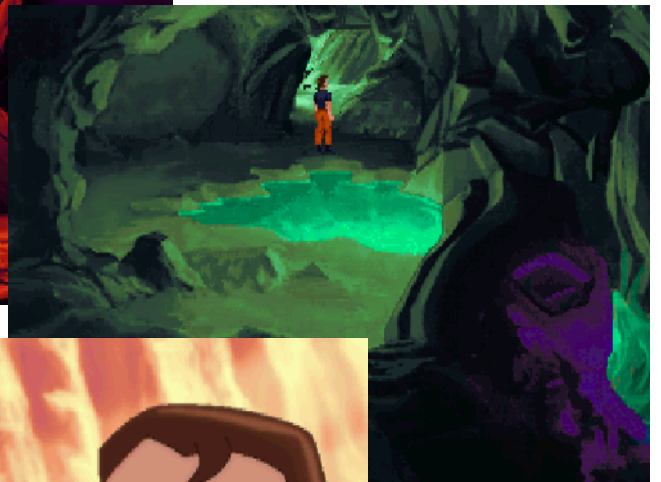


nastrój nieznannej, tajemniczej planety (ukazał się osobny soundtrack, choć szczerze mówiąc – nie wiem w jakim celu ktoś miałby po prostu słuchać tej muzyki, chyba że jako płyty relaksacyjnej). Wszystkie dialogi są mówione (gra ukazała się wyłącznie na nośniku CD), a podłożeniem głosów zajęli się prawdziwi aktorzy (ciekawostka – Bostonowi głosy użycza znany z filmu „Terminator 2” Robert Patrick). Do udźwiękowienia nie mam zastrzeżeń.

Rozgrywka

Jak już wcześniej wspominałem, mamy do dyspozycji aż trzech bohaterów (plus dwójka, z którą możemy porozmawiać przy pomocy komunikatora w pierwszej części gry), bezpośrednio sterujemy jednak tylko komandorem Lowem. Zresztą, nasza ekipa rozpada się prędej niż Drużyna Pierścienia i wkrótce, poza epizodami i

rych „często-gęsto” brak jakichkolwiek wskazówek. Nastawcie się też, że sporo się w „The Dig” nachodzi, dlatego warto wyrobić sobie nawyk korzystania z dwukliku (szybkie przejście pomiędzy znanymi lokacjami), gdzie tylko się da, a gdzie się nie da – przerywania animacji klawiszem Escape (obsługiwane, na przykład,



ewentualnymi konsultacjami przez Pen Ultimate (tak w latach 90-tych wyobrażano sobie chyba smartfony), przyjdzie nam raczej pracować samemu. A nad czym pracować? Mimo, że interfejs został maksymalnie uproszczony (lewy przycisk to interakcja, prawy to inwentarz, temat dialogu wybieramy klikając na obrazek), a przedmiotów do zebrania nie jest wcale dużo, to czeka nas porcja solidnego główkowania (szczególnie „podoba mi się” fragment, gdy sterując na ślepo poleceniami z konsoli mamy poprawić przy pomocy zdalnie sterowanego robota ustawienie soczewki i przywrócić zasilanie – mistrzostwo świata. Jeśli ktoś przeszedł ten fragment bez uciekania się do rozwiązania, to chylę czoła). Cała gra to w dużej mierze podobne łamigłówki, do któ-

jak nasza kula transportowa przetacza się majestatycznie podwodnym tunelem za pierwszym razem może być urzekające, za piątym – lekko nudne, ale za dwudziestym – zdecydowanie irytujące).

Kilka słów na koniec

„The Dig” to z pewnością gra warta bliższego poznania. Oprawa jest całkiem przyzwoita, rzekłbym nawet – dość dobra, scenariusz (mimo słabego moim zdaniem zakończenia) – interesujący, no i z pewnością nie ukończycie jej zbyt szybko (choć może po prostu rzucicie ją w diabły utknąwszy po raz kolejny w martwym punkcie). Poza tym – to kawał historii z nieistniejącego już studia LucasArts. Dodatkowo

wo, jako bonus dla wszystkich, którzy alergicznie podchodzą do języków obcych – można ją w prosty sposób „spoloniżować”, podmieniając raptem dwa pliki. Odnosiń do spolszczenia (nawet niezłego, choć zdarzają się orty i zabawne literówki) zamieszczam w ramce.

Konrad Czuba

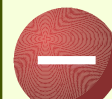


Spolszczenie gry:

http://grajpopolsku.pl/downloads.php?cat_id=1&download_id=345



- ciekawy (choć początkiem lekko wyświechtany) scenariusz,
- przyzwoita oprawa audiowizualna,
- łatwy do opanowania interfejs.



- dziwny styl rysowania postaci w przerywnikach filmowych,
- absurdalny poziom trudności niektórych zagadek,
- gra sprawia wrażenie ukończonej trochę „na siłę”.



Nasza pierwsza gra

Kurs programowania z wykorzystaniem Visual Studio 2005 Express Edition i VBCC – część 6

Zanim rozpoczniemy szóstą odsłonę naszego kursu tworzenia gry, poprawimy drobne błędy z poprzedniego odcinka – przynajmniej tam je znalazłem, więc niezwłocznie naprawiam. Przejdźmy do funkcji *initTiles* w pliku **tile.c**. Zawsze zwracamy tam wartość *RT_OK*, co jest błędem, bo gdy funkcja – mówiąc kolokwialnie – się wysypie, to i tak powróci wyżej z poprawną wartością, co jest niedopuszczalne. Szybciutko należy to zmienić na *iResult*. W pliku **fileIO.c** w funkcji *WriteFile* odnajdziemy błędnie zapisany kod powrotu. W przypadku gdy plik nie zostanie otwarty, funkcja powinna zwracać kod *RT_FAILED_WRITE_FILE*. O zgrozo – teje wartości nie odnajdziemy w pliku **types.h**, gdzie zwyczajowo tam się jej spodziewamy i należy ją dopisać w *enum ReturnCodes*. Także tam dopisujemy *RT_FAILED_TIMER_MSGPORT*, *RT_FAILED_TIMER_DEVICE*, *RT_FAILED_TIMER_IOREQ*, które przydadzą się w nowej odsłonie modułu *timer*. Biję się w piersi, bo tam też były co najmniej dziwne wartości zwracane w przypadku niepowodzenia. Oprócz tego w pliku **types.h** dopiszemy makro, które znakomicie ułatwi nam życie z wielkością tablicy – przestaniemy się martwić jej rozmiarem. Dopisujemy linię.

```
#define ARRAY_SIZE(array) sizeof(array) / sizeof(array[0])
```

Makro to znajduje zastosowanie wszędzie tam, gdzie potrzebujemy przebiec całą tablicę i znudziło nam się ręczne wklepianie jej rozmiaru. W pojedynczym przypadku użycia tablicy wydawać by się mogło, że lepiej jest napisać magiczną liczbę, oznaczającą ilość elementów tablicy. To niestety jest bardzo złudne i prędko można się przekonać, że program działa niewłaściwie, gdy tylko przyjdzie nam zmienić ilość elementów tablicy i zapomnimy zaktualizować ową magiczną liczbę. Na podstawie własnego doświadczenia powiem, że z całą pewnością zapomnimy. Poniżej prosty przykład użycia makra.

```
int i;
UBYTE tab[] = { 0,1,2,3,5,6,7,8,10 };
for (i = 0; i < ARRAY_SIZE(tab); ++i)
{
    tab[i] = 0;
}
```

Proste ćwiczenie dla czytelnika, to policzenie rozmiaru tej tablicy – najlepiej dwa razy, aby przekonać się, że lepiej używać makra.

W tym odcinku będziemy dalej ulepszać grę i dodamy krótką demonstrację, przy czym nie mam tu na myśli wypuszczenia gry w wersji demo. To o co chodzi? A to, że często w grach można zauważyć, że jak nie uruchamiamy gry przez jakiś czas, to ona „przechodzi się sama” pokazując nam, jak mamy grać, zazwyczaj pokazując odległe plansze. Wszystko po to, aby nas zachęcić do grania, gdyż po samym tytule czasami ciężko dojść o co w grze chodzi i jakiego typu jest to gra. Problem dodania demonstracji gry można rozwiązać brutalnie, żeby nie powiedzieć banalnie – wystarczy grać

na emulatorze i zrobić filmik, po czym załadować go i odtworzyć. Pomijam tu kwestie formatu filmu i potrzebnej pamięci. Ta brutalna i naiwna solucja, w naszym przypadku, nie wchodzi w grę. Śmiesznie by wyglądała gra, która jest plikiem rzędu 20 kB, a do niej dołączony plik z animacją zajmującą lekko ponad 20 MB. W takim razie jakie są inne rozwiązania tego problemu? Mi do głowy przyszło jedno i myślę, że jest dobre. W grze poruszamy się statkiem kosmicznym za pomocą joysticka bądź klawiatury, a ruchy przechowujemy w zmiennych boolowskich. A gdybyśmy tak zasymulowali ruchy, jakie wykonuje gracz, zmieniając wartości zmiennych odpowiedzialnych za ruch naszego bohatera, to wtedy rozwiązałibyśmy nasz problem. Na pewno będziemy potrzebowali tablicę, która te ruchy (lewo, prawo, góra, dół) będzie zawierać. Ponadto każdy taki ruch musi następować w określonym czasie. A zatem będziemy przechowywać w tablicy zarówno ruch, jak i czas, po którym należy wykonać następną czyn-

ność. Z pomocą nam przyjdzie taka struktura:

```
typedef struct
{
    UBYTE dir;
    UBYTE secs;
} DemoDir;
```

Aby ułatwić sobie zadanie, skorzystaliśmy z dobrodziejstwa *typedef* i w ten oto sposób wystarczy pisać tylko *DemoDir*, zamiast *struct DemoDir*. Sprytny czytelnik z pewnością zapyta o sens budowania takiej struktury, skoro oba typy są takie same. Ja zrobiłem to ze względu na czytelność kodu, a jeśli ktoś chce, może użyć zwyczajnej tablicy jednowymiarowej. Pozostawiam to jako ćwiczenie dla chętnych.

Skoro już wiemy, co mamy przechowywać w tablicy, to pozostaje nam sama tablica. Użyłem

```
#define Right 1
#define Left 2
#define Up 3
#define Down 4
#define Fire 5

static DemoDir DemoDirTable[] =
{
    {Left, 3},
    {Left, 1},
    {Down, 1},
    {Right, 1},
    {Right, 4},
    {Down, 1},
    {Fire, 1},
};
```

wioduce zakończenie demonstracji i powrót do ekranu tytułowego. Powyższą tablicę należy czytać w następujący sposób: ruszamy joystickiem w „Lewo”, czekamy 3 sekundy, znowu ruch w „Lewo” i czekamy sekundę i tak dalej. Odliczanie czasu można zrealizować za pomocą *timer.device* (pliki **timer.c** i **timer.h** w naszym projekcie), tyle że będzie to wymagało większych zmian w **timer.c** i **timer.h**. Zabieramy się do pracy.

Na początku dwie pomocne konstrukcje – w sumie ułatwiające pisanie. Do tego dorzucimy nową strukturę, trzymającą potrzebne wskaźniki do sprawnego używania *timer device*.

```
typedef struct timerequest TimeReq;
typedef struct IOrequest ReqIO;
typedef struct
{
    struct MsgPort* port;
    TimeReq* request;
} Tim;
```

Przedstawię teraz zmienione dwie główne funkcje, które są lokalne w module *timer*. Są to: *init* i *kill*, które odpowiadają za inicjalizację i zwolnienie zasobów.

```
static int init(Tim* t)
{
    t->port = CreateMsgPort();
    if (0 == t->port)
        return RT_FAILED_TIMER_MSGPORT;
    t->request = (TimeReq*)CreateIORequest(t->port, sizeof(TimeReq));
    if (0 == t->request)
        return RT_FAILED_TIMER_IOREQ;
    BYTE error = OpenDevice(TIMERNAME, UNIT_VBLANK, (ReqIO*)t->request, 0);
    if (0 != error)
        return RT_FAILED_TIMER_DEVICE;
    t->request->tr_node.io_Message.mn_Node.ln_Type = NT_UNKNOWN;
    t->request->tr_node.io_Command = TR_ADDREQUEST;
    return RT_OK;
}
```

Najpierw tworzymy port, na którym będziemy nasłuchiwać wiadomości z device'a. Następnie alokujemy strukturę *timerequest*, a dalej otwieramy device. Nie będę się rozpisywał, bo temat już jest znany z wcześniejszych odcinków. Nowością jest ustawienie typu noda w wiadomości na nieznaną (*NT_UNKNOWN*). Dzięki takiemu zabiegowi będziemy mogli bez problemu dowiedzieć się w funkcji *kill* czy request został wysłany i przedsięwziąć odpowiednie kroki typu przerwanie requesta. Po krótko teraz o funkcji zwalniającej użyte zasoby.

```
static void kill(Tim* t)
{
    if (t->request)
    {
        if (NT_UNKNOWN != t->request->tr_node.io_Message.mn_Node.ln_Type)
        {
            AbortIO((ReqIO*)t->request);
            WaitIO((ReqIO*)t->request);
            CloseDevice((ReqIO*)t->request);
            DeleteIORequest(t->request);
        }
        if (t->port)
        {
            DeleteMsgPort(t->port);
        }
    }
}
```

Na początku robimy użytek z ustawienia typu noda na *NT_UNKNOWN* i jeśli request został wysłany, to typ noda został zmieniony i spokojnie możemy użyć *AbortIO* i *WaitIO*. Następnie zamykamy device, zwalniamy strukturę *timerequest* i usuwamy port. W związku z tym że moduł *timer* (**timer.c** i **timer.h**) będzie odpowiadał zarówno za odmierzenie czasu w grze co 1/50 sekundy, jak i będzie liczył czas pomiędzy ruchami w demonstracji gry, to będziemy potrzebowali dwóch struktur *Tim* i czterech funkcji odpowiedzialnych za inicjalizację i zwalnianie. W przypadku pierwszego odmierzenia wyglądać to będzie tak, jak na listingu, który znajduje się na następnej stronie.



```

int mainTimer;
int initMainTimer(void)
{
    int nResult = init(&mainTimer);
    g_nMainTimerSig = 1L << mainTimer.port->mp_SigBit;
    if (RT_OK == nResult)
    {
        mainTimer.request->tr_time.tv_secs = 0;
        mainTimer.request->tr_time.tv_micro = 20000;
        SendIO((ReqIO*)&mainTimer.request);
    }
    return nResult;
}
void killMainTimer(void)
{
    kill(&mainTimer);
}
    
```

Mamy tu *mainTimer*, który będzie trzymał port i *timerequest*, aby móc odmierzać co 1/50 sekundy. Tutaj podczas inicjalizacji od razu wysyłamy *requesta*, dzięki czemu główna pętla będzie działała tak, jak poprzednio. Uwolnienie spowoduje się do wywołania wcześniej opisanej funkcji. Oprócz tego mamy funkcję *signalsMainTimer*, którą będziemy wywoływać w **boxo.c**.

```

void signalsMainTimer(void)
{
    while (TRUE)
    {
        if (0 == GetMsg(mainTimer.port))
        {
            break;
        }
    }
    mainTimer.request->tr_time.tv_secs = 0;
    mainTimer.request->tr_time.tv_micro = 20000;
    SendIO((ReqIO*)&mainTimer.request);
}
    
```

W tym przypadku najpierw odbieramy wszystkie wiadomości z naszego portu i wysyłamy ponownie *timerequest*. Tak na marginesie wytłumaczę, dlaczego tutaj stosuję pętlę *while(TRUE)*. Ważny czytelnik powie, że przecież mogłem napisać to o wiele prościej pisząc.

```

while (GetMsg(mainTimer.port))
; /* nic nie rob w pętli while*/
    
```

Muszę przyznać, że początkowo tak sam robiłem, ale po wygenerowaniu kodu źródłowego okazało się, że VBCC (z włączonymi optymalizacjami) generuje gorszy kod niż w przypadku, gdy używamy konstrukcję *while(TRUE)*. Gorszy w tym sensie, że jest go więcej i jest powolniejszy, co poniekąd wynika ze specyfikacji instrukcji *while*. Nie oznacza to jednak, aby wszystkie pętle zamieniać na *while(TRUE)* – trzeba to empirycznie zbadać dla każdego przypadku. Dlatego nie powinno się optymalizować za wcześnie, bo może się okazać, że po poprawkach kodu, poprzednia wersja była o wiele lepsza. To taka mała dygresja, a my wracamy do głównego wątku.

Teraz czas na funkcję *init* dla odliczania czasu w demonstracji gry. Funkcję *kill* pomijamy, bo wygląda bardzo podobnie, jak wyżej.

```

int initDemoTimer(void)
{
    int nResult = init(&demoTimer);
    g_nDemoTimerSig = 1L << demoTimer.port->mp_SigBit;
    return nResult;
}
    
```

Od razu rzuca się w oczy brak wysłania *requesta*, ale po chwili zastanowienia wszystko jest jasne, gdyż wysłanie *requesta* zostanie wywołane, gdy uruchomione zostanie demo gry. Funkcję *signalsDemoTimer* będziemy wywoływać w **demo.c**.

```

void signalsDemoTimer(ULONG secs)
{
    while (TRUE)
    {
        if (0 == GetMsg(demoTimer.port))
        {
            break;
        }
    }
    demoTimer.request->tr_time.tv_secs = secs;
    demoTimer.request->tr_time.tv_micro = 0;
    SendIO((ReqIO*)&demoTimer.request);
}
    
```

Oczywistą konsekwencją zmian w module *timer* będzie zmiana w pliku **boxo.c** odpowiadającym za inicjalizację naszej gry. W funkcji *init* dodajemy *initMainTimer()* i *initDemoTimer()* tak, jak jest to czynione dla innych funkcji inicjujących. Bardzo podobnie robimy w *close()*. Pętla główna, czyli *loop* też ulegnie zmianie, a kilka słów wyjaśnienia wymaga blok kodu:

```

if (g_bDemoMode)
{
    if (signals & g_nDemoTimerSig)
    {
        DemoRun();
    }
}
    
```

Dopiero gdy zmienna *g_bDemoMode* jest ustawiona na wartość *TRUE*, następuje sprawdzenie czy przyszedł sygnał z *demoTimer* i następuje wywołanie funkcji sterującej demem gry. Sama funkcja *DemoRun* została zaszyta w nowym module *demo* (**demo.c** i **demo.h**). Zwyczajowo tworzymy dwa nowe pliki **demo.c** i **demo.h**, po czym modyfikujemy **makefile.mak** dodając odpowiednie formy dla zmiennych *OBJ* i *LINKOBJ*. Tam też wstawiamy informacje dla kompilatora o sposobie przetwarzania tychże plików.

```

OBJ = $(OBJDIR)\boxo.o \
      $(OBJDIR)\sound.o \
      $(OBJDIR)\demo.o

LINKOBJ = $(OBJDIR)\boxo.o \
          $(OBJDIR)\sound.o \
          $(OBJDIR)\demo.o

...

$(OBJDIR)/demo.o: $(SRCDIR)/demo.c
$(CC) $(CFLAGS) $(SRCDIR)/demo.c -o $(OBJDIR)/demo.o
    
```

Plik nagłówkowy zawiera deklarację wyżej wymienionej struktury *DemoDir* i deklarację funkcji widocznej na zewnątrz – *DemoRun()*. W **demo.c** umieszczamy tablicę zawierającą ruch wraz z czasami i ciało funkcji *DemoRun*.

```

static UBYTE cnt = 0;
void DemoRun(void)
{
    if (cnt >= ARRAY_SIZE(DemoDirTable))
    {
        cnt = 0;
    }
    UBYTE dir = DemoDirTable[cnt].dir;
    if (dir == Right)
    {
        g_bRight = TRUE;
    }
    else if (dir == Left)
    {
        g_bLeft = TRUE;
    }
    else if (dir == Down)
    {
        g_bDown = TRUE;
    }
    else if (dir == Up)
    {
        g_bUp = TRUE;
    }
    else if (dir == Fire)
    {
        g_pFnc = &Title;
    }
    signalsDemoTimer((ULONG)DemoDirTable[cnt].secs);
    cnt++;
}
    
```

Sama funkcja jest mało skomplikowana. Pierwszy *if* służy do zapętlenia tablicy ruchów, a następne gniazdo „*ifów*” przekształca wartości z tablicy na ruchy w grze. Ostatnie przekształcenie przycisku spowoduje powrót do ekranu tytułowego.

Do omówienia pozostały nam zmiany w module *game*. Z nowych rzeczy znajdziemy tam licznik, który jest zerowany w funkcji *Title*, a uruchomienie demo gry następuje w funkcji *titleLoop*. Jak można się spodziewać demo odpalamy tylko wtedy, gdy licznik spełnia warunek i nie jesteśmy w trybie demo. pokazujemy na ekranie krótki komunikat, że będzie demo i wywołujemy funkcję odpowiedzialną za demo gry. Zmieniamy także wskaźnik tak, aby gra uruchomiła się w trybie demo.

```

cnt++;
if (cnt == 125 && g_bDemoMode == FALSE)
{
    PrintTxt("demo", 128,32);
    cnt = 0;
    g_bDemoMode = TRUE;
    DemoRun();
    g_pFnc = &NewDemoGame;
}
    
```

Kod odpowiedzialny za uruchomienie demo gry pozwala nam na modyfikację planszy startowej. Oczywiście wtedy musimy zmienić odpowiednio tablicę ruchów. Można też zwiększyć atrakcyjność demo dodając losowy wybór planszy. Sama funkcja jest bardzo prosta i nie wymaga większego komentarza – najważniejszy jest numer planszy, który został ustawiony na pierwsze etapy.

```

static void NewDemoGame(void)
{
    CleanLevel();
    g_nLives = 1;
    g_nLvNumber = 0;
    g_pFnc = &NextLevel;
}
    
```

Warto dodać informację, że gra uruchomiła się w trybie demo. Najlepiej posłużyć się tym, co już mamy, czyli użyć funkcji *PrintTxt* pokazującej tekst na ekranie. Najważniejsze to użyć jej w takim miejscu, aby napis nie był przysłaniany przez planszę bądź przez nasz statek kosmiczny. Innymi słowy - napis musi być na pierwszym planie. Zatem umieszczamy go na samym końcu funkcji *GameLoop*.

```

if (g_bDemoMode)
{
    PrintTxt("demo", 128,96);
}
    
```

Ciekawym urozmaicheniem może być dodanie migającego tekstu „demo”. To wiąże się z dodatkową zmienną odpowiadającą za miganie tekstu i trzeba zmienić bądź napisać drugą funkcję, która będzie zmyślała tekst. Minusem tego podejścia będzie to, że musimy umieścić tekst w takim miejscu, gdzie tło pod tekstem nie jest grafiką. Kompleksowym rozwiązaniem będzie zapamiętywanie tła pod tekstem i dlatego proponuję, aby czytelnik w ramach ćwiczenia zaimplementował podejście łatwiejsze. Jako wskazówkę dodam, że trzeba się przyrzec bliżej funkcji *BitMapRastPort*.

W tym odcinku to wszystko. Zachęcam do eksperymentów i zadawania pytań na forum PPA.

Asman

Cała paczka z kodem do pobrania z poniżej zamieszczonego odnośnika.

<http://www.ppa.pl/artykuly/download/ftp/boxo6.zip>



Adapter CF-IDE, czyli jak z karty CF zrobić dysk i zainstalować system

(poradnik w krokach – kilkuset)

Amiga bez dysku twardego to komputer w zasadzie tylko do gier – i to tych najprostszych. Dysk, jak wiadomo, podnosi komfort zabawy na naszym ulubionym komputerze i nie ma tu większego znaczenia czy jest to Amiga 600, czy też wypasiona Amiga 1200. Wadą naszych Amig jest to, że musimy kupić dysk 2.5" IDE, które nie dość, że są drogie, to coraz trudniejsze do dostania. Idealnym rozwiązaniem jest adapter kart Compact Flash – IDE. Takich adapterów na rynku jest wiele – w praktycznie każdym sklepie, który jeszcze zajmuje się Amigą można znaleźć taką przelotkę. Ja jednak zaryzykowałem i kupiłem w normalnym sklepie z częściami PC – przelotka kosztowała mnie niespełna 25 zł. Zaletą tego rozwiązania była dostępność – kartę miałem od razu, nie musiałem czekać kilku dni na przesłanie paczki od sprzedawcy z portalu aukcyjnego czy innego sklepu internetowego. Dodatkowo w razie reklamacji nie muszę ponosić kosztów wysyłki do sprzedawcy – wystarczy przejść się do sklepu. Przelotka wyposażona jest w tzw. śledzia do montażu w obudowie tower, którego jednak można odkręcić od płytki. Tak też zrobiłem, dzięki czemu adapter mieści się bez najmniejszego problemu w obudowie Amigi. Dodatkowym „bajerem” są diody LED, w które wyposażony jest adapter – po włączeniu komputera, wewnątrz Amigi bardzo ładnie świeci.

Podłączenie

Adapter możemy podłączyć do Amigi na dwa sposoby: za pomocą taśmy 2.5" (44 PIN) i z takiej taśmy podpinane jest też zasilanie albo – jeżeli mamy A1200 i Fast-ATA – normalną taśmą do dysków 3.5" (40 PIN). Przy zastosowaniu drugiego sposobu, musimy podłączyć



Taśma 2,5" łącząca Adapter z płytą główną

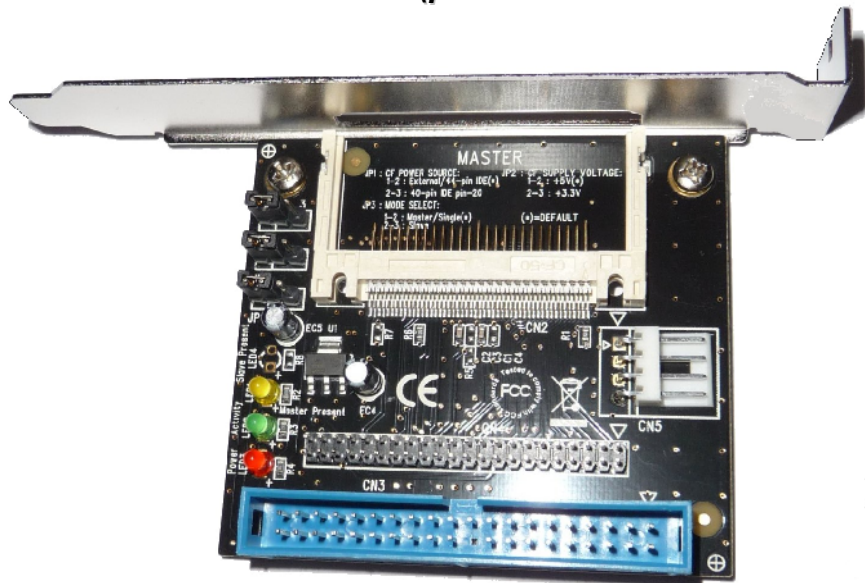
zewnętrzne zasilanie do naszego adaptera – służy do tego wtyczka (taka sama jak w stacji dyskiety). Dodatkowo

adapter ma przełącznik *master/slave*, dzięki czemu na jednej taśmie możemy podpiąć dwa takie urządzenia.

W moim przypadku adapter został podłączony do Amigi 600, dlatego też skorzystałem z taśmy 2.5" (IDE 44 PIN). Kartę, jaką wybrałem do kontrolera, to CF Kingston 4 GB (z kwiatusem) – ten model karty sprawia podobno najmniej problemów, choć inne karty, jakie posiadam, też nie sprawia-



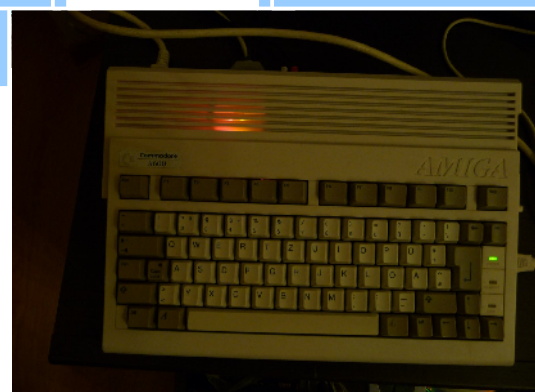
Karta Kingston 4 GB



Adapter CF-IDE

ly: Sandisk Extreme III 4 GB, Sandisk ULTRA 2 GB, Sandisk ULTRA 4 GB, Kingston 4 GB (wersja z zielonym liściem), Kingston 256 MB (wersja z kwiatkiem – na tej karcie instalowałem system podczas pisania tego artykułu). Na każdej z nich system poprawnie się zainstalował, a Amiga prawidłowo rozpoznała karty.

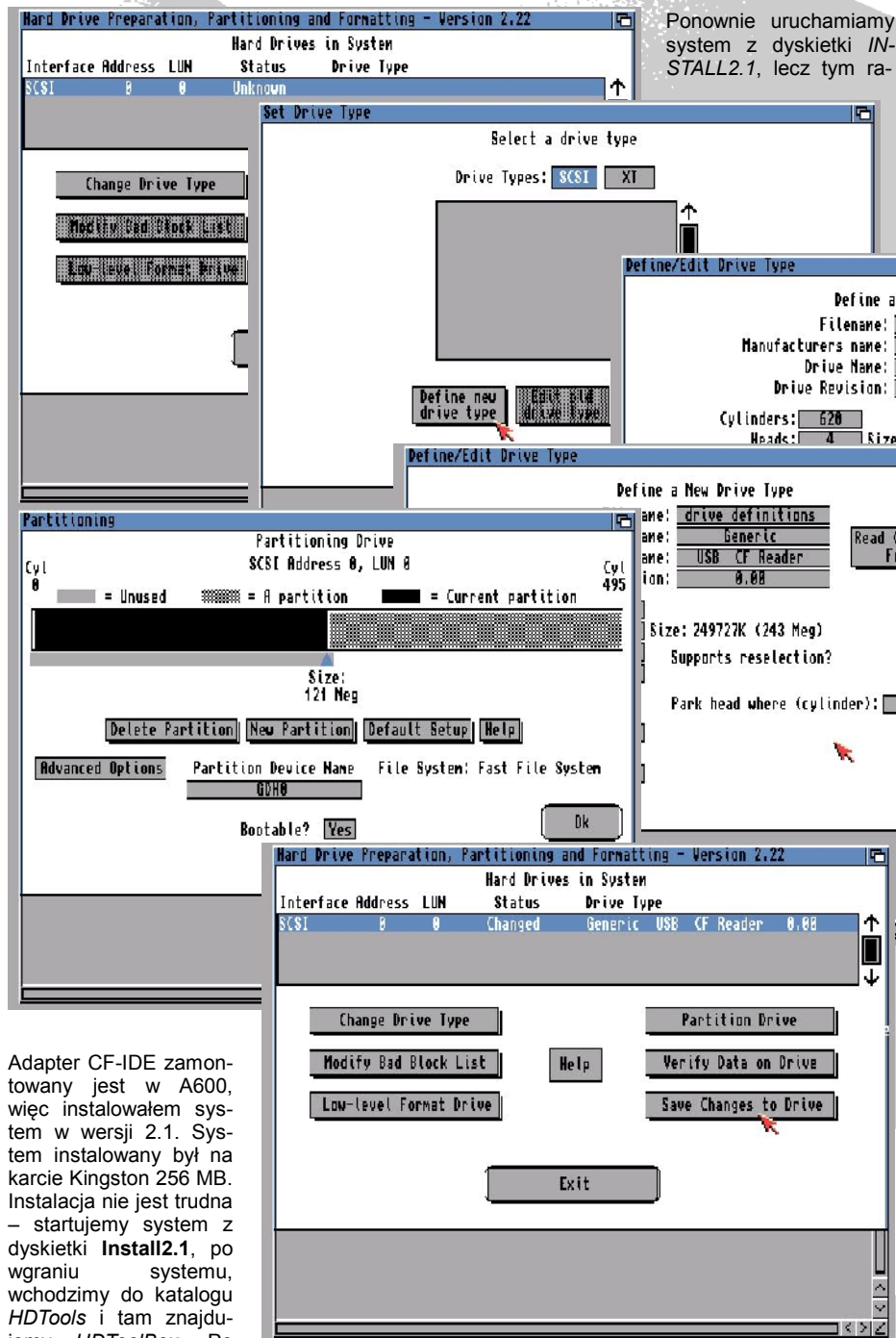
Najważniejsze jest to, że po podłączeniu adaptera działa dioda odczytu dysku – sprawdzone w A600 ROM 37.350 i A1200 ROM 3.0/3.1.



Wnętrze Amigi 600 oświetlone diodą LED adaptera



Instalacja systemu



Adapter CF-IDE zamontowany jest w A600, więc instalowałem system w wersji 2.1. System instalowany był na karcie Kingston 256 MB. Instalacja nie jest trudna – startujemy system z dyskietki **Install2.1**, po wgraniu systemu, wchodzimy do katalogu **HDTTools** i tam znajdujemy **HDTToolBox**. Po uruchomieniu programu pojawi nam się okienko z informacją o dysku – UNKNOWN. Następnie klikamy w opcję **CHANGE DRIVE TYPE**, a potem na **DEFINE NEW DRIVE TYPE**. Pojawi się nam ekran z błędem – „UNIT IS NOT A DISK”. Klikamy w OK i w opcję **READ CONFIGURATION FROM DRIVE**. Po chwili powinny pojawić się nam dane naszej karty. W polu **DRIVE NAME** wpisujemy nazwę dysku – ja wpisałem **USB CF READER** i klikamy w OK. Okno zamyka się i pojawia się główne okienko **HDTToolBox**, gdzie klikniemy w opcję **PARTITION DRIVE**. W tym momencie przystąpimy do ustawiania partycji na dysku – dokonujemy tego wg własnych potrzeb, choć musimy pamiętać, aby partycja DH0 była bootowalna. W tym celu zaznaczmy opcję **BOOTABLE – YES**. Po ustawieniu partycji, naciskamy OK, a następnie na **SAVE CHANGES TO DRIVE**. Następnie wy-

beramy **EXIT**, czekamy sekundę i resetujemy naszą Amigę.

Ponownie uruchamiamy system z dyskietki **INSTALL2.1**, lecz tym ra-

WHDLoad znajdziemy katalogi C, S, Devs, których zawartość kopiujemy do odpowiednich katalogów na dysku systemowym. Gry przeznaczone dla WHDLoad kopiujemy do wybranego przez nas katalogu i to cała filozofia.

Zaletą tego rozwiązania jest przede wszystkim cena. Karty CF 2 GB lub 4 GB można znaleźć za grosze, adapter kosztował niespełna 25 zł. Cała inwestycja zamknęła się w ok. 70 zł, łącznie z taśmą IDE 2.5” – dysk twardy byłby znacznie droższy. Kolejna sprawa to cisza –

jedynym głośnym elementem takiego zestawu jest cykająca stacja dyskietek. Dodatkowo karta widoczna jest od razu po włączeniu Amigi. Nie istnieje również obawa, że Amiga wy-startuje

szybciej niż dysk się rozkręci, a więc nie ma potrzeby resetu po włączeniu Amigi, co często zdarza się przy tradycyjnym dysku twardym.

Moim zdaniem największą zaletą tego rozwiązania jest możliwość prostego przenoszenia danych z PC do Amigi. Wystarczy posiadać WinUAE oraz czytnik kart CF. Uruchamiamy WinUAE, wchodzimy do opcji **HARD DRIVES**, klikamy w **ADD HARD DRIVE**. Pojawi nam się okienko z wyborem dysków, wybieramy naszą kartę CF – w moim przypadku to karta 256 MB – i zaznaczamy opcję **read/write**. Kolejnym krokiem jest zaznaczenie jakiegoś katalogu na dysku PC, na którym będziemy trzymać pliki przeznaczone dla Amigi. Klikamy w **Add Directory or Archive**, uzupełniamy odpowiednie opcje, klikamy w **Select Directory** i wybieramy nasz katalog – u mnie jest to ścieżka **H:\Amiga\Audio\ConVl**. Tym sposobem przenoszę dane pomiędzy PC i Amigą 600 – jest to chyba najprostszy i najtańszy sposób.

Szczerze się przyznam, że wad tego rozwiązania nie zauważyłem. Po kilku tygodniach intensywnego zabawy nie napotkałem na nic, co mógłbym zmienić lub poprawić czy też okazało się jakąś uciążliwością. Amiga startuje błyskawicznie, gry chodzą, demo scenowe też, nie ma błędów odczytu, nic się nie wiesza – jednym słowem IDEAL.

Zabawa, zalety i wady

W chwili obecnej moja Amiga posiada następującą konfigurację: procesor 68000, 2 MB pamięci Chip i 9.5 MB pamięci Fast, ROM 37.350 oraz jako twardy dysk kartę CF 4 GB Kingston. Amigi tej używam głównie do gier WHDLoad oraz starszych dem scenowych. Instalacja pakietu jest prosta, w archiwum z

Aleksander „trOLLO” Giedyk



Retro Game Compo

W dniach 26-27 października odbyła się w Gdańsku impreza RetroKomp/LoadError Party 2013. Była to druga edycja spotkania entuzjastów szeroko pojętej retro informatyki oraz nowoczesnych rozwiązań sprzętowych i softwarowych, wywodzących się z klasycznych platform, a także pierwsza edycja scenowego party LoadError, odbywającego się po zakończeniu części otwartej imprezy.

Nie będę w tym miejscu opisywał całej (nota bene bardzo udanej) imprezy – chciałem skupić się tylko na amigowych wątkach konkursu Retro Game Compo, którego byłem pomysłodawcą i prowadzącym. Celem konkursu było wyłonienie najlepszej gry na platformy retro. Konkurs trwał kilka miesięcy, zgłoszenia były na bieżąco prezentowane na stronie *retro-komp.org*, a sam konkurs był intensywnie komentowany m. in. na forum *ppa.pl*. Ostateczne głosowanie odbyło się podczas imprezy RetroKomp/LoadError, a głosować mogli wszyscy uczestnicy, każdej produkcji przydzielając punkty w zakresie 1-10. Głosowanie poprzedziła prowadzona przeze mnie prezentacja produkcji, na którą składały się parominutowe filmy z każdej gry, okraszane trzymającą w napięciu narracją prowadzącego.

Główną zasadą konkursu było to, że gra musi być stworzona od podstaw lub musi być portem gry z innej platformy do tej pory niedostępnej na platformie docelowej. Regulamin dopuszczał zarówno gry pisane przez programistów, jak i stworzone za pomocą aplikacji wspomagających tworzenie gier. Przyjmowaliśmy zarówno pełne wersje, jak i wersje demonstracyjne.

Łącznie do konkursu zgłoszono 13 produkcji, w tym 8 na Amigę, 3 na ZX Spectrum oraz po jednej na małe Atari i C64. Wyniki konkursu przedstawiają się następująco:

Miejsce	Produkcja	Platforma	Punkty
1	Krunel	ZX Spectrum	284
2	Montezuma	Amiga	210
3	CBM Asteroids	Commodore 64	208
4	Elixir Vitae	ZX Spectrum	205
4	Kiki Pong	Atari XL/XE	205
5	Deadly Labyrinth of Lord Xyrx	ZX Spectrum	204
6	Metro Cross	Amiga	199
7	Magic Ball	Amiga	197
7	Solomon's Key	Amiga	197
8	Bomber	Amiga	194
9	Castle Hack	Amiga	186
10	Key Master	Amiga	165
11	Crazy Crane	Amiga	149

Moje podsumowanie i wnioski z powyższych wyników pozwolę sobie przedstawić na koniec artykułu. Najpierw opiszę wszystkie produkcje, starając się jednocześnie subiektywnie je ocenić.

Montezuma

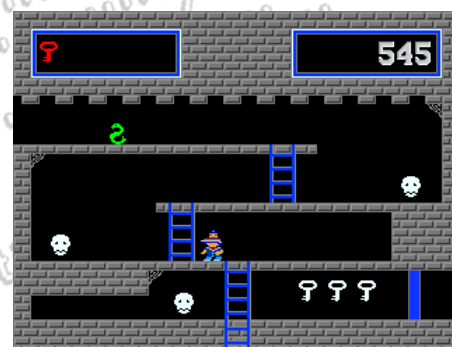
Gra napisana w języku AMOS, autorem całości jest J.P.M. – lubmil. Gra działa na każdej Amidze z 1 MB pamięci. Jak sama nazwa wskazuje jest to produkcja wzorowana na atarowskim „Montezuma's Revenge”. Według autora, kolejne wersje tej gry powstawały od 1991 do 2003 roku. Poszukiwacz skarbów i przygód Panama Joe wyrusza na niebezpieczną wyprawę do wielkiej świątyni Azteków, w której ukryty jest niesamowity skarb pełen klejnotów i złota. Aby się do niego dostać, musi przebić się przez liczne poziomy świątyni. Gra polega na bieganiu, skakaniu, omijaniu pułapek, otwieraniu zamkniętych przejść oraz używaniu przedmiotów, takich jak pochodnie do oświetlenia drogi albo miecze do walki z przeciwnikami. „Montezuma's Revenge” uważana jest za jedną z pierwszych gier platformowych w dziejach komputerowej rozrywki. Gra dosyć niespodziewanie zajęła drugie miejsce w Retro Game Compo i moim zdaniem jest to wynik zdecydowanie „na wyrost”. Może to kwestia tego, że nigdy nie byłem fanem „Montezumy” i nie czuję żadnego sentymentu do tej gry. Sama rozgrywka nie sprawiała mi zbytnej przyjemności. Gra ma toporną grafikę i animację, szczątkową oprawę dźwiękową i wydaje mi się po prostu nudna. Zdaje mi się, że wysoka pozycja gry w Compo jest wynikiem głównie nostalgii głosujących, którzy w młodości musieli spędzić wiele godzin z „Montezumą” na maszynach 8-bitowych.

Metro Cross

Port komercyjnej gry z Atari ST, stworzony metodą „inżynierii odwrotnej” (reverse engineering), czyli de-asmblacji kodu. Autorem konwersji jest Asman, a grę tworzy bardzo fajne cracktro Wanted Team, zakodowane przez Sachego. Gra działa na każdej Amidze z 1 MB pamięci. „Metro Cross” to gra zręcznościowa. Gracz ma do pokonania 24



Autor artykułu podczas prowadzenia Compo



Montezuma

poziomy, a jego zadaniem jest bieg przez tor przeszkód i zbieranie rzeczy, za które przyznawane są punkty. Postać biegnie automatycznie, zadaniem gracza jest unikanie przeszkód i zbieranie artefaktów. Gra oryginalnie dostępna była na automatach, a jej wydawca – firma Namco – przygotowała także konwersje na Atari ST, ZX Spectrum, Amstrada, C64 i NES. Amigowcom przyszło czekać ponad 20 lat na swoją wersję. Gra „Metro Cross” była jednym z moich amigowych faworytów w Retro Game Compo. Owszem, nie jest to oryginalna produkcja, jednak proces jej tworzenia ze strony programisty był bardzo wymagający. Produkcja jest grywalna, animacja płynna, oprawa graficzna i dźwiękowa są na niezłym poziomie. Sam kilkakrotnie wracałem do gry po to tylko, żeby sprawdzić czy uda mi się uzyskać lepszy wynik. Może nie jest to produkcja na poziomie komercyjnych gier z czasów świetności Amigi, ale na pewno dostarcza sporej porcji rozrywki.

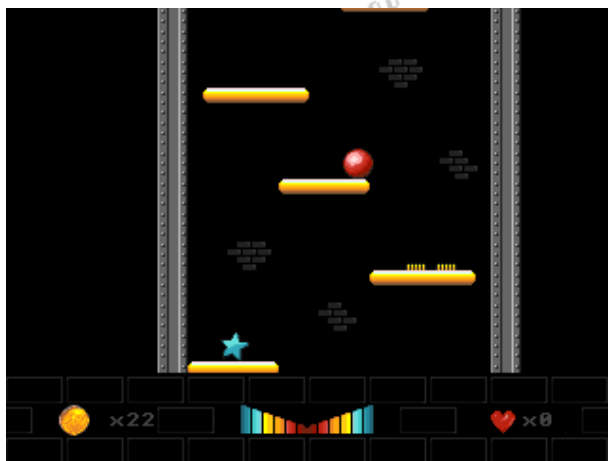


Metro Cross

Magic Ball

Gra stworzona w programie Backbone przez Czesława Mnicha, znanego także jako Leon, który narysował grafikę, skomponował muzykę i przygotował 8 poziomów rozgrywki. Minimalne wymagania gry to 1,5 MB wolnej pamięci i system w wersji minimum 2.0. Do wygodnej rozgrywki zalecam procesor 020. „Magic Ball” jest grą platformową z elementami zręcznościowymi. W grze sterujemy czerwoną kulą, której zadaniem jest przemierzanie mrocznych podziemi kryjących wiele ukrytych skarbów. Do pomocy w pokonaniu różnorodnych przeszkód posłużą nam klucze oraz dźwignie rozmieszczone na planszy. Podczas naszej przygody po podziemnych korytarzach natrafimy na wielu wrogów utrudniających nam wędrówkę. Obecna wersja gry zawiera tylko wspomniane 8 poziomów – w pełnej wersji planowane jest dodanie kolejnych.

Muszę przyznać, że gra „Magic Ball” spodobała mi się od pierwszego wejrzenia. Nie da się ukryć, że Leon przygotował dobrą oprawę graficzną. Zarówno ekran menu, jak i sama gra wyglądają więcej niż przyzwoicie. Grze towarzyszy także nienachalny moduł muzyczny. Rozgrywka jest przyjemna, gra przypomina nieco wydaną parę lat temu produkcję „Downfall” – w pierwszych etapach kontrolowana przez nas piłka spada z góry i musimy tak nią pokierować, żeby uniknąć przeszkód i zebrać jak najwięcej bonusów. W kolejnych etapach gra zamienia się w platformówkę z bocznym przesuwem. Pewnym mankamentem gry może być fakt, że stworzono ją w tzw. game markerze, jednak w czasie rozgrywki mało kto bę-



Magic Ball

Solomon's Key

Kolejny port z Atari ST przygotowany przez Asmana. Podobnie jak w przypadku „Metro Cross” grze towarzyszy cracktro Wanted Teamu, a port jest wykonany techniką inżynierii odwrotnej. Główny bohater imieniem Dana ma za zadanie zdobyć tytułowy klucz Salomona, aby uwolnić świat od demonów. Celem gry jest



Solomon's Key

pokonywanie kolejnych etapów, w których należy zdobyć klucz otwierający przejście do następnej komnaty. Dana, oprócz biegania i skakania, może tworzyć lub niszczyć pomarańczowe bloki, a także generować kule ognia, aby unicestwić demony. Pomarańczowe bloki mogą być zniszczone poprzez dwukrotne uderzenie głową bohatera. Po drodze Dana może zbierać bonusy, aby zwiększyć swoją siłę ognia i zdobyć dodatkowe życie, a także przedmioty, które zwiększają punkty bonusowe i odblokowują ukryte pokoje. Gra zawiera zarówno elementy przygodowe, zręcznościowe, jak i logiczne.

„Solomon's Key”, podobnie jak „Metro Cross” jest portem komercyjnej gry, więc możemy się spodziewać dobrej (w każdym razie jak na standardy Atari

ST) oprawy audiowizualnej i niezłej grywalności. Gra jest dość trudna, samo zrozumienie zasad i logiki rozgrywki zajęło mi trochę czasu i ostatecznie gra nie przypadła mi do gustu. Wiem jednak, że „Solomon's Key” ma swoich fanów i zwolenników i na pewno należy cieszyć się, że mamy na Amidze kolejną profesjonalnie napisaną grę.

Bomber

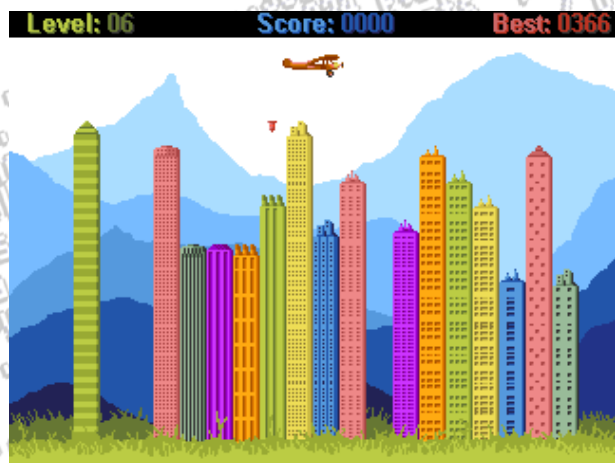
Gra zręcznościowa, przygotowana przez zespół w składzie: Wojciech Jancz (programowanie w Amosie), Czesław Mnich (grafika), Andrzej Drozd (muzyka). Gra działa na każdej Amidze z 1 MB pamięci. „Bomber” jest wzorowany na klasycznej grze „City Bomber” wydanej w 1982 roku przez firmę Creative Software na komputery Commodore 64. W grze wcielamy się w rolę pilota podczas II Wojny Światowej. Naszym zadaniem jest zniszczyć wszystkie budynki, jakie znajdują się na ekranie. Z każdym następnym poziomem budynki są coraz wyższe, co znacznie podnosi poziom trudności. Dodatkowo samolot po każdym przelocie leci coraz niżej, uniemożliwiając trafienie wyższych budynków. Grze towarzyszy wojenna muzyka, dzięki której możemy poczuć się jak podczas prawdziwej walki.

„Bomber” jest kolejną grą, w której palce maczał Leon i to właśnie jego grafika jest największym atutem gry. Zarówno budynki jak i tła oraz samolot są ładnie narysowane i kolorowe. Rozgrywka jest dość prosta i według mnie trochę za mało urozmaicona, żeby przyciągnąć na dłużej niż kilkanaście minut – przydałoby się ją nieco urozmaicić dodając jakieś dodatkowe elementy, bonusy do zebrań, przeszkadzajki itp. Jest to jak najbardziej do zrobienia.

Drugim elementem, którego mi zabrakło, jest zapisywanie listy najlepszych wyników. Gra ma jednak potencjał i niewielkim nakładem pracy można ją urozmaicić tak, aby rozgrywka była bardziej satysfakcjonująca dla gracza.

Castle Hack

Jest to tak naprawdę demo silnika gry, która jest na etapie powstawania. Bardzo dobrze się jednak stało, że autorzy, którzy zresztą uczest-



Bomber



niczyli na miejscu w imprezie, postanowili zaprezentować preview na Compo. Twórcami gry są Kain i Proxy odpowiedzialni za kod w języku C i grafikę, wspomagani graficznie przez Krótkiego. Gra działa na każdej Amidze z 1 MB pamięci – docelowo będzie w nią mogło grać do 4 graczy jednocześnie. Pomysł na grę zrodził się z braku dobrych gier na czterech graczy, które by zajmowały czas na dłużej niż kilka godzin. „Castle Hack” inspirowany jest takimi tytułami jak „Gauntlet” i „Hammer Watch”. Celem gry jest przedzierzgnięcie się przez kolejne piętra lochów, wycinając przy tym hordy złych potworków. No i siebie nawzajem, jeśli ktoś ma ochotę. Gra została zaprezentowana w Compo w bardzo wczesnym stadium rozwoju. Na dobrą sprawę jest to sam silnik – jeszcze bez przygotowanych etapów i bez muzyki. Gra ma jednak według mnie bardzo duży potencjał – już we wczesnym demie widać płynny przesuw obrazu i ładnie animowane obiekty. Widać, że chłopaki mają i zapał i talent. Myślę, że gdyby gra była dokończona, to byłby to murowany faworyt konkursu. Należy mieć tylko nadzieję, że jakoś w roku 2014 będzie nam dane pograć w finalną wersję gry.



Castle Hack

Key Master

Gra stworzona w Amosie przez Selura, który także narysował grafikę. Jak większość zgłoszonych gier, zadowoli się Amigą 500 z 1 MB pamięci. Gra oparta na produkcji z ZX Spectrum „Adventures of Sid Spider” – zbierz klucze, otwórz nimi odpowiednie drzwi i udaj się do wyjścia. Drzwi otwieramy kluczem odpowiedniego koloru. Naraz można dźwigać po jednym kluczu w danym kolorze (klucz tęczy otwiera wszystkie drzwi). Dodatkowe obiekty to młot (usuwa beczki), teleпорты (przenoszą w wybrane miejsce) oraz magiczne zwierciadła (zamieniające się w przedmiot przy dotknięciu). Niektóre etapy wymagają zebrania odpowiedniej liczby monet do ukończenia, a buty dają dodatkowe punkty ruchu, które podczas rozgrywki ubywają. Muszę przyznać, że gra „Key Master” bardzo mnie zaskoczyła. Pierwszy raz pozytywnie – poziomem wykonania i dobrą grywalnością oraz po raz drugi negatywnie – niską pozycją na Compo. Naprawdę nie rozumiem, dlaczego tak ładnie wykonana graficznie gra znalazła się tak nisko w zestawieniu. Selur kończył grę dosłownie na ostatnią chwilę, ale według mnie warto było czekać. Poza dobrą grafiką gra zapewnia całkiem przyjemną rozgrywkę – etapy stają się coraz trudniejsze i zróżnicowane, a poza kombinowaniem z kolejnością zbierania kluczy i otwiera-

nia drzwi sukcesywnie pojawiają się kolejne elementy i urozmaicenia rozgrywki. Czego brakuje? Może paru fajnych modułów muzycznych, jakiegoś ekranu tytułowego, możliwości zapisywania najlepszych wyników punktowych? Selur, tak trzymać!

Crazy Crane

Ostatnie miejsce w Compo zajęła gra „Crazy Crane” stworzona przez Grzegorza „Krashana” Kraszewskiego i Leona. Gra wymaga systemu w wersji 3.0, pracuje zarówno na komputerach z chipsetem ECS, AGA, jak i na kartach graficznych w oknie na ekranie Workbench. „Crazy Crane” jest grą logiczną, w której do zadań gracza jest odgadnąć hasło składające się z kilku wyrazów. Zasady gry są bardzo proste – po wyczycaniu bloków z literami musimy w każdej kolumnie wybrać po jednej literze. Jeśli zaznaczyliśmy prawidłowe litery odgadując określony wyraz, to reszta nieużywanych liter automatycznie zostanie usunięta z hasła. Do gry dodatkowo dołączona jest punktacja, która za szybkie odgadnięcie hasła da nam dodatkowe punkty. Gra „Crazy Crane” przegrała według mnie głównie ze względu na oszczędną grafikę. Nie jest to wada, ponieważ przy tym typie gry logicznej zasadniczo nic więcej nie jest wymagane. „Crazy Crane” daleko także do statusu gry ukończonej. Podobnie jak w przypadku „Castle Hack” jest to, według mnie, jedynie silnik gry, do którego należy dopisać całą otoczkę. Nie zaszkodziłoby dodatkowe zestawy klocków, zróżnicowane tła, oprawa dźwiękowa czy chociażby funkcja zapisywania najlepszych wyników na serwerze, w celu konkurowania z innymi.

Podsumowanie

Podsumowując, można stwierdzić, że Retro Game Compo pokazało, że mamy w Polsce grupę zdolnych zapaleńców, chcących tworzyć ciekawe gry na klasyczne Amigi. Pozytywnie nastraja ilość zgłoszonych produkcji – teraz czas popracować nad ich jakością. Bez wątpienia twórcy zwycięskiej gry „Krunel” pokazali, w którą stronę należy iść. Do maksimum wykorzystali ograniczo-



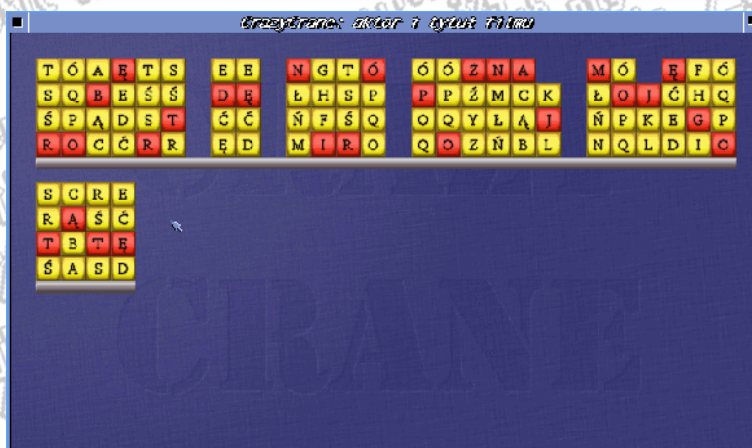
Key Master

ne możliwości sprzętu, wrażenie robił też sam zestaw ksyw osób odpowiedzialnych za grę – każdy, kto jest chociaż minimalnie obeznany ze sceną, rozpozna od razu takie osoby jak Tygrys, Atom, Slayer, Catman czy Ziutek. Właśnie w kierunku szerokiej współpracy i dążenia do maksymalnego wykorzystania możliwości retro-sprzętu powinni iść amigowcy. Dobrym wzorem do naśladowania jest nie tylko spectrumowy „Krunel”, ale chociażby recenzowana przeze mnie w niniejszym numerze PPA gra „Moebius Goatlizard”...

Na zakończenie chciałem podziękować wszystkim uczestnikom Retro Game Compo – za Wasz odzew i pracę włożoną w zgłoszone gry. Myślę, że po trochu wszyscy jesteście zwycięzcami. Chciałem także podziękować wszystkim, z którymi było mi dane współpracować przy organizacji RetroKomp/LoadError, ze szczególnym uwzględnieniem dwóch osób – Sachego i Mccnexa, bez których wielkiego zaangażowania imprezy by nie było. Bardzo dziękuję też mojej partnerce Kasi, która dzielnie znosiła moją niedostępną w okresie przygotowywania imprezy.

Wszystkie produkcje do pobrania ze strony <http://retrokomp.org/wordpress/>. W artykule wykorzystano zredagowane fragmenty opisów gier nadesłane przez autorów. Autorką zdjęć jest Kasia Koselak.

Mateusz „Twardy” Eckert



Crazy Crane



The Incredible Adventures of Moebius Goatlizard

Często czytam narzekania, że nowych gier na Amigę powstaje jak na lekarstwo. Rzeczywiście, w porównaniu z platformami 8-bit, takimi jak ZX Spectrum czy C64, nowych i ciekawych produkcji na Amigę klasyczną jest mało. Nie warto rozwodzić się w tym miejscu na temat powodów tej sytuacji, ale faktem jest, że każda nowa produkcja, tym bardziej tak dobrze wykonana, jak *"The Incredible Adventures of Moebius Goatlizard"* (w skrócie „Moby Goat”), nie może przejść niezauważona.

Grę napisała brytyjska grupa Black Jet, która wcześniej wydała opartą na podobnym pomysle produkcję na komputer Sam Coupe (*„The Lost Disks of Sam”*). Warto także dodać, że pierwotna koncepcja gry narodziła się na ZX Spectrum jako *„Endless Forms Most Beautiful”*. Gra uruchamia się na każdej Amidze i wystarczą jej 512 kB pamięci. Dodatkowa pamięć wykorzystywana jest do odtwarzania muzyki w menu oraz do dodatkowych efektów dźwiękowych.

O co w tym biega?

Oficjalna historyjka do gry jest bardzo prosta i zawiera się w jednym zdaniu: „Nieokielznane pożądanie do wszystkich rzeczy, które są kosmiczne, przywiodło Moebiusa do świata Amigi, gdzie musi zebrać cały międzygalaktyczny stuff, aby zaspokoić swój nieczny nałóg”. Rozgrywka polega na tym, że biegamy po platformach ułożonych poziomo na całym ekranie gry, a naszym zadaniem jest zebranie wszystkich artefaktów w danym etapie, przy jednoczesnym unikaniu kontaktu z przeciwnikami. Na każdej platformie znajdują się teleportery, dzięki którym możemy przenieść się na poziom powyżej, bądź poniżej tego, na którym się aktualnie znajdujemy. Dodatkowo poziomy zmieniamy wychodząc prawą bądź lewą stroną platformy. Dzięki umiejętnemu korzystaniu z teleporterów unikamy kontaktu z przeciwnikiem, a tym samym natychmiastowej utraty życia. Nie jest to wcale takie proste, ponieważ nasi

wrogowie nieustannie biegają w lewo i prawo, w dodatku przyspieszając bądź zwalniając w zależności od położenia naszego bohatera. Dysponujemy także ograniczoną liczbą teleportacji poprzez „hiperprzestrzeń”. Gdy zrobi się za gorąco, po prostu przyciskamy fire i nasz Moebius materializuje się w innym (losowym) miejscu. Od czasu do czasu pojawiają się dodatkowe bonusy i power-upy (np. tymczasowa nieśmiertelność bądź czasowe zatrzymanie przeciwników, większa szybkość, dodatkowe życie). Rozgrywka odbywa się w dość szybkim tempie i z początku wydaje się nieco trudna, jednak po rozpoznaniu zasad i schematów, jakie występują w grze, staje się bardzo przyjemna. Gra oferuje pięć różnych poziomów trudności oraz tryb dla dwóch graczy.

Co widać, co słychać?

Mimo minimalnych wymagań sprzętowych, gra wygląda naprawdę dobrze. Grafika ma dobry styl i coś co można nazwać retro-feeling. Zarówno ekran tytułowy i końcowy oraz ekrany wyświetlające się pomiędzy etapami są wykonane starannie i estetycznie. Także podczas samej rozgrywki mamy do czynienia z bardzo dobrą oprawą graficzną. Etapy są kolorowe, a jednocześnie czytelne, animacja postaci jest szybka i bardzo płynna. Widać, że gra została zakodowana w assemblerze – jednocześnie na ekranie bardzo szybko biega 9 postaci, a gra nie zwalnia nawet na A500 z 0,5 MB pamięci. Uwagę zwraca także efekt „skakania” całego ekranu po zakończeniu danego poziomu gry. Oprawa dźwiękowa gry jest z kolei dosyć oszczędna. Podczas wyświetlania ekranu tytułowego przygrywa nam przyjemny chiptune, a w czasie samej rozgrywki mamy już tylko efekty dźwiękowe.

je się IDENTYCZNIE. Na podstawowej konfiguracji nie ma nawet jednej klatki animacji mniej, niż na ponad 100 razy szybszym sprzęcie. Gra bez problemu uruchamia się z twardego dysku. Jedynym zauważalnym minusem może być brak tablicy z możliwością zapisania najlepszych wyników.

Grę można pobrać za darmo ze strony autorów. Przez pewien czas można było także zamówić wersję dyskietkową, w specjalnie na tę okazję przygotowanym opakowaniu. Dla osób, które zakupiły wersję dyskietkową autorzy zafundowali bonus w postaci dodatkowego trybu rozgrywki.

P.S. Jako ciekawostkę dodam, że gra wzbudziła zainteresowanie zwiedzających i grających na RetroKomp/LoadError'13.

Mateusz „Twardy” Eckert

Grę można pobrać za darmo ze strony autorów:

<http://blackjet.co.uk/amiga.php>



Jeden z etapów gry

Podsumowanie

„The Incredible Adventures of Moebius Goatlizard” jest produkcją dopracowaną pod każdym względem. Gra jest ładna, wciągająca i dobrze napisana. Osobiście przetestowałem ją na A500 bez rozszerzenia pamięci, A600 z 2 MB Chip i 4 MB Fast, A1200 z procesorem 68060 oraz na WinUAE i na każdym z tych sprzętów gra zachowu-

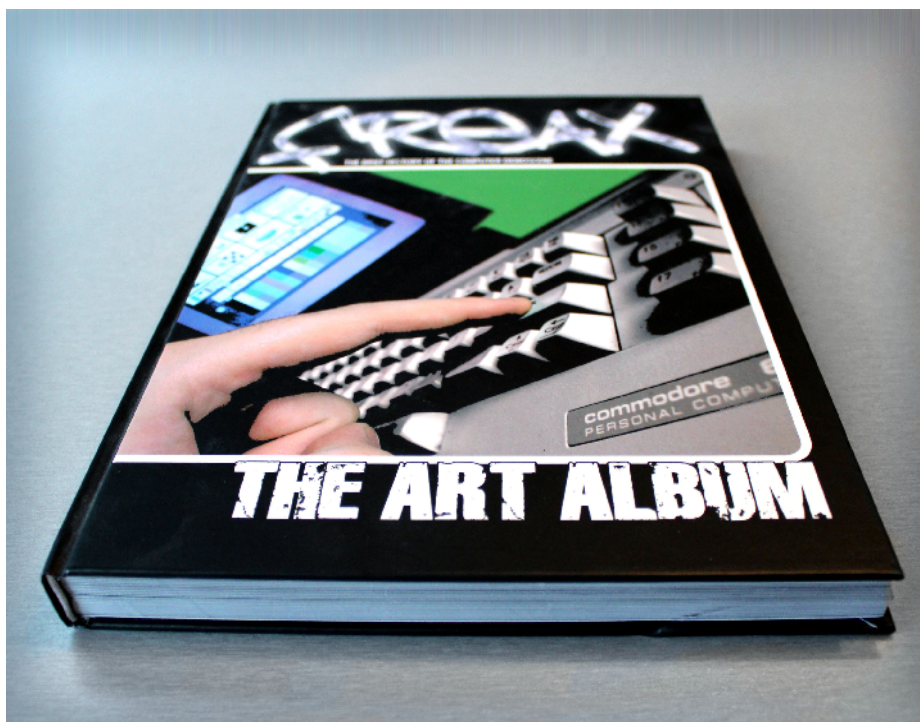


Chyba koniec gry...

FREAX – The Art Album

Obserwując niektóre współczesne wystawy i wydarzenia kulturalne, zastanawiam się czasami czy słowo „sztuka” cokolwiek jeszcze dzisiaj oznacza. Czy fakt legitymowania się przez kogoś dyplomem ASP automatycznie implikuje, że mamy do czynienia z „prawdziwym artystą”, który tworzy „prawdziwą sztukę”? Cóż, nie ukrywam, że z każdym kolejnym rokiem, mam co do tego coraz większe wątpliwości. Być może to właśnie poprzez znużenie i irytację ową „sztuką współczesną”, a być może i poprzez falę popularności vintage computingu i retro gier, które to przedzierają się ostatnio do (pop)kultury, demoscena w końcu zaczyna wychodzić z podziemia. Coraz częściej można usłyszeć głosy, że środowisko sceny i jej produkcję, mają ze sztuką znacznie więcej wspólnego, niż się to wszystkim kiedyś wydawało.

W ostatnich latach daje się zauważyć zmieniający się stosunek „mainstreamu” do demosceny. Można wymienić liczne przykłady, świadczące o tym, że scena zaczyna być traktowana poważniej. Jest nim chociażby „Moleman 2 – demoscena: Sztuka algorytmów” – głośny węgierski film dokumentalny o demoscenie, wyreżyserowany przez Szilárda Matusika, który pokazywany był między innymi na 28. Warszawskim Film Festival. Dokument ten jest zresztą w całości dostępny w serwisie YouTube i to z polskimi napisami (polecam). Tematyka demoscenowa trafia również na uczelnie – pisane są o niej prace, prowadzi się wykłady (np. „Demoscena – cyfrowe graffiti”, który odbył się na Politechnice Poznańskiej podczas Zjazdu Twórców Gier 2012). Przewija się też w galeriach. Oł, choćby w listopadzie 2013 roku – dla przykładu – we wrocławskiej galerii Centrum Kultury Agora odbyło się otwarcie wystawy



„Demoscena – między sztuką, a technologią”. Bez wątpienia coś się więc „ruszyło”.

„FREAX – The Art Album” wpisuje się w ten właśnie nurt. Nurt, pokazujący, że scena to dzisiaj zjawisko artystycznie ciekawe i inspirujące, które w połączeniu z powracającą – wspomnianą już przeze mnie – modą na klimaty retro, może dziś zainteresować szersze

grono odbiorców niż tylko wiadomą „grupę wtajemniczonych”. Recenzowany album to tak naprawdę tom drugi większej całości. Tamas Polgar, autor tej publikacji, wydał już wcześniej książkę „FREAX Volume 1: C64 and Amiga demoscene (hi)story”, opisującą w niezwykle barwny i przystępny sposób historię demosceny C64 i Amigi (planowany jest także tom trzeci – dotyczący PC). Jako powód powstania albumu Tomcat (xywa autora) wyjaśnia, że mamy już na rynku płyty DVD z kompilacjami starych dem, mamy płyty CD z klasycznymi modułami i remixami, ale dziwnym trafem jakoś nikt nigdy nie zebrał najlepszych scenowych grafik i nie wydał ich w formie drukowanego albumu. Tamas wypełnił więc lukę.

Już na pierwszy rzut oka widać, że „FREAX – The Art Album” to w żadnym wypadku jakieś tam chałupnicze wydawnictwo, drukowane w garażu na drukarce igłowej Star LC-10 i powielane na ksero w piwnicy. To album wydany w taki sposób, że równie dobrze mogłoby to być publikacja dla jakiejś znanej galerii czy muzeum. Do tej wysokiej jakości przyzwyczaił nas już zresztą wydawca (patrz wspomniany pierwszy tom serii). Album liczy 286 kolorowych stron (doskonałej jakości papier kredowy), jest formatu A4 (30 x 21 cm) i ma twardą, grubą okładkę. Jak na tego typu wydawnictwa przystało, jest słusznie „ciężki wagowo” i prezentuje się wprost bajecznie. Publikacja zawiera około 1600 grafik demoscenowych twórców związanych z C64, Amigą, ZX Spectrum, Atari ST i Falcon, Amstradem CPC i PC. Mamy tu spory przekrój techniczny, poczynwszy od ASCII/ANSI i dwukolorowej grafiki w niskiej rozdzielczości, poprzez 8-bit, a na truecolorowych pracach skończywszy. Większość z nich to oczywiście dzieła ręcznie pikselowane/ryso-



Kto powinien zakupić ten album? Po pierwsze – fani demosceny, klimatów retro i kolekcjonerzy. Dla nich jest to pozycja obowiązkowa. Po drugie – polecilibym go osobom, które interesują się szeroką pojętą sztuką i które chciałyby poznać nowe intrygujące zjawisko, jakim jest demoscena. Po trzecie – rekomenduję go wszelkim projektantom i grafikom, którzy na co dzień zajmują się tworzeniem i/lub projektowaniem użytkowym. „FREAX – The Art Album” może być doskonałym źródłem inspiracji i wzbogaceniem warsztatu – szczególnie godny

IX

Title: Eyeface
Author: Rack
Group(s): Majic 12/Absolute

128

Album można zamówić na stronie wydawnictwa MAZ-Sound (<http://www.maz-sound.com>) za 29.95 euro (plus koszty wysyłki). Czasem można go także kupić na różnych imprezach

Tomasz Pacyna (Slayer/Ghostown)
info@pixeldreams.pl





Z listów starego amigowca...

„Są chwile, gdy wolałbym martwą widzieć Cię, nie musiałbym się Tobą dzielić, nie nie...”

Gdy widzę, jak kolejna osoba na forum PPA z uporem godnym lepszej sprawy uświadamia wszystkich dookoła, że gdyby Amiga była oficjalnie martwa „to dopiero by było, panie dziejku”, wszyscy siedzielibyśmy radośnie w retro-gamedevie albo wreszcie „pokazali tym patałachom z atarowej/zx-owej/commodorowej (niepotrzebne skreślić) sceny, gdzie jest ich miejsce”, to przypominają mi się właśnie te, pierwsze wersy klasycznej piosenki zespołu Hey „Zazdrość”. Oczywiście w piosence podmiot liryczny „była kobietą”, więc wyglądało to trochę inaczej, ale chyba łapiecie ideę, prawda?

Myślę, że po tym poetyckim wstępie nadeszła pora na wyrażenie jasno mojego zdania. Otóż nie zgadzam się ze stwierdzeniem, że AmigaOS 4.x, AROS czy MorphOS (kolejność alfabetyczna, zanim ktoś dorobi jakieś teorie) były jakimiś heretyckimi, schizmatyckimi odłamami „prawdziwego amigowstwa”, żerującymi na Wielkim Dziedzictwie i godnymi jeno potępienia. Aby to zrozumieć, konieczne jest cofnięcie się pamięcią do czasów, których spora część orędowników „czystości rasowej” po prostu nie pamięta. Dlaczego? Mam swoje teorie, ale „o tem potem”.

Dla tych, którzy czytają ten tekst przypadkiem albo są zbyt młodzi, żeby pamiętać – 29 kwietnia roku pamiętnego (a wiecień był piękny w 1994) firma Commodore oficjalnie ogłosiła bankructwo. Amigowcy zostali osieroceni, bez firmy-matki, bez planów na przyszłość – w sytuacji niemalże beznadziejnej. Wielu stwierdziło wówczas „no trudno, kupię sobie peceta” i opuściło „tonący okręt” (tak zwana „pierwsza fala” to – z mojego doświadczenia – z reguły ci, którzy traktowali Amigę jako sprzęt do gier. Dzięki szybkiej „inwestycji” w Optimusa z 486 i SVGA znów można było zostać „królem podwórka”). Wielu jednak zostało – bądź to z wierności marce, bądź to z przyzwyczajenia albo po prostu – bo używanie tego, a nie innego komputera sprawiło im po prostu frajdę.

Czas upływał, kolejne próby reaktywacji marki okazywały się nieskuteczne, obietnice „no-

wych, lepszych komputerów” opartych na nowoczesnych (produkowanych!) procesorach i przepisania pod tę nową architekturę systemu operacyjnego naszych „przyjaciółek” okazywały się bądź to marketingową ściemą, bądź listą pobożnych życzeń rzucanych bez pokrycia. Wąski strumyczek oprogramowania wysychał coraz bardziej, kolejni amigowcy opuszczali (nieraz ze smutkiem) ulubioną platformę, przenosząc się na PC. Amiga jednak nadal żyła, dzięki swoim użytkownikom. Czy to z wierności marce, czy z przyzwyczajenia, czy też po prostu dla przyjemności korzystania z takiego a nie innego komputera – zostali. Nie byli jednak stadem bezrozumnych baranów (owieczek?) czekających na kolejne zmiany właściciela marki i obietnice gruszek na wierzbie. Amigowanie to kreatywność. Społeczność postanowiła działać. Z frustracji, zniecierpliwienia i determinacji powstały rozszerzenia sprzętowe, nowe karty graficzne i muzyczne, busboardy, adaptory, przelotki i przejściówki. Powstały łatki, hacki, zamienniki, obejścia przestarzałych funkcji istniejącego oprogramowania, a w końcu – nowe systemy operacyjne. Kochani, proszę się teraz skupić, bo to niezmiernie ważne jest – owe systemy powstały nie po to, żeby „dokopać Amidze”, „żerować na marce”, „wykorzystać modę na retro” czy „zakłamać pojęcie Prawdziwej Amigi”. Powstały po to, żeby **nadal używać systemu i oprogramowania, które się zna i lubi oraz cieszyć z tego frajdę**. Capisci? Frajdę, której trochę mało zostaje, gdy w utrzymanie skanowanego sprzętu i załatanego oprogramowania wkładamy za dużo wysiłku. My wiemy, że nie dogonimy uciekającego w zawrotnym tempie świata IT, wiemy że taniej i szybciej można pracować na PC z elektroszrotu – nie trzeba nam tego przypominać. Podobnie jak Wam, kochani amiortodoksi nie trzeba przypominać w kółko, że to samo można powiedzieć o zabawie z Amigami jak najbardziej klasycznymi. Jeśli do kogoś przemawia Karol Marks – „każdemu według jego możliwości, każdemu według jego potrzeb” – mamy inne potrzeby i inne możliwości. Ja trzymam swoją A1200 w pokoju u dzieci i włączam od wielkiego dzwonu – na przykład, gdy przyjdą znajome dzieciaki (OK, może mają PS5 czy XBOX1440, ale takiego sprzętu nie mają na pewno). I to jest OK. Ktoś inny trzyma u siebie w sypialni wszystkie modele Amig od Commodore i odpala codziennie każdą, żeby nie czuła się zaniedbana i to też

jest OK. A że nie otworzy na żadnej z nich ulubionej strony internetowej, nie zrobi przelewu do banku czy nie obejrzy filmu z YouTube – jemu to nie przeszkadza, ma od tego PC. I niech ma, tylko proszę – bez wyrzutów, że ja nie robię jak on i że wolę używać do tych celów amigowego oprogramowania.

Teraz mamy rok 2013, a przy boskich auxiliach może nawet 2014. Wielu spośród tych, którzy onegdaj (z tego czy innego powodu) zmieniło platformę, powraca teraz myślami do szczęśliwych lat dzieciństwa, kiedy żona nie zrzędziła, dzieci nie domagały się coraz to nowych gadżetów, a szef w korporacji nie objeżdżał za byle co. Wklepują w Google słowo „Amiga” i... nie wierzą własnym oczom! Zdrada! Nie o taką Amigę żeśmy... Hmm... Nie takiej żeśmy się spodziewali? Najdrożsi, mamy (my, ci którzy zostaliśmy, którzy wybraliśmy jedną z dróg, którymi podąża współczesny (post?)amigowy świat) do Was wielką prośbę – nie miejcie do nas pretensji o to, że pozbawieni Waszego wsparcia, mądrości i pomocnej dłoni rozwinięliśmy tę platformę w kierunku, który Wam nie odpowiada, OK? Nie rozdzierajcie szat nad tym, że śmiemy ciągle korzystać z oprogramowania, które niektórzy spośród nas nadal piszą, że używamy naszych systemów na co dzień, że ciągle sprawia nam to frajdę. Szaty mogą się jeszcze przydać, jak przyjdą mrozy. A energię przeznaczoną na „uświadamianie nas” i „wyprowadzanie z błędu” w złudnej nadziei, że rzucimy się z powrotem na 3.1 z łatkami i z dziką furią zajmiemy retrodeveloperką lepiej jest przeznaczyć na retrodeveloperkę właśnie.

„Kochani ludożercy (...)

Nie mówcie odwróceniem tyłem:

ja mnie mój moje

mój zołdek mój włos

mój odcisk moje spodnie

moja definicja prawdziwej Amigi

tylko to co ja lubię

Kochani ludożercy

nie zjadajmy się Dobrze?

macie jeszcze tyle wspańskich

gier/dem na klasyka do napisania”

Konrad Czuba





Scenowe graffiti

Ani malowanie pisanek, ani święcenie bab i kołaczy nie należą – wbrew pozorom – do ulubionych zajęć prawdziwych scenowców w czasie Świąt Wielkanocnych. Większość z nich przebywa bowiem wtedy w niemieckim mieście Saarbrücken, w industrialnej hali E-Werk. To właśnie tam organizowane jest **Revision Party**. W tym roku impreza odbyła się już po raz trzeci, znów okazując się sukcesem.

Licznie zgromadzoną publiczność czekało w Niemczech kilka ekscytujących wydarzeń. Dla amiscenowców było to – między innymi – **Oldskool Demo Compo**. W konkursie, pośród produkcji na platformy 8 i 16 bitowe, spotkały się dwa demo na Amigę 500 legendarnych grup Lemon i Spaceballs. Można się więc było poczuć przez chwilę jak na The Gathering 1993, kiedy to oba te teamy wystawiły w amigowych konkursach swoje produkcje. „*Rink a Dink: REDUX*” grupy Lemon, które zajęło miejsce pierwsze, to niespodziewany powrót po 20 latach. Demo, mimo, że jest remiksem tego sprzed dwóch dekad, robi spore wrażenie. Oldskulowa dbałość o szczegóły, dopracowane przejścia między efektami i świetna oprawa graficzno-muzyczna, z pewnością mogą się podobać (kod – Paradroid, grafika – Facet, Prowler, muzyka – Magnar (ex. Lizard/Andromeda), Nuke). Drugie demo na OCS – „*Straff*”, grupy Spaceballs, które zajęło miejsce czwarte, prezentuje zupełnie odmienny styl. To nowoczesny, psychodeliczny, koderski show z idealnym synchro, pokazujący, że nawet dziś, można wysmażyć coś zupełnie nowego na A500 (kod – Slummy, 3D – Quisten, muzyka – Tyson).

Amiga Demo Compo przyniosło natomiast kolejny pojedynek polskich grup. Zarówno



Z lewej: „Machinist” Elude, z prawej: „Smoke & Mirrors” Ghostown & Loonies

Elude, jak i połączone siły grup Ghostown i Loonies dały z siebie wszystko, a wynik tego starcia ważył się do ostatnich minut głosowania. Oba demo prezentują zupełnie inną stylistykę. „*Machinist*” grupy Elude (kod – Kiero, grafika – Ubik, Substrate, muzyka – Traymuss) to powrót do wcześniejszych klimatów tego tematu. Mamy więc mroczną, przytłaczającą atmosferę, porywające scenki 3D, skomplikowane efekty i epicką monumentalność. „*Smoke & Mirrors*” grup Ghostown i Loonies (kod – Blueberry, grafika – Slayer, Darklight, Codi, muzyka – JazzCat) to z kolei dynamiczna, steampunkowa podróż, z gitarowym soundtrackiem zgranym z efektami i sporą ilością ręcznie rysowanej grafiki, połączonej z obiektami 3D. Co prawda głosujący przeważyli szalę w stronę „*Smoke & Mirrors*”, ale wiele osób gratulowało

później obu grupom tego starcia w compo, nazywając je nawet „the epic battle”.

Na koniec wspominek z Revision 2013 jeszcze kilka słów o idolu nastolatek i ikonie amigowej sceny, którego ksywa to – jak wszyscy wiemy – Britelite. Oj, zmarł on bardzo wszystkich swych fanów, obecnych na party. Już od pierwszych godzin imprezy, z prędkością karty Cyberstorm, partowiczów obiegła informacja, że Peter przyjechał w tym roku do Niemiec bez demo. Okazało się to jednak przewrotnym żartem. Britelite i koledzy z grupy Dekadence, owszem, nie zrobili wprawdzie demo, ale przygotowali 64-kilobajtowe intro, które równie dobrze nazwać można demem właśnie. „*Thrice*” (kod – Britelite, grafika – Bracket, muzyka – Response), bo o nim mowa, to produkcja na procesory 68060, w której udało się upchnąć aż trzy utwory muzyczne, masę efektów, w tym scenki 3D, a wszystko ubrane w świetny design. Pojawiły się nawet głosy, że to najlepsze amigowe 64k intro od czasu kultowego „*Gift*” grupy Potion. Warto to osobiście sprawdzić.

Pod koniec września odbyło się w Łodzi party **We-Can**. Z amigowych akcentów trzeba odnotować tam fakt powrotu grupy Whelpz (we wrześniu minęło równe 10 lat od ostatniej produkcji sygnowanej tą nazwą). Ich tajemnicze, amigowe demo zajęło tam pierwsze miejsce w konkursie. Niestety ku rozczarowaniu „sofa scenerów”, którzy nie odwiedzili miasta Łódź, produkcja nie trafiła po imprezie do „dystrybucji”. Jak mówią nieoficjalne plotki, Cahir – koder demo – wykorzystał w nim fragmenty ścisłego tajnego raportu dotyczącego katastrofy smoleńskiej i z tego też powodu produkcja ta (wciąż) nie została udostępniona. Ponoć sprawę tę zbadać ma nawet komisja śledcza, która na pewno rozwieje całą tę „mgłę niejasności”.

Nie od dziś wiadomo, że amigowcy mają dwa ulubione zajęcia. Pierwsze z nich to przesuwanie ikonki na białe, a drugie to „czekanie na (amigowego) godota” w postaci nowego sprzętu, systemu, programu (niepotrzebne skreślić). Okazuje się, że scenowcy wcale nie są gorsi – jedni czekają na nowe demo TBL, a inni na... **demoZoo**. Jest to inicjatywa grupki



Góra: „Rink a Dink: REDUX” Lemon, dół: „Straff” Spaceballs



osób, za którą stoją Gasman i Menace, która pracuje nad – doprawdy rewolucyjną – internetową bazą demosceny. Portal ma zawierać wyniki z wszystkich imprez od 1980 roku, z możliwością wygodnego przeglądania grafiki, muzyki i dem każdej z nich. Ma zawierać wszystkie produkcje, ze screenshotami, soundtrackami, czytelnie posegregowane, według grup i indywidualnych autorów. Całość ubrana w lekki, nowoczesny design oraz otwarta do edycji dla chętnych. Projekt powstał już kilka lat, a jego finalizację zapowiedziano na koniec roku 2013. Osoby, chcące dowiedzieć się więcej, odsyłam do filmowego zapisu wykładu, który odbył się na Revision 2013 i jest dostępny na stronie <http://demozoo.org>.



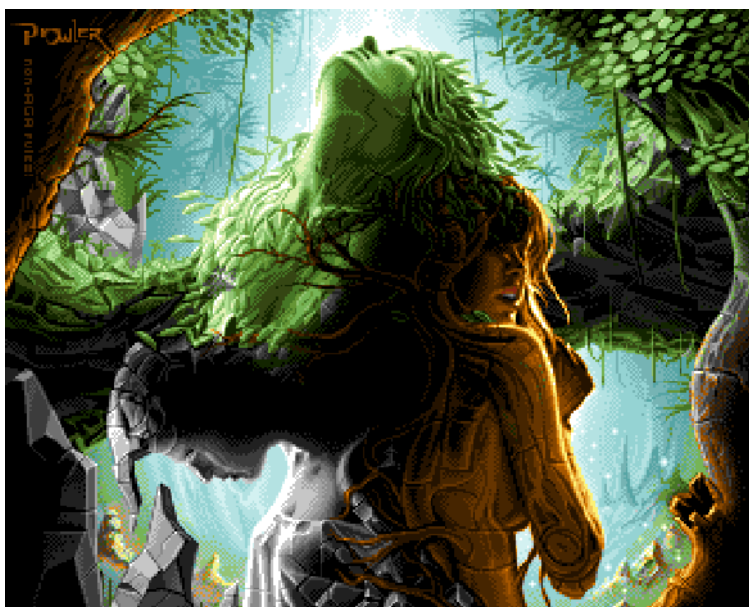
Retro Komp/Load Error Party 2013 to impreza, która odbyła się w Gdańsku pod koniec października. Niekwestionowanym bohaterem konkursowej części wydarzenia został Tygrys. Nie dość, że gra „Krunel”, którą zakodował na ZX Spectrum pokonała 12 innych gier w konkursie na najlepszą grę – w tym 8 pozycji na Amigę (więcej szczegółów w innym artykule w numerze), to jeszcze jego demo na tegoż ZX-a (sygnowane labelami Lamers & speccy.pl) wygrało połączony konkurs na najlepsze demo. Wspominam o tym z dwóch powodów. Po pierwsze – zwycięska produkcja „4D Demo 8” (kod – Tygrys, grafika – Atom, muzyka – Ziutek) jest remiksem amigowego dema grupy Lamers o tym samym tytule (moim zdaniem jest jednak znacznie lepsze niż oryginał), a po drugie w konkursie wystawiono też demo na Amigę 060. „Sepulka (preview)” nie poradziła sobie, co prawda, ze wspomnianym „4D Demo 8”, ale warto wspomnieć, że jest

to kolejny scenowy „powrót z zaświatów” – tym razem zaktywizowała nam się grupa Dinx Project. Ich demo (kod – Oster, grafika – Groh, muzyka – Mccnex) należy traktować raczej jako pierwszą „wprawkę” po latach nieaktywności i ponownym „wdrożeniu się w amigowy temat” niż poważną produkcję, ale warto je odnotować. Trzymam kciuki za pełną wersję (lub całkiem nową produkcję) wydaną choćby i na Revision 2014, bo potencjał w „odrodzonej” grupie z pewnością jest.

Co prawda mamy dopiero koniec listopada 2013, ale już teraz można napisać, że był to rok całkiem udany dla sceny OCS/ECS. Do dziś ukazało się w 2013 roku więcej produkcji na „pięćsetkę” niż w całym roku poprzednim. Oczywiście sporo z nich to typowe gnioty, ale coraz więcej dem to rzeczy solidne, jak choćby demo z **Assembly 2013** „Norwegian Pillow” teamu Dekadence, przyjemne intro „Funky Room” grupy Y-Crew (wydane na Gerp 2013), 4-kiłobajtowe intro „Piramida” grupy Scoopex z Datastorm 2013 czy music disk „Chip



64k intro „Thrice” grupy Dekadence



„15 Years of Fame” grupy Up Rough

Chop #15” wydany przez Desire. Do „klasyki gatunku”, prócz dem Lemon i Spaceballs, opisanych wyżej, przejdzie jednak jeszcze z pewnością jedna produkcja. Otóż, na coraz bardziej znaczącym demoparty **Datastorm**, które odbyło się w lutym 2013 w szwedzkim Göteborgu, najciekawszą amigową produkcją okazało się „15 Years of Fame” grupy Up Rough. Czysty, świeży, lekki design, niezły kod i doskonała grafika Prowlera (jednego z najlepszych współczesnych pixel artystów) zaowocowały jednym z ciekawszych dem na Amigę 500 w ostatnim czasie, które z powodzeniem może konkurować z klasycznymi tytułami. (kod – Booger, grafika – Optiroc, Prowler, Spot, TSC, muzyka – Qwan).

Finnish Amiga Party to – jak sama nazwa wskazuje – amigowa impreza, która odbywa

się w Finlandii, dokładnie w Helsinkach, na początku grudnia. Impreza, za którą odpowiada charyzmatyczny Nosfe, jest bardzo kameralna, ale powoli zaczyna zyskiwać coraz większą „przychylność mediów” i rozgłos. Poprzednia edycja imprezy (2012) przyniosła kilka ciekawych produkcji, między innymi interesujące dema na AGE „Gazer Unz” grupy Tracton (kolejne amigowe demo znanej pecetowej grupy) i „No more progress” grupy Jumalauta. Warto wymienić też niezłe demo na A500 „The End” grup Trilobit i FLO. Za kilkanaście dni odbędzie się trzecia edycja tej imprezy. Jestem tym bardziej ciekaw czy FAP 2013 znowu nas zaskoczy, szczególnie biorąc pod uwagę bardzo restrykcyjną zasadę, która ustanowił Nosfe, a która brzmi: „YOU MUST HAVE AT LEAST ONE ENTRY TO COMPETITIONS IF YOU COME TO THE PARTY.” Mam dziwne wrażenie, że w Polsce party z takim punktem regulaminu przyniosło by zerową frekwencję i kompletny brak produkcji. A jak będzie w Finlandii? Cóż, przekonamy się już niedługo.

Tomasz Pacyna (Slayer/Ghostown)
info@pixeldreams.pl



Produkcje z Finnish Amiga Party 2012