



Polskie Pismo Amigowe

Numer 3/2012 (9)

cena 19 zł

BVision vs Voodoo

BSzili's VADE

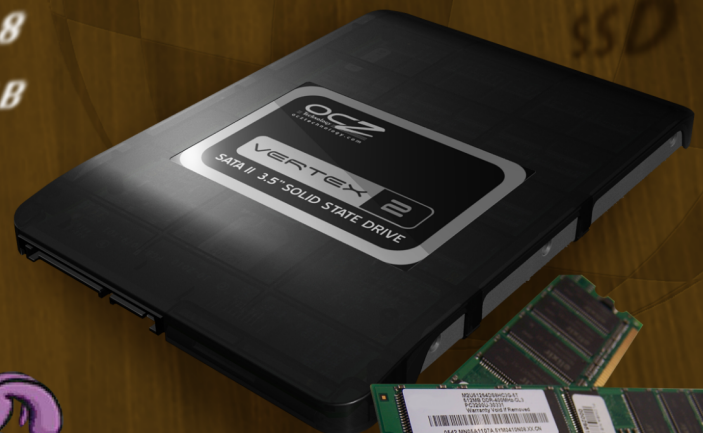
SLOT4 ZORRO

SOFTYS



HxC SD Floppy Emulator
EGS Spectrum 24/28
D-LINK DUB E100 USB
Viper 520 CD

SSD



SRAM



WAPAZ!

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 zapełnioną stronę pisma.

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

Termin nadsyłania artykułów upływa **31 stycznia 2013** roku.

Zapraszamy do udziału!



Polskie Pismo Amigowe

Redaktor naczelny: Sebastian Rosa.

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

Korekta: Aleksander Piotr Chyliński.

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.

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 2010–2012.
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 zapełnić numer, który właśnie trzymacie w rękach. W odróżnieniu od poprzedniej praktyki zdecydowaliśmy się docenić trud każdego z autorów za wkład w powstanie magazynu. Dzięki osobie, która chciałaby pozostać anonimowa, za każdą opublikowaną w magazynie stronę przewidzieliśmy tzw. „wierszówki”. Na chwilę obecną może jest to skromna kwota, lecz niech to będzie pewnego rodzaju sposób na wyrażenie wdzięczności. Być może w przyszłości darczyńców będzie więcej i autorzy za swoją publikację będą nagradzani nieco hojniej. Dziękujemy za nadesłane artykuły, a darczyńcy za ofiarowany datek.

W numerze

	Wiadomości	3
	Sprzętowa sztafeta Jafeta – część 1	5
	BVision vs Voodoo	7
	Zorro II Micronika	8
	Zorro IV + Mediator ZIV	9
	FastATA Zorro IV	10
	Wywiad z BSzili	12
	Sołtys	14
	Nasza pierwsza gra – kurs programowania pod AmigaOS – część 4	16
	Sprzęty trOLLO – część 1	18
	EGS Spectrum 24/28	20
	Karta sieciowa D-LINK DUB E100 USB w Amidze klasycznej	21
	Poradnik początkującego posiadacza Amigi – karty procesorowe dla A1200 ..	22
	Turbo „pięćsetka” – karta turbo Viper 520 CD	25
	HxC SD Floppy Emulator rev F	27
	Sqrxz (remake)	28

Od redakcji

Dziwięta odsłona naszego periodyku stoi pod znakiem sprzętu. Praktycznie został przez niego zasypany. Co jednak ciekawe, opisywane i recenzowane pozycje są wyjątkowo osobliwe, przez co, pomimo że sprzęt nie jest nowy, mogą spotkać się z dużym zainteresowaniem. Na początek pragnąłbym zwrócić uwagę na artykuły, które dotyczą slotów Zorro IV we współpracy z Mediatorem oraz FastATA. „Wynalazki” krakowskiej firmy Elbox wydają się być ciekawą alternatywą, aby stosunkowo niewielkim kosztem tchnąć nieco nowego ducha w leciwie Amigi klasyczne i dodatkowo mieć gwarancję jakości wykonania oraz obsługi technicznej, co w obecnych czasach, w przypadku amigowych akcesoriów, jest wyjątkową rzadkością. Dodatkowo, w numerze znajdziecie również porównanie kart graficznych BVision oraz Voodoo (gdyby ktoś stał przed dylematem) oraz artykuł o wyjątkowej perle w kategorii kart graficznych – EGS Spectrum 24/28 produkcji amerykańskiej firmy GVP.

Nie zapomnieliśmy również o początkujących – to do nich kierowany jest obszerny artykuł o możliwościach rozszerzeń, zarówno przez karty rozszerzenia pamięci, jak i karty procesorowe Amigi 1200. Będąc w temacie „dopalaczy” polecamy również opis kolejnego amigowego reliktu – karty turbo Viper 520 CD. „Doom” na A500? Tak, to możliwe.

Pozostając jeszcze przez chwilę w tematyce sprzętu warto odnotować dwa nowe cykle, któ-

re – mam nadzieję – przetrwają przez kilka numerów. Pierwszy to „Sprzętowa sztafeta Jafeta”. Na otwarcie serii mamy recenzję tabletu graficznego i jego współpracę z systemem MorphOS. Drugi cykl to „Sprzęty trOLLO”, gdzie Aleksander Giedyk stara się prezentować w zwięzły sposób nowości na rynku komputerowym i ich współpracę z Amigą, choć to nie jedyny cel. Debiut obejmuje opis współpracy z Amigą dysków SSD, zestawienie kompatybilnych monitorów LCD oraz kilka słów o pamięci SRAM w A600.

Dla tych, którym sprzęty wydają się obce, możemy polecić recenzję gry „Sołtys” – polska produkcja, która od niedawna wspierana jest przez ScummVM, oraz nowej gry dla Amig klasycznych „Sqrxz (Remake)” czy chociażby wywiad z BSzili – niezwykle płodnym w ostatnim czasie programistą, który m. in. jest autorem port UADE dla systemu MorphOS.

To oczywiście nie są wszystkie niespodzianki, które dla Was przygotowali nasi autorzy. Nie sposób ich wszystkich tutaj opisać, gdyż mija się to chyba z celem. Lepiej po prostu przekręcić stronę i oddać się lekturze. Mam nadzieję, że przyjemnej, a czas na nią przeznaczony będzie mile wspomniany i skłoni Was do napisania własnego artykułu, który zasiliłby łamy naszego pisma.

Sebastian Rosa



MorphOS 3.1 i zapowiedź 3.2

MorphOS Team udostępnił wersję 3.0, a następnie 3.1 tworzony przez siebie systemu operacyjnego. W najnowszej odsłonie dodano wyczekiwaną przez fanów obsługę laptopów PowerBook G4 produkcji Apple. Do systemu wprowadzono również szereg usprawnień, poprawek błędów (m. in. w Ambiancie, bibliotece Cairo czy sterownikach 3D) i nowych funkcji. Wśród nowości należy również wymienić zmianę w wysokości opłaty licencyjnej. Teraz wynosi ona odpowiednio: 111,11 euro dla laptopów i 79 euro dla pozostałych obsługiwanych komputerów (z wyjątkiem Efiki, dla której opłata rejestracyjna spadła do 49 euro). Dodatkowo podczas Planeta Amiga MorphOS Team zapowiedział, że MorphOS 3.2 będzie działać m. in. na komputerach zbudowanych w oparciu o płytę Sam460ex.

<http://morphos-team.net/>



AmigaOS 4.1 Update 5 i 6

Firma Hyperion Entertainment wydała piąte i szóste uaktualnienie systemu AmigaOS 4.1. Wśród poprawek i zmian w „piątce” znalazły się:

- Zoptymalizowane procedury DMA dla komputerów opartych na procesorach 440ep i 460ex
- Sterownik audio i mikser dla układu SM502 obecnego na płycie AmigaOne 500 (SAM 460ex)
- Uaktualnienie sterowników Warp3D dla kart Radeon (R200) oraz Permedia2
- Sterownik do kart Catweasel, umożliwiający obsługę stacji dysków, układu SID i joysticka
- obsługa MIDI za pomocą dołączonej do systemu biblioteki *camd.library*
- usprawnienie emulacji Amigi klasycznej (do systemu dołączono gotową do użycia kopię systemu w wersji 3.1).

Szóste uaktualnienie wprowadziło nowy sposób aktualizacji, który będzie przeprowadzany przez aplikację AmiUpdate.

Uaktualnienia są darmowe dla wszystkich zarejestrowanych użytkowników i dostępne jest dla każdej obsługiwanej platformy.

<http://blog.hyperion-entertainment.biz/?p=683>



ScummVM 1.5.0

Dostępna jest nowa wersja ScummVM. Najważniejsze zmiany to wprowadzenie obsługi kolejnych gier przygodowych. Fani tego gatunku na pewno ucieszą się z możliwości uruchomienia „*Once Upon A Time: Little Red Riding Hood*”, „*Backyard Baseball 2003*”, „*Blue Force*”, „*Darby the Dragon*”, „*Dreamweb*”, „*Geshia*”, „*Gregory and the Hot Air Balloon*”, „*Magic Tales: Liam Finds a Story*”, „*Sleeping Cub's Test of Courage*”, „*Sohtys*”, „*The Princess and the Crab*”. Ponadto wprowadzono poprawki i usprawnienia w poszczególnych silnikach odpowiedzialnych za różne interpretacje.

<http://scummvm.org/>



Sterownik RadeonHD

Firma A-EON Technology poinformowała o podpisaniu kolejnej umowy z Hanssem de Ruiterem zapewniającej dalszą kompatybilność AmigaOS 4 w zakresie dotyczącym obsługi kolejnych kart graficznych serii RadeonHD. Sterowniki opracowane przez Hansa pozwalają korzystać z kart graficznych pracujących na układzie R520 (X1000-1950), serii R700 (HD4000), Evergreen (HD5000), Northern Island (HD6000), Southern Island (HD7000), a także karty RadeonHD serii 7. Wspomniane znajdują wykorzystanie w działaniu z systemem AmigaOS 4.1 Update 5. Nowością jest również pełna implementacja zarządzania buforem pamięci wirtualnej, inteligentny shadow buffering na pamięci wirtualnej oraz uaktywnienie *RadeonHD_Rm.resource*. Zgodnie z zapewnieniami Hansa de Ruitera, obecna wersja jest w pełni gotowa na to, aby w systemie AmigaOS 4.x pojawiła się obsługa 3D.

<http://www.a-eon.com>



Individual Computers

Firma Individual Computers zapowiedziała nowy sprzęt dla klasycznej linii Amig. Są wśród nich trzy rozszerzenia, odpowiednio dla Amigi 500, Amigi 600 i Amigi 1200, jak również rozszerzenie pamięci dla większych amigowych modeli.

Karta **ACA500** wyposażona ma być w procesor 68000 z zegarem 14 MHz i 2 MB pamięci. Wbudowany na niej kontroler umożliwi podłączenie kart CF oraz napędów CD, oferując jednocześnie możliwość uruchomienia systemu komputera z poziomu karty CF nawet pod systemem AmigaOS 1.3. **ACA500** posiadać będzie również slot procesora kompatybilny z kartami rozszerzeń przeznaczonymi dla A1200, co pozwoli na podłączenie nawet ACA1231 i cieszyć się pod A500 szybszym procesorem. Przewidywana cena to 79,90 euro.

ACA1220 to urządzenie, które na pokładzie posiada procesor 68020 16,67 MHz i daje możliwość wetknięcia aż do 128 MB pamięci. Dodatkowo, dla posiadaczy **ACA500** (opisane powyżej) może służyć jako rewelacyjny „dopalacz” dla wysłużonej A500. Kartę będzie można opcjonalnie wyposażyć w zegar czasu rzeczywistego. Przewidywana cena: 89,90 euro (z zegarem za dodatkowe 19,90 euro).

W październiku na sklepowych półkach pojawiła się karta **ACA620** – zamiennik dla wyprzedanego modelu ACA630. Dzięki niej A600 korzysta z dobrodziejstw procesora 68020 16,67 MHz wraz z funkcją maprom. Na pokładzie karty znajduje się nawet 12 MB pamięci + od 1 do 3 MB zarezerwowanych na maprom (w zależności od rewizji). Cena: 99,90 euro.

Firma nie zapomina także o posiadaczach „dużych” Amig klasycznych i wkrótce będzie oferować rozszerzenie pamięci **BigRamPlus**, które pozwoli dolożyć do A3000(T) i A4000(T) 256 MB pamięci. Karta wpinana w sloty Zorro III ma być dostępna w cenie 99,90 euro.



Timberwolf w fazie RC

Timberwolf to nazwa projektu mającego na celu stworzenie dla AmigaOS portu przeglądarki internetowej Firefox. Aby zrealizować ten cel, we wrześniu 2009 roku otwarto bounty. Założenia obejmowały stworzenie portu, który będzie niezależny od jakichkolwiek rozwiązań emulujących inne środowiska, np. X serwer czy GTK. Miała to być aplikacja działająca na i wykorzystująca wszystkie specyficzne cechy systemu AmigaOS. Stawiane wymagania minimalne obejmowały działanie programu na AmigaOS 4.1, choć dozwolone było korzystanie z rozwiązań trzecich (np. inny, zewnętrzny zestaw wskaźników myszy). Bounty zostało przypisane Hansowi-Joergowi i Thomasowi Friedenom. Bounty zostało uznane za ukończone, a kwota 6732,72 euro została wypłacona. Nie oznacza to bynajmniej, że zakończą się dalsze prace nad przeglądarką. Program po kolejnej, tzw. „czwartej wersji beta” wszedł w stadium „release candidate”. Wersje RC pojawiły się już trzy. Nie stanowią one jakiegoś przełomu – stale eliminowane są z nich zauważone błędy i tylko czasami pojawiają się drobne usprawnienia, takie jak dostosowanie interfejsu użytkownika do systemowej czcionki, obsługę MIME na potrzeby menadżera pobierania plików (np. otwieranie z jego poziomu pobranych archiwów). Powinna być również zauważalna niewielka poprawa prędkości działania dzięki kompilacji przy wykorzystaniu nowej wersji biblioteki Cairo, jak również poprawne działanie bazy zakładek na systemie plików JXFS. Jednocześnie Freidenowie zaznaczają, że kolejnym elementem prac rozwoju **Timberwolta** będzie uaktualnienie kodu źródłowego do wersji 14.0 Firefoxa. Dodatkowo, po wydaniu AmigaOS 4.2 zakłada się implementację OpenGL i WebGL w Timberwoltie.

<http://os4depot.net/share/network/browser/timberwolf.lha>



Koniec sklepu AirSoft Softwair

Z powodów osobistych właściciela z dniem 31 lipca 2012 sklep internetowy AirSoft Softwair został zamknięty. Tylko do tego dnia były przyjmowane zamówienia na produkty firmy, czyli **Hollywood** oraz **Hollywood Designer**. Nie oznacza to jednak końca rozwoju programu. Autor zapewnia, że nadal będzie pracować nad swoimi programami i będą pojawiać się uaktualnienia – nie będzie jedynie możliwości ich nabycia. Na chwilę obecną nie można określić czy zawieszenie działalności sklepu jest permanentne, czy też tymczasowe.

<http://www.airsoftsoftwair.com>



Sprzętowa sztafeta Jafeta

Część 1 – radosne smyranie, czyli zabawy z tabletem

Każdy, kto próbował naszkicować coś na komputerze metodą „odręczną” dość szybko spostrzegł, że myszka nadaje się do tego dość średnio. Szczęściem współczesna cywilizacja zna rozwiązanie tego problemu, a imię jego – tablet. Współcześnie stosowane tablety graficzne (oprócz możliwości traktowania ich jak zwykłą kartkę papieru) oferują też wykrywanie siły nacisku, przyciski funkcyjne oraz ziliard innych funkcji, w które nie będziemy się na razie wgłębiać. Ważne, że są i dają się z powodzeniem wykorzystać. Czy da się jednak takie urządzenie podłączyć do naszych ulubionych komputerów, a jeśli nawet podłączyć da się to urządzenie zadziała?

Pelen czarnych myśli rozpakowywałem zdobywczy (braterska pomoc od niejakiego Chama) tablet, a był to niedrogi Trust TB-1100, czyli model który – mówiąc szczerze – nie jest urządzeniem ze szczególnie wysokiej półki. Darowanemu wielbłądowi nie patrzy się jednak w zęby, tylko od razu podłącza do portu USB (zwłaszcza że instrukcja o MorphOS-ie wspomina bardzo niewiele, a konkretnie – nic).

Zanim jednak odnalazłem wolny port w mojej maszynie, zdążyłem omieść wzrokiem posiadane urządzenie. Jest to niewielka zaokrąglona tabliczka o wymiarach ok. 22 cm szerokości, 19 cm długości i 1 cm wysokości (wykonana, dodajmy, z dość kiepskiej jakości srebrnego plastiku). Umieszczona pośrodku przestrzeń robocza ma wymiary 12,7 x 10,3 cm, przy czym faktyczny obszar, z którego tablet przyjmuje dane, jest ograniczony dodatkowym około centymetrowym marginesem (faktyczna przestrzeń robocza ma więc wymiary 101 x 76 mm). Folia, która go pokrywa daje się unieść tak, że umożliwia podłożenie pod spód np. fotografii, co umożliwia obrysowanie jakiegoś jej elementu za pomocą rysika i wprowadzenie go w ten sposób do programu graficznego. U góry tabletu umieszczono zagłębienie, w którym można (z braku lepszego miejsca) umieścić niepotrzebny chwilowo, bezprzewodowy (zasilany baterijką AAA) rysik. Tuż obok niego znajduje się miejsce, z którego wychodzi kabel USB. A skoro już o kablu mowa, to wreszcie wymacałem z tyłu obudowy „Miniacza” pusty port, w który z niemałą satysfakcją wrażliwym kończącą go wtyczkę.

Zadymiło, zahucztało i na ekranie Ambienta ukazał się requester Poseidona. „Wykrył, skubańcuchi!” – ucieszyłem się widząc, że tablet rozpoznawany jest prawidłowo jako urządzenie interfejsu HID. Szybko chwyciłem rysik w swe spoczone łapska i zacząłem go przesuwając nad powierzchnią tabletu. „On się rusza, on naprawdę się rusza!” – przemknęło mi przez myśl, gdy obserwowałem, jak wskaźnik myszki przesuwa się po ekranie. Perspektywę nieskrępowanego mazania i smyrania zakłóciło jednak spostrzeżenie, że odległość, o jaką przesuwa się kursor, nijak ma się do skali mojego tabletu. Nie byłem w stanie w żaden sposób przekroczyć jakiegś magicznej bariery, ograniczającej mnie do malutkiego kwadracika



w górnym, lewym rogu ekranu. „Masz dwa wyjścia” – powiedziałem do siebie stanowczo: „Albo zwariować, albo zacząć grzebać w opcjach Poseidona”. Wybrałem to pierwsze. Po tygodniu chodzenia na rzesach i posługiwania się niezrozumiałym belkotem, wyciszony solidną porcją neuroleptyków uruchomiłem jednak wreszcie ustawienia USB.

Po chwili konfuzji i oglądania listy urządzeń ze wszystkich stron okazało się, że Poseidon ochrzczył mój tablet wielce wymowną nazwą „APT-USB-3”. Pokrzepiony tym faktem zaznaczyłem sprzęt na liście, wduśliłem przycisk „Settings” i w niecałe ćwierć sekundy później cieszyłem swe oczy oknem konfiguracyjnym dla klasy HID. Próbując się rozeznac w gąszczu niezrozumiałych ustawień („nie staraj się zrozumieć, tylko kombinuj tak, żeby działało” – powiedział mi kiedyś Cham, a starszego brata się słucha) kliknąłem w końcu na zakładkę „Action”. Zamiast spodziewanej akcji natrafiłem jednak na taką gmatwaninę opcji, że gotów byłbym już rzucić to wszystko w diabły i zająć się jakimś spokojniejszym zajęciem, jak na przykład wypas owiec. W tym samym jednakże momencie usłyszałem nagle głos, a brzmienie miał jak ze spiznu: „Nie czyni tego, Jafecie, gdyż naczelny nigdy nie domknie tego numeru PPA i skończy na dobre z amigowaniem, że nie wspomnę o wierszówce, która Ci przepadnie. Zamiast tego zainteresuj się zwracanymi przez tablet wartościami dla osi X i Y przy wybranym Stylus (Digitizer) na liście „Reports and collections”. Posłuszny poleceniu zerknąłem na oczekiwane przez Poseidona zakresy wartości dla „X axis” i „Y axis”, następnie przełączyłem ptaszek przy „Report current values”, posmyrałem jeszcze trochę po tablecie i doznałem olśnienia: stos oczekiwał wartości z zakresu od 0 (min) do 12000 (max), a mój tablet z ledwością dociągał wartości (val) do 1999 w poziomie i 1499 w pionie! Pozostało już tylko uświadomić Poseidonowi, jak straszliwie się pomylił preceniacząc rozdzielczość

mojego Trusta. Postanowiłem dokonać tego za pomocą przycisku „Clip” i kalkulatora. Działek 1999 oraz 1499 przez 12000 otrzymałem odpowiednio 0,16658 i 0,12492, zatem dla parametru „Clipping” zaznaczyłem „Stretch” i ustawiłem suwakami wartości procentowe od 0 do 17% (dla osi X) oraz 12% (dla osi Y). Możecie sobie wyobrazić moją radość, gdy: po pierwsze – Poseidon nie zawiesił się, gdy wprowadziłem wspomniane ustawienia, po drugie – nie nic wybuchło, gdy je zapisałem, a po trzecie – wszystko nareszcie zaczęło działać tak, jak należy (to znaczy nie do końca, gdyż jak już pewnie zauważyliście – wartości procentowe podałem z pewnym przybliżeniem, zatem mój obszar roboczy nie jest idealnie dopasowany do wielkości tabletu; nie są to jednak jakieś astronomiczne wartości (tracimy 49 niedociągniętych punktów w poziomie i 41 przyciętych w pionie), mnie to jednak wystarcza (co nie znaczy, że nie jestem chętny poznać lepszego rozwiązania tego problemu, o ile ktokolwiek wpadnie na takie).

Napisałem „wszystko zaczęło działać”? No dobra, nie całkiem. Do pełni szczęścia potrzebowałbym jeszcze poprawnej obsługi przycisków na rysiku oraz wsparcia dla przycisków funkcyjnych umieszczonych u góry przestrzeni roboczej tabletu. W kwestii przycisków sprawa jest złożona. Poseidon zgodnie z prawdą widzi trzy: pierwszy to tzw. „tip switch”, czyli nacisk na tablet (tu nic nie musiałem zmieniać – przełączenie domyślnie rozpoznawane jest jako lewy przycisk myszy), drugi to tzw. „barrel switch” i ten za Chiny ludowe nie chce u mnie działać (tzn. przypisany jest do prawego klawisza myszy, ale przełączanie nic nie daje. „Report current values” też nie widzi jego wciśnięcia. Być może jednak uległ on uszkodzeniu w wyniku licznych crash-testów wykonywanych na urządzeniu przez moje pociechy – na poparcie tej hipotezy świadczy fakt, że przycisk jest niesprawny, nawet gdy podłączam tablet do komputera PC). Trzeci („tablet pick”), do-



myślnie ustawiony jako środkowy przycisk myszy, musiałem więc przełączyć na prawy, żeby dało się w miarę wygodnie pracować np. z Personal Paintem, które zakładają pod prawym przyciskiem myszy rysowanie kolorem tła/drugorzędnym. Co do przycisków funkcyjnych – Poseidon niby widzi jakieś (co dziwne, podaje że jest ich 16, podczas gdy na tablecie doliczyłem się tylko ośmiu), ale nie udało mi się na razie zmusić ich do współpracy. A szkoda, bo ustawienia stosu USB pozwalają na naprawdę fantastyczne przypisanie wartości.

Skoro wspominałem o tym, że urządzenie przeszło już w swym życiu niemało, to wypada mi pochwalić firmę Trust za okazane wsparcie – otóż zdarzyło się, że rysik, który otrzymałem w zestawie, został nieodwracalnie uszkodzony (jeden z moich wielbłądów podeptał go, a następnie przeżuł i wypluł do wiadra z wodą). Pogodziłem się już, że sprzęt należy spisać na straty, ale napisałem jeszcze maila do działu obsługi klienta i nawet dostałem odpowiedź. Trochę łamaną polszczyzną, ale pracownik firmy Trust odpowiedział, że przyślą mi do domu nowy rysik. Wystarczyło podać adres i wkrótce mogłem znowu cieszyć się sprawnym sprzętem. Tak trzymać, panowie (i panie) z Trusta!

No dobrze, ale my tu nawijamy o kwestiach technicznych, a nawet reklamujemy producenta, artykuł prawie się skończył, a jeszcze nie wspominałem, jak sprawuje się przetestowane z tabletem oprogramowanie. No to lecimy:

- Personal Paint – działa bez zarzutu, rysuje się bardzo wygodnie (świetna zabawka dla dzieciaków), szkoda tylko, że nie ma obsługi nacisku, no ale nie wymagamy zbyt wiele.
- ImageFX – właściwie to samo, choć (dzięki zaawansowanym opcjom kontroli pędzla i retuszu) dostajemy do ręki coś więcej. Jeśli jednak chcemy coś po prostu wypikselować, to w mojej opinii Personal nie ma konkurencji.



PPaint i szybki bazgrołek agenta Bogdana

- Sketch – obiecujący program do prostego rysowania, tym bardziej że obsługuje siłę nacisku i daje możliwość bezproblemowej pracy z przezroczystością. Niestety, ma kilka uprzykrzających życie bugów, które (miejmy nadzieję) da się jednak poprawić.
- Gribouillis – bez wątpienia program dający naszemu tabletowi (a właściwie nam, dzierżącym rysik) największe możliwości (od samych możliwości kontroli pędzla dostaniem ataku histerii). Jednocześnie jest najbardziej „zasobożernym” z nich, ale nie będziemy przecież z jego pomocą wykrywać nic filtrami, więc prędkość działania jest jak najbardziej akceptowalna.

Podsumowując, opisywany sprzęt na pewno nie jest skierowany do profesjonalistów (a nawet – rzekłbym – półprofesjonalistów). Przestrzeń robocza jest niewielka, rozdzielczość dość niska, a jakość wykonania – hmm... mogłaby być lepsza. Na dodatek, niestety, nie

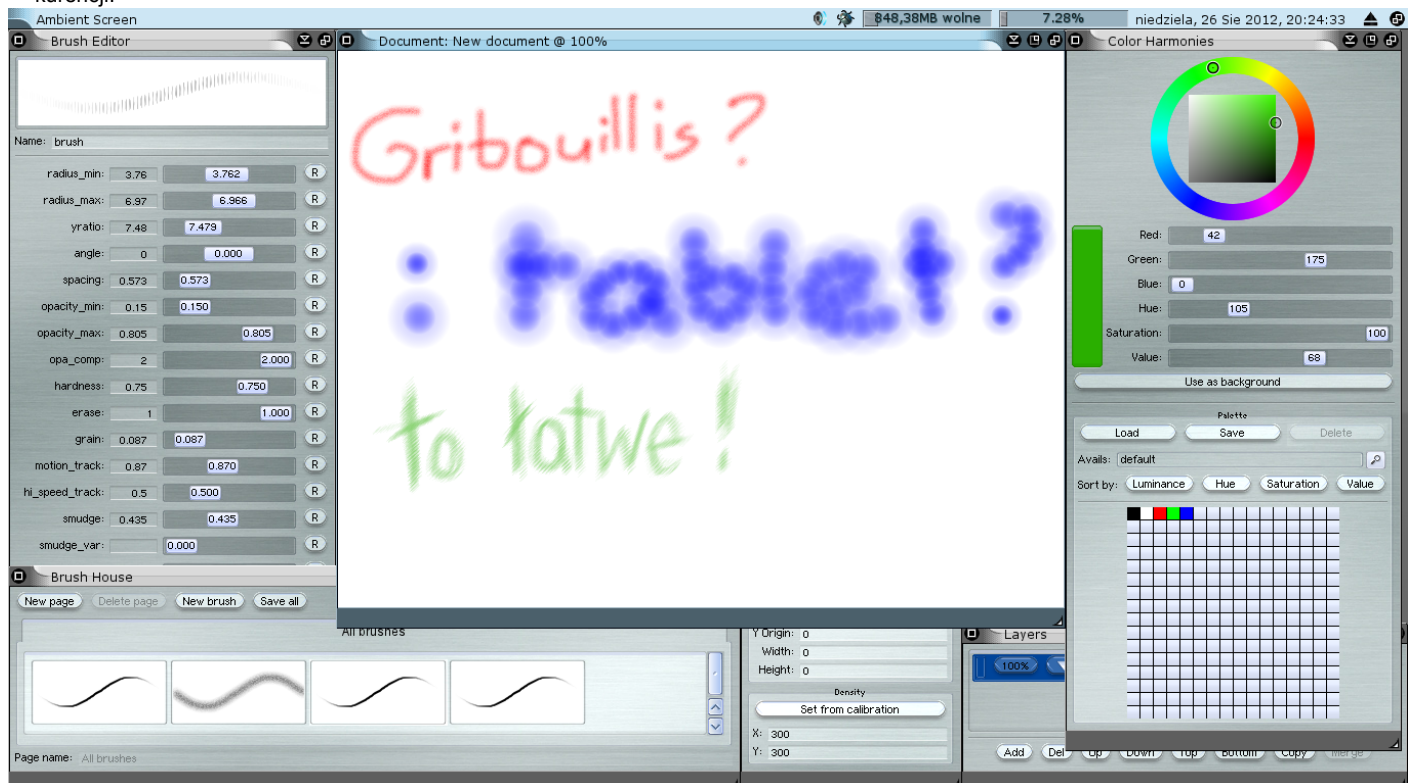
wszystkie udostępniane przez tablet opcje daje się wykorzystać pod systemem MorphOS (choć być może część zauważonych niedociągnięć wynika raczej z mojego bezapelacyjnego lamerstwa). Niemniej jednak urządzenie jest, działa i daje niemałą satysfakcję z użytkownika.

Jafet

Wireless Tablet TB-1100

Kod EAN: 8713439120509
Waga produktu: 545 g
Powierzchnia robocza: 76x101mm

<http://www.trust.com/products/product.aspx?artnr=12050>



Gribouillis w akcji



BVision vs Voodoo

Stoisz przed dylematem, którą kartę graficzną wybrać? Nie wiesz, która jest lepsza, a która gorsza? W tym artykule postaram się przedstawić wszystkie wady i zalety kart graficznych BVision oraz Voodoo 3 PCI, a także pomóc dokonać wyboru tym, którzy wahają się nad jednym z rozwiązań.

BVision

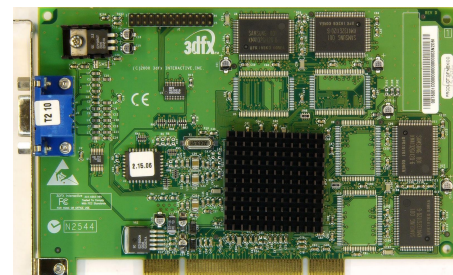
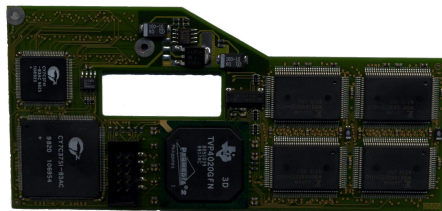
BVision to karta graficzna wpinana do slotu rozszerzeń karty Blizzard PPC. Ma to zarówno wady, jak i zalety. Zaletą jest fakt, że karta komunikuje się bezpośrednio z kartą turbo, co korzystnie wpływa na szybkość. Poza tym karta jest dosyć kompaktowa, dzięki czemu nie koliduje z większością istniejących amigowych rozszerzeń. Możemy mieć w Amidze zainstalowane sloty Zorro czy nawet Mediatora i BVision absolutnie nie będzie z tym kolidowała. Wadą jest niewątpliwie fakt, że BVision wymaga Blizzarda PPC i z niczym innym nie będzie współpracować.

Karta jest bardzo droga. Kosztuje więcej niż używany Mediator do Amigi 1200, mimo że jej możliwości nie są oszałamiające. Jednak ma coś, czego inne karty nie mają – BVision umożliwia bezproblemowe promowanie obrazu. Oznacza to, że karta wyświetla obraz „Boot Menu” Amigi oraz tryby PAL na ekranie monitora VGA bez konieczności instalacji jakiegokolwiek oprogramowania czy osprzętu. Jest to bardzo ważna zaleta, ponieważ przy awarii systemu nie musimy biegać z monitorem działającym w trybie PAL i szukać przyczyny usterek. Wszystko naprawimy, jeśli nie w „Boot Menu” to w trybie awaryjnym „no startup-sequence”. Tego nie potrafi Voodoo ani żadna inna karta graficzna bez scandoublera lub dyskietki ratunkowej promującej obraz Amigi do VGA. W pewnym stopniu potrafi to zrobić AmigaOS 3.9 na chipsecie ECS/AGA po załadowaniu ROM Update z Boing Bag 2, lecz jest to opowieść na osobny artykuł. Muszę tutaj jednak ostudzić zapał graczy – owe promowanie karty BVision nie działa z grami, które pracują tylko w trybie PAL. Automatyczne promowanie następuje tylko i wyłącznie dla ekranów systemowych.

Dużą zaletą BVision jest jej uniwersalność. Karta współpracuje nie tylko z AmigaOS 3.x, ale również z AmigaOS 4.x, systemem MorphOS oraz, na chwilę obecną, w eksperymentalnej wersji NetBSD. Powiem szczerze – nie orientuję się, jak wygląda sprawa z Linuksem.

Voodoo

„Wudzielec” jest kartą PCI. Oznacza to, że jeśli chcemy używać jej w Amidze, musimy posiadać jeden z trzech dostępnych mostków PCI. Do wyboru są G-Rex, Prometheus oraz Mediator. G-Rex, podobnie jak BVision, wymaga karty turbo Blizzard PPC. Jest to mostek



BVision i Voodoo

PCI wpinany w to samo złącze co BVision. Jego dużą zaletą jest to, że współpracuje zarówno z AmigaOS 3.x, jak i systemem MorphOS oraz eksperymentalnie z NetBSD. Wadą jest brak współpracy z AmigaOS 4.x. Mostek ten jest trudno dostępny, gorzej oprogramowany, a jego software nie jest już rozwijany. Podobnie sprawa wygląda z Prometheusem. Aby go używać, musimy mieć w Amidze sloty Zorro III oraz kartę turbo z Amigi 3000/4000. Mediator natomiast zadowolony się kartą turbo z procesorem 68040 (działa na niektórych 68030, ale szczerze mówiąc mija się to z celem), choć nie pogardzi też procesorem PPC, na którym Voodoo rozwija skrzydła. Jeśli ktoś chciałby więcej poczytać o mostkach PCI dostępnych dla Amigi, to zachęcam do lektury artykułu „Zorro vs PCI” z siódmego numeru naszego pisma. Temat został tam szerzej opisany.

Voodoo niestety nie potrafi promować ekranów w trybach PAL, tak jak robi to BVision. Małym wyjątkiem jest tutaj AmigaOS 4.x, który po załadowaniu kernela potrafi na tej karcie wyświetlić „Boot Menu” Amigi. Nie ma też większych problemów z nabyciem karty. Co prawda nie zalewa ona portali aukcyjnych ani działu Giełda, to nie stanowi ona takiego rzadkiego rarytasu, jakim jest BVision. Karta jest również tania – można ją nabyć w kwocie poniżej 100 PLN.

W boju

Na wstępie napiszę, że używałem BVision pod kontrolą CyberGraphics 4, a Voodoo 3, wpięte do Mediatora ZIV, pod Picasso96. Karta turbo była taka sama dla obydwu konfiguracji – Blizzard PPC. Nie robiłem też żadnych konkretnych testów – wszystko właściwie sprawdzałem na „przysłowiowe oko”. Bardziej interesuje mnie praktyczne działanie aniżeli słupki z testami.

W trybie 2D obie karty zachowywały się właściwie podobnie z niewielką przewagą dla Voodoo. Jednak nie odczułem jakiejś miazdzącej przewagi. Bardzo podobało mi się promowanie „Boot Menu” na BVision – szkoda, że Voodoo tego nie potrafi, gdyż to się naprawdę przydaje w kryzysowych momentach. Nie trzeba kombinować z dwoma monitorami. Zarówno gry, jak i programy działały tak samo dobrze na BVision i Voodoo – przetestowałem „Exodus”, „Quake”, „Quake II”, „Myst”, „Payback”.

Zupełnie inaczej było z trybami 3D. Miałem spore problemy z poprawnym skonfigurowaniem Warp3D. BVision pokazała „Gears Test 3D” po stosunkowo krótkich bojach, natomiast Voodoo z Mediatorem ostro stawiała opór. Warp3D działał, ale tylko w trybie 68k – na PPC nie chciał się uruchomić. W końcu jednak udało się, choć konfiguracja była żmudna i trudna. Przetestowałem „Payback” – było wiadać przewagę Voodoo, choć nieznacznie. Natomiast „Quake” i „Quake II” dostały bardzo wyraźnego kopa. BVision nie dała rady uciągnąć 640x480 ani tym bardziej 800x600 – Voodoo poradziła sobie bez problemów. Zarówno w „Quake”, jak i „Quake II” miałem ponad 20 klatek na sekundę. Inne gry 3D, takie jak „Freespace” czy „Hexen II” również ukazywały przewagę Voodoo. W „WipeOut” było to trochę mniej widoczne.

Nie samymi jednak grami człowiek żyje. System operacyjny AmigaOS 4 również pokazuje przewagę Voodoo nad BVision. Jest to zauważalne nawet w trybie 2D. Warto również zaznaczyć, że Mediator TX daje dodatkowego „kopa” karcie Voodoo w trybie 3D pod AmigaOS 4.x. Dodatkowo, jak już wcześniej wspominałem, AmigaOS 4 promuje po załadowaniu kernela „Boot Menu”, tak więc topnieją właściwie wszystkie zalety BVision. Warto też zaznaczyć, że BVision posiada 8 MB pamięci graficznej i jest oparta na procesorze Permedia, a Voodoo 3 wyposażone jest w 16 MB i jest dużo wydajniejszym akceleratorem.

Czy w takim razie BVision jest całkowicie dyskwalifikowane? Z całą pewnością nie. Wszystko zależy od naszych potrzeb. Chcąc używać systemu MorphOS pod „klasykiem” czy NetBSD pozostaje nam tylko BVision. Można też użyć starszych kart typu CyberVision, jednak wymagają one slotów Zorro i są niewiele tańsze od BVision. Jeśli jednak interesuje nas głównie AmigaOS 4 lub 3.x i zależy nam na wydajności, to proponuję zainteresować się Mediatorem i kartą Voodoo. Poza Voodoo możemy podłączyć do niego również Radeona 9200, co już jest „skokiem w kosmos”. W tym przypadku jednak prawdziwą przyjemność z korzystania będą mieli użytkownicy AmigaOS 4. Niestety wsparcie 3D nie działa na starszych wersjach tego systemu, ale to również temat na osobny artykuł.

Benedykt Dziubałowski



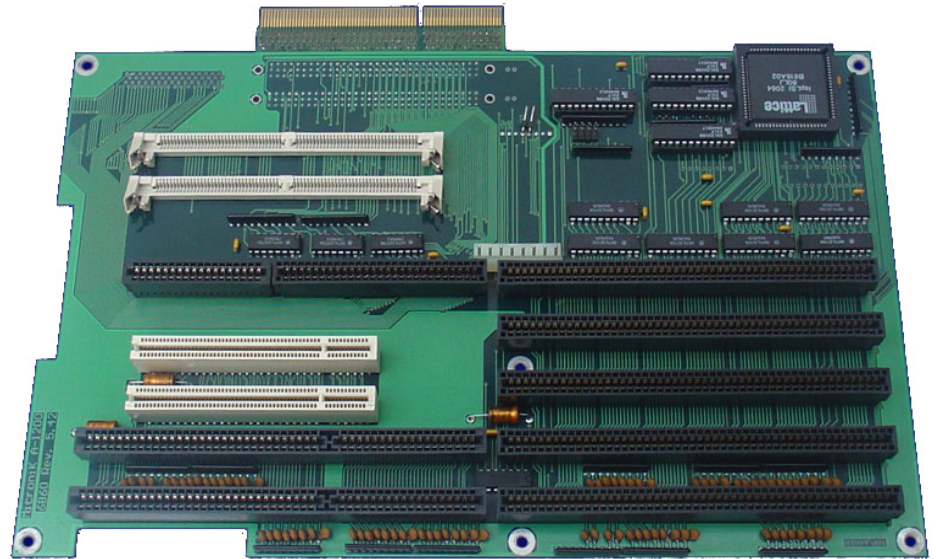
Zorro II Micronika

Wiele lat temu, świętej pamięci firma Micronik produkowała sloty Zorro II do Amigi 1200. Była to rzecz ciekawa, ponieważ można było używać części kart z Amigi 2000 oraz 4000. Posiadacze Amigi 1200 mogli podłączyć sobie kartę graficzną Picasso II lub IV, CyberVision 64 oraz wiele innych wcześniej dla nich niedostępnych. Sloty te przez cały okres ich produkcji były jednak stale ulepszone. Wszystkich wad nie udało się wyeliminować – jak choćby znane problemy konfliktu z kartami rozszerzeń Apollo 040/060. Trzeba było, jak zwykle, kombinować, do czego amigowcy zdążyli przywyknąć.

Zwykle sloty Zorro II wyposażone są w szereg mniej lub bardziej ciekawych gniazd. Poza samymi slotami Zorro II w ilości pięciu sztuk znajdziemy również sloty ISA oraz PCI (w zależności od wersji), slot video, złącze zasilania oraz port aktywacyjny slotu video. Ciekawostką są zwłaszcza te dodatkowe (ISA, PCI oraz slot video), które są praktycznie bezużyteczne – działają tylko wtedy, gdy mamy specjalną, dedykowaną Amidze kartę będącą sprzętowym emulatorem komputera PC. Za pomocą takiej karty możemy aktywować sloty ISA i używać ich ze wspomnianym emulatorem. Powstało też kilka sterowników, które umożliwiały używanie kart sieciowych z PC pod systemem Amigi. Najprawdopodobniej firma Micronik zdecydowała się na instalację slotów ISA oraz PCI ze względu na to, że zarówno Amiga 2000, jak i 3000 oraz 4000 miały je na swoim pokładzie. Niestety bez sprzętowego emulatora czy mostka Zorro <-> ISA zwanego GoldenGate nic nie zrobimy. Powodem tego jest fakt, że owe sloty nie mają fizycznego kontaktu z magistralą Amigi. Jeśli przyjrzymy się uważnie tym slotom, to zauważymy, że jedyne co jest do nich podpięte to... zasilanie.

Sloty PCI są jeszcze bardziej bezużyteczne niż ISA. Sądzę, że dodano je prawdopodobnie w nadziei wykorzystania ich w przyszłości. Z tego, co się orientuję, żaden z amigowych emulatorów nie robi z nich użytku. Testowałem emulator z procesorem 386 (nie wiem jak wygląda sprawa 486, ale sądzą, że podobnie). Możemy jednak w te sloty włożyć współczesny komputer połówkowy i cieszyć się dosyć potężnym pecetem wpiętym w naszą Amigę. Osobiście jednak znalazłem dla tych slotów inne przeznaczenie – wpinałem w nie kartę USB wyposażoną w pięć portów. W ten sposób mogłem ładować dowolne urządzenie bez potrzeby zajmowania portów USB w PC czy Macintoshu. Czy już słyszycie te pytania osób, także amigowców „czy to działa”?

Slot video jest w wersji AGA, co nie powinno dziwić. Jest on jednak nieaktywny. Aby go uaktywnić, potrzebujemy urządzenia zwanego „aktywatorem slotu video” – specjalny kabelek plus trochę elektroniki. Nakładany on jest na układy komputera i wpinany do specjalnego gniazda w slotach Zorro. Mała uwaga – nie wszystkie wersje slotów Zorro posiadają owe złącze. Po uaktywnieniu slotu możemy sobie włożyć w to złącze scandoubler lub genlock. Niestety aktywator jest trudno dostępny na rynku wtórnym.



Zorro II

Niektóre wersje slotów Micronika mają również gniazda SIMM, w które, jak łatwo się domyślić, możemy włożyć kości SIMM o pojemności maksymalnie 8 MB. Warunek jest jednak taki, aby żadna karta turbo nie była podpięta do Amigi. Wydaje się to bezsensowne rozwiązanie, ale mnie się zdarzyło używać takiego rozszerzenia pamięci przez pewien czas. Warto zaznaczyć, że sloty Zorro z gniazdami SIMM są tańsze niż rozszerzenie pamięci do Amigi. Dodatkowym atutem w tym przypadku jest to, że całość uzyskujemy w jednym komplecie: sloty i możliwość powiększenia ilości dostępnej pamięci. Brakuje jednak dwóch rzeczy: zegara czasu rzeczywistego i miejsca na koprocessor. Jednak bez tego można się obejść.

Złącze zasilania to typowy patent Micronika – niezgodny z niczym innym produkowanym przez firmy trzecie. Pół biedy, gdy posiadamy obudowę tower od Micronika – wówczas wtyczka oryginalnego zasilacza jest zgodna z opisywanymi slotami Zorro II. Znika również problem z posiadaniem tulejek dystansowych – są one już w obudowie wmontowane na stałe. Nic tylko włożyć, przykręcić i podłączyć. Pamiętać tylko trzeba o nadbudowie do obudowy Infnitiv – konieczna jest, gdy chcemy dołączyć do Amigi jakąś kartę turbo i jednocześnie zamknąć obudowę.

Użytkownicy innych obudów muszą kombinować. Najprościej jest zaadaptować zasilacz AT w wtyczkę Micronika i przeloczyć piny zasilania do ich patentu. Nie jest to wcale trudne. Bez problemu można znaleźć schematy zarówno zasilania Micronika, jak i zasilacza AT. W końcu są to dokładnie te same zasilacze różniące się jedynie wtyczką. Należy pamiętać, że dzięki takiemu podejściu rozwiązujemy problem „braku prądu” w Amidze – niestraszne nam są wtedy prądożerne karty turbo. Do montażu brakuje jeszcze tulejek dystansowych. Od biedy można je sobie darować, ale wtedy sloty Zorro II wiszą jedynie na złączu karty procesorowej, co nie jest ani dobre, ani zalecane. Tulejki można łatwo zdobyć – wystarczy zmierzyć

jak długie muszą być i udać się do dowolnego sklepu ze śrubkami.

Same sloty Zorro II nie oferują praktycznie niczego nadzwyczajnego. Karty włożone do tych slotów pracują tak samo, jak karty włożone do Amigi 2000 (Picasso II lub CyberVision 64/3D). Inne karty, takie jak np. TandemIDE również nie sprawiały problemów. Problem pojawił się przy próbie uruchomienia kontrolera SCSI dla slotów Zorro II. Testowałem A2092 oraz Oktagon 2008 – żaden z nich nie chciał współpracować ze slotami Zorro II wpiętymi do Amigi 1200. Problematyczna jest również, wspomniana wcześniej, współpraca Zorro II z kartami turbo Apollo. Standardowy problem jest taki, że Amiga „nie widzi” pamięci zainstalowanej na karcie Apollo, jeśli ta jest wpięta w Zorro. Podobnie problematyczna jest obsługa karty graficznej CyberVision 64/3D w połączeniu z Apollo i ROM-em 3.1. Zwykle Amiga nie uruchamia się w takiej konfiguracji (choć to jednak loteria). Odmian slotów Zorro II powstało wiele i nie wszystkie sprawiały problemy z kartami Apollo.

Czy warto kupować sloty Zorro II Micronika? Wszystko zależy od tego, jakie mamy plany. Jeśli interesuje nas pozbycie się problemów „prądowych” plus ewentualnie jakieś małe rozszerzenie (np. Tandem IDE), to Zorro II Micronika to dobra i względnie tania inwestycja. Niemniej jednak budowanie Amigi 1200 w mocniejszej konfiguracji z kartą graficzną, muzyczną opartej na Zorro II trochę mija się z celem. Lepiej zainwestować w sloty Zorro IV produkcji krakowskiej firmy Elbox. Jeśli jednak chcemy postawić na względnie niską cenę i jak najszybszą konfigurację, to warto zainteresować się slotami PCI w postaci Mediatora. Trzeba pamiętać, że choć samo Zorro II zbyt drogie nie jest, to już same karty są czasami nawet bardzo drogie.

Benedykt Dziubałowski



Zorro IV + Mediator ZIV

Dawno temu Krakowianie znad Wisły wyprodukowali dosyć ciekawe sloty Zorro przeznaczone do Amigi 1200. Slotów Zorro było już trochę wtedy na rynku (sukcesywnie produkowała je firma Micronik), jednak wyrób Krakowiaków był zdecydowanie lepszy.

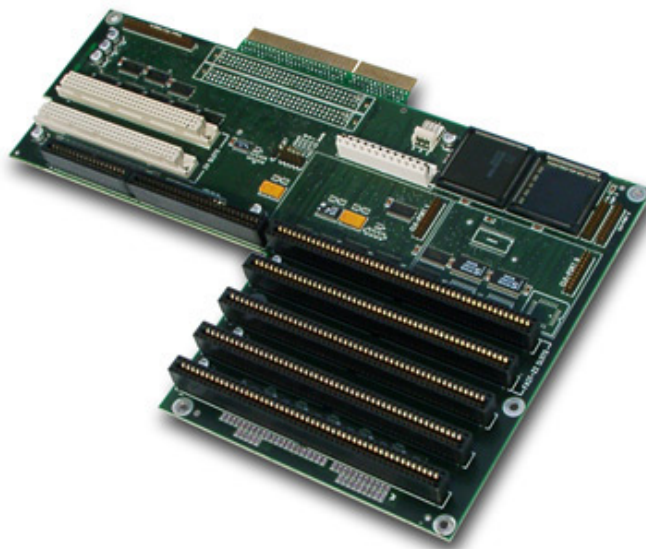
Pierwsze, co rzuca się w oczy, to brak slotów ISA oraz PCI, które były zamontowane w slotach Micronika. Poza tym Zorro IV wygląda jakoś solidniej. Oprócz tego na płycie znajdują się dwa dziwaczne, białe złącza. Są to dwa sloty Zorro IV, ale o tym później.

Podstawą samej karty są oczywiście sloty Zorro II w ilości pięciu sztuk, co nie powinno dziwić – taką samą ilość miał wyrób Micronika. Ciekawą sprawą natomiast jest fakt, że dwa pierwsze sloty są w pełni 32-bitowe. Oznacza to, że działają dwa razy szybciej, ale tylko z dedykowanymi kartami. Pierwszą kartą, która wykorzystywała tę prędkość była karta graficzna produkcji DCE. Wcześniej karty produkowane przez phase5 nie wykorzystują zwiększonej szybkości tych slotów. Kolejną kartą, robiącą użytek ze slotów, jest USB Deneb. Oprócz tego Budda Flash (kontroler IDE) dostaje sporego przyspieszenia. Sloty oczywiście dogadują się również z 16-bitowymi urządzeniami starszej generacji. Zrealizowano to w dosyć prosty sposób – na Zorro IV mamy zworkę, która przełącza dwa pierwsze sloty w tryb rozszerzony.

Drugą ciekawą rzeczą, jaką znajdziemy na slotach Zorro IV Elboku, są cztery porty zegara. Niestety tylko jeden z nich jest aktywny, a szkoda. Konsultowałem ten temat z firmą Elbox, jednak nie byli mi w stanie podać gotowego rozwiązania. Niemniej ponoć da się je wszystkie aktywować w warunkach domowych. Sloty posiadają oczywiście port video, niestety również jest on nieaktywny. Bardziej obeznani użytkownicy Amigi powinni jednak wiedzieć, że aktywny slot video wymaga kilku sygnałów, których na złączu krawędziowym po prostu nie ma. Dlatego niezbędny jest aktywator slotu video. A tutaj jest pies pogrzebany – jest to praktycznie nie do zdobycia. Pomocy należy szukać w firmie Elbox.

Intrygującą sprawą w zasilaniu slotów Zorro IV jest typowe złącze zasilacza AT – mamy P8 i P9. Nie trzeba kombinować z patentem Micronika, o ile nie mamy oryginalnej obudowy Infnitiv Micronika oraz oryginalnego zasilacza. Po prostu podłączamy i działają. Posiadacz kart Apollo cieszą się zapewne, że wyrób Elboku nie ma problemów z tymi kartami – za pomocą zworki można ustawić typ naszej karty turbo i cieszyć się bezstresową pracą Amigi (może nie do końca bezstresową, ale jeśli sloty są sprawne, to nie będą nam sprawiać problemów).

Najciekawszą rzeczą zamontowaną na Zorro IV Elboku są dwa sloty Zorro IV. Jest to po-



© eFUNZIne 2003

myst Krakowiaków i występuje on jedynie na tej karcie. Rozszerzeń, które je wykorzystują, jest dokładnie tyle, ile slotów na karcie: Mediator ZIV oraz Fast-ATA ZIV. To drugie zasadniczo niczym się nie różni od kontrolera Fast-ATA montowanego w miejscu kości ROM. Jest to na pewno zdrowsze rozwiązanie dla naszej Amigi. Nie trzeba rzeźbić podstawkami po układach Amigi – wystarczy wpiąć kontroler do slotu Zorro IV i sprawa załatwiona. Mamy również dużo lepszy dostęp do gniazd kontrolera. Wszystkie te zalety przyćmiewa jedna wada – Fast-ATA ZIV jest dwukrotnie droższa od swojego klasycznego odpowiednika. No cóż, coś za coś.

Punktem kulminacyjnym tego opisu jest jednak Mediator przeznaczony konkretnie pod te sloty. Jest to chyba najbardziej osobliwa wersja Mediatora, jaką wyprodukował Elbox. Do slotu Zorro IV wpinamy specjalny adapter, który łączy się za pośrednictwem dwóch taśm IDE z Mediatorem przykręconym do laminatu Zorro IV. Byłem dosyć zdziwiony, ponieważ cała konstrukcja jest nadzwyczaj stabilna. Mediator ZIV wyposażony jest oczywiście w MMCD oraz jeden mach z nalepką MM. Zwykle Mediatorzy takie nalepki mają dwie. Pozytywnie byłem również zaskoczony całą filozofią kompletu Zorro IV plus Mediator. Na samym początku byłem pewny, że do dyspozycji będą miał jedynie sloty PCI, które zasłaniać będą sloty Zorro, przez co nie będzie możliwe używanie długich kart typowych dla Zorro (mam tu na myśli karty dźwiękowe, kontrolery SCSI, karty I/O itp.). Jednak panowie z Elboku dobrze przemyśleli sprawę – Mediator jest tak skonstruowany, że przysłania jedynie dwa dolne sloty Zorro. Można ich oczywiście używać nadal, ale tylko z krótkimi kartami, takimi jak Tandem IDE czy Deneb. Pozostałe trzy sloty Zorro mieszczą pełnowymiarowe karty.

Drobna uwaga dla użytkowników standardowego kontrolera – Mediator ZIV zasłania całkowicie wbudowany kontroler dysku twardego. Aby go używać, musimy kombinować z prostopadłą przelotką dla dysku 3,5". Przelotka Elsatu nadaje się tu doskonale. Niestety standardowe przelotki dostępne na aukcjach internetowych nie nadają się. Dla mnie jest to dodatkowy bodziec do zakupu kontrolera Fast-ATA dla slotów Zorro IV. Jeszcze jedna rzecz wyróżnia Mediatora ZIV od pozostałych „Medków” – jako jedyny wymaga wpięcia zasilania do gniazda „molex” wbudowanego na płytę Mediatora. Ja o tym zapomniałem i niestety siedziałem cały dzień, zastanawiając się, dlaczego po wpięciu karty telewizyjnej nie widać pozostałych kart, no i dlaczego Sound Blaster 128 nie wydobywa z siebie dźwięku pomimo poprawnej konfiguracji.

Na tym koniec różnic. Mediator ZIV działa dokładnie tak samo szybko i tak samo pewnie jak Mediator LT4 czy 1200. Jednak dzięki temu rozwiązaniu możemy stworzyć sobie Amigę w dosyć ciekawej, choć niestety drogiej konfiguracji.

Benedykt Dziubałowski

Mediator IV oraz sloty Zorro IV są do nabycia w firmie Elbox

<http://www.elbox.com>



FastATA Zorro IV

Jako że jestem szczęśliwym posiadaczem slotów Zorro IV krakowskiej firmy Elbox, zacząłem się zastanawiać nad wykorzystaniem ich mocy. Sloty są dwa, więc na pierwszy ogień poszedł Mediator ZIV stworzony głównie z myślą o nich. Bardzo fajna sprawa. Dzięki temu zyskałem w Amidze 1200 jednoczesny dostęp do slotów PCI oraz Zorro II. Z uwagi na to, że jeszcze jeden slot Zorro IV został wolny, kombinowałem co poza szpilką można tam wcisnąć. Szybka wizyta na stronie producenta uświadomiła mi, że dostępna jest specjalna wersja kontrolera FastATA przeznaczona dla tych slotów. Cena jednak odstrasza. Zacząłem się zastanawiać czy nie prościej kupić klasyczną „atę” wpinaną w ROM Amigi. Po dłuższej refleksji przypomniało mi się, że kiedyś dawno temu podłączałem w ten sposób kontroler. Wyglądał to mniej więcej tak, że na podłożu kładło się płytę Amigi (z wyjętym ROM-em), następnie nakładano się kontroler i kości ROM, po czym stawiało się na tym wszystkim, aby zapewnić maksymalny styk, jaki jest wymagany do prawidłowego działania. Chciałem uniknąć powtórki z historii i mimo większej ceny zamówiłem FastATA ZIV.

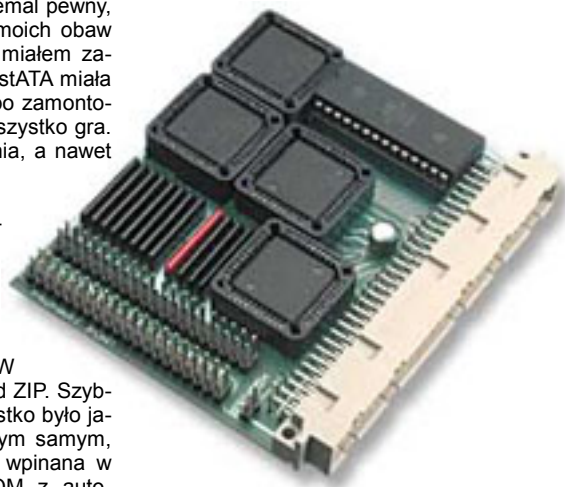
Instalacja

Kilka dni później dotarła do mnie starannie zapakowana paczuszka. W środku było tekturowe pudełko z napisem FastATA ZIV. Czym prędzej oglądnąłem się rozpakowywaniem zawartości i oglądaniem tego cuda. Poza samym kontrolerem w pudełku była faktura, karta gwarancyjna, dyskietka instalacyjna oraz przelotka CF-IDE wraz z przelotką SATA-IDE. Same te dwie rzeczy są warte około 40-50 PLN. Tutaj mam je w gratisie, a raczej w cenie. Zasadniczo to mi się podobało. Nie zamierzałem wykorzystywać tych dwóch gadżetów z moją flagową Amigą 1200, tak więc przelotka CF wylądowała w klasycznej Amidze 1200 desktop, a przelotka SATA-IDE powędrowała do peceta. Ja zająłem się samym kontrolerem. Pierwsze, co zwróciło moją uwagę to fakt, że mimo iż FastATA jest produkowana już ponad 10 lat, to nadal nie ma obudowy na gniazdo IDE. Bardzo łatwo powykrywić piny, a już nie tak łatwo je wyprostować. Panowie z Elboxa – proszę, zwróćcie na to uwagę! Druga rzecz to, według reklamy, najnowsza FastATA posiada obsługę napędów SATA oraz CF. Niestety na samej karcie takich gniazd nie znajdziemy. Wszystko realizowane jest za pośrednictwem dołączonych przelotek. Ktoś się czuje nabity w butelkę? Nie wiem, na czym ma polegać owa „obsługiwalność”, ale chyba już w tych najstarszych modelach można było podłączyć za pośrednictwem odpowiedniej przelotki dysk SATA czy kartę CF do Amigi. Ba, opisywany trick działał także z pominięciem FastATA na standardowym kontrolerze! Trzecią intrygującą mnie rzeczą był kabelek, który należy wpiąć w 37 pin standardowego kontrolera Amigi. Ów kabelek dostarcza sygnał pracy dysku twardego i radośnie oznajmia nam, że coś tam sobie chrupie. Przy bardzo głośnych dyskach, tak zwanych traktorach, jest on zbędny ze względów oczywistych. W cichutkich dyskach warto go podłączyć. W pierwszej chwili byłem pewny, że kabel jest zdecydowanie za krótki. Po

pierwszych przymiarkach byłem niemal pewny, że tak faktycznie jest. Powodem moich obaw był fakt, że w dolnym slocie ZIV miałem zamontowanego Mediatora ZIV, a FastATA miała być wpięta do górnego. Niemniej po zamontowaniu kontrolera okazało się, że wszystko gra. Długość przewodu była odpowiednia, a nawet mały zapas został.

Włączyłem moją Amisę i przystąpiłem do pracy. Najpierw wszedłem do bootmenu i byłem trochę zaskoczony. Do kontrolera były podłączone faktycznie trzy urządzenia: twardy dysk i DVD na pierwszym kanale oraz napęd ZIP na drugim. W bootmenu widoczny był tylko napęd ZIP. Szybki telefon do firmy Elbox i już wszystko było jasne. FastATA ZIV jest właściwie tym samym, co FastATA dla Amigi 3000/4000 wpinana w sloty Zorro III. Karta ma własny ROM z autobootem, czego nie ma FastATA wpinana w ROM Amigi. Takie rozwiązanie sprawia, że karta działa niezależnie od pozostałych kontrolerów i umożliwia bootowanie Amigi z czterech urządzeń. Autoboot daje jednak nie tylko to. Nadal mamy możliwość instalacji zwykłej FastATY, co daje już nam osiem urządzeń (z sześciu możemy bootować komputer). FastATA ZIV nie wyłącza standardowego kontrolera. Ów kontroler nadal działa i można z niego korzystać, o ile nie mamy FastATY wpinanej w ROM-y. Najbardziej w tej sytuacji ciekawiło mnie bootowanie z napędu ZIP. Zanim jeszcze zdążyłem zainstalować sterowniki, to czym prędzej sformatowałem dyskietkę o pojemności 100 MB i zacząłem kopiować na nią partycję systemową mojej Amigi. Co ciekawe, do obsługi napędu ZIP nie są potrzebne żadne dodatkowe sterowniki (chyba że chcemy mieć dyskietkę w formacie FAT do wymiany danych z PC – wówczas wymagana jest odpowiednia mountlista). W trakcie kopiowania uświadomiłem sobie, że nie są nadal wgrane żadne sterowniki do FastATY, a mimo to wszystkie cztery kanały są widoczne w systemie. Tak właśnie działa ROM FastATY. Po przekopiowaniu systemu czym prędzej wcisnąłem reset i obserwowałem co się dzieje. Napęd ZIP w FastATA ma dosyć wysoki priorytet startu – wyższy niż stacja dyskietek DF0. Tak więc nie musiałem wybierać dyskietki z bootmenu. Amiga sama wystartowała z napędu ZIP. Trwało to trochę długo, ale się udało. Dla mnie jest to bardzo przydatna rzecz – mam dyskietkę ratunkową z kompletnym systemem i tym, co jest dla mnie niezbędne. Koniec kombinowania z duetem dyskietka rozruchowa plus napęd CD.

Na zakończenie procesu instalacyjnego zainstalowałem sobie wgranie sterowników, które miały dać kopa całej Amidze. Komputer z radością połknął dyskietkę dołączoną przez producenta, a ja za sprawą kilku kliknięć na blacie Workbenchu uruchomiłem program instalacyjny. Amisia zapytała mnie grzecznie czy może zmodyfikować *startup-sequence* oraz gdzie mam podpięte poszczególne urządzenia (CD-ROM oraz napęd ZIP). Reszta poszła gładko. Ostatnią czynnością było zainstalowanie sterownika do napędów CD – Allegro CDFS. Jest to patent Krakowiaków. Całość zwiększyłem restartem komputera. Czynność poinstalacyj-



Zdjęcie: Elbox
<http://www.elbox.com>

na to uruchomienie programu ustawień FastATY. W nim zdefiniowałem tryby PIO dla poszczególnych urządzeń. Dla twardego dysku i DVD były to tryby PIO5, a dla napędu ZIP PIO0. Ważne jest, aby wolne urządzenia, takie jak ZIP, działały na osobnym kanale. Gdybym napęd ZIP miał na jednej taśmie z dyskiem twardym, to on także musiałby działać w trybie PIO0, co byłoby sporym marnotrawstwem możliwości. Wszystko było już gotowe, tak więc nie pozostało mi nic innego jak wykonać testy szybkości. Ułatwił mi to mały programik drivespeed załączony na dyskietce instalacyjnej. Jego użycie jest bardzo proste – w shellu wpisujemy:

```
drivespeed [device urządzenia] [unit]
```

Dla przykładu wygląda to tak:

```
Drivespeed scsi.device unit 1
```

Nic nie stoi na przeszkodzie, aby sprawdzić szybkość np. SCSI Blizzarda. Wówczas zamiast *scsi.device* wpisujemy odpowiednio *blizzppc.device* lub inny „dewajs”, który jest zgodny z technologią IDE (*trackdisk.device* odpowiedzialny za stację dyskietek nie działa). Oczywiście nie wolno zapomnieć o stosownym unicie. Na wyniki trzeba trochę poczekać (od kilku do kilkudziesięciu sekund – szczególnie długo trwa test napędu ZIP).

Wyniki były bardzo pozytywne. Dysk twardy IDE 40 GB przyspieszył z 2,2 MB/s do 6,5 MB/s. 200% szybciej robi wrażenie! DVD także przyspieszyło znacząco, ale nie było to już tak spektakularne. Najbardziej ucieszył mnie przyrost prędkości napędu ZIP – bez FastATY wyciągał ledwo 200 kB/s, a po aktywacji sterownika prędkość wzrosła do ponad 1 MB/s, co daje wzrost prędkości o 400%. Jednak pozostały mały niedosyt. FastATA reklamowana jest sloganem do 16,6 MB/s. Niestety testy wykazały, że moja Amiga nawet nie zbliżyła się do tego wyniku. Ponoć na kartach Apollo da się wycisnąć nieco ponad 10 MB/s, ale do 16 MB/s to nadal daleko.



FastATA w praniu, czyli testy praktyczne

Mój ulubiony emulator jabłuszka – ShapeShifter – dostał bardzo fajnego kopa. System wczytuje się dużo sprawniej niż bez kontrolera. Same aplikacje oraz gry również działają dużo lepiej. Dodam tylko, że używam plikopartyjki, która sama w sobie jest wolna. Mając teraz tak duży dysk twardy, muszę się pokusić o przetestowanie natywnych partycji „maczka”. Takie gry jak „Quake II”, „Shogo” czy „Freespace” wczytują się szybko. Nie ma już czasu na wypalenie papierosa, gdy standardowy kontroler Amigi w trybie PIO0 doczytywał kolejny poziom gry. MP3 pod AmigaAMP również wczytują się błyskawicznie. Można teraz ustawić większy bufor na plik lub wręcz użyć opcji Preload. Całość działa wystarczająco szybko. MakeCD również pokazuje pazurki – nagrywanie z szybkością x16 nie sprawia problemu. Niestety całość chodzi w trybach PIO a nie DMA, tak więc podczas nagrywania nie powinno się robić nic innego w systemie.

Po zamontowaniu FastATY postanowiłem wymienić też dysk twardy – z 40 GB przesiadłem się na 160 GB. Ten dysk również działa z prędkością 6,5 MB/s. Zanim jednak nastąpiła całkowita wymiana dysków, musiałem przenieść swoje dane z jednego dysku na drugi. Do przekopiowania było niespełna 9 GB danych. FastATA miała to kilka godzin – na standardowym IDE Amigi pewnie z półtorej dnia by to trwało. Ogólne wrażenia są jak najbardziej pozytywne. Dzięki slotom Zorro IV i FastATA Zorro IV oraz Mediatorze ZIV użytkownicy Amigi 1200 mogą się poczuć prawie jak użytkownicy Amigi 4000, choć prawie robi wielką różnicę. Fajną sprawą dla megalomanów jest fakt, że możemy zamontować w Amidze dwa kontrolery FastATA ZIV plus do tego klasyczną FastATA wpinaną w ROM Amigi. Wówczas do dyspozycji mamy 12 urządzeń, z czego bootować możemy z 10. Jeśli komuś jest nadal mało, może sobie sprawić SCSI w Amidze. Jeśli nadal jest mało, to pozostaje już jedynie wykup jakiejś serwerowni i sprzężenie jej z Amigą.

Autoboot FastATA pokazuje pazurki. Do tej pory partycja rozruchowa systemu musiała być w obszarze pierwszych dwóch gigabajtów dysku. Zasadniczo w pełni to wystarczało, ale jeśli ktoś bardzo chciał mieć w paru wersjach AmigaOS 3.x, AmigaOS 4.0, AmigaOS 4.1, NetBSD, to zaczęło się robić ciasno. Klasyczna FastATA ZIV rozwiązuje ten problem. Dzięki własnemu ROM-owi cały dysk twardy dostępny jest od razu. Warunek jest tylko jeden – należy użyć systemu plików obsługującego partycje powyżej 4 GB. Nowsze wersje FFS, a także SFS, PFS radzą sobie z tym bez problemu. Dla lubiących problemy i brzydzących się nowymi systemami plików Elbox przygotował dla FastATY tryb SPLIT. Wówczas cały dysk twardy, niezależnie od systemu plików, dostępny jest zaraz po uruchomieniu komputera. Dotyczy to również klasycznych wersji FastATA. Jest jednak jeden haczyk – dysk pracujący w trybie SPLIT nie będzie widziany przez Amigę bez FastATA.

Trzeba ponarzekać

FastATA ZIV to naprawdę dobre urządzenie, dużo lepsze od tego wpinanego w kości ROM. Sama płytka kontrolera wykonana jest solidnie. Nie powinno to dziwić – Elbox raczej nie odstawia fuszery. Jednak jak każdy wyrób, nie jest bez wad. Aby nie było zbyt różowo, to parę słów krytyki. Po pierwsze, wspomniany brak obudowy na złącza IDE jest według mnie problemem. Po drugie, Allegro CDFS, reklamowane jako najlepszy sterownik CD dla Amigi, nie w pełni obsługuje macintoshowy system plików, co dla mnie jest bardzo ważne i dlatego też wróciłem do swojego starego CDFS-a, który z tym problemów nie ma. Powrót nie był jednak taki łatwy. System firmowany przez Elbox obsługuje UDF (DVD dwuwarstwowe), z czym z kolei zwykłe filesystemy mają problem. Po trzecie, brak możliwości wystartowania Amigi z CD/DVD, pomimo tego, że FastATA umożliwia boot z napędu ZIP. Tutaj powiem, że jest to ogromna szkoda. Wątpliwości także budzi montaż FastATY w slotcie Zorro IV. Nie ma tutaj przebojów z wiecznym dociskaniem tak, jak to ma miejsce w klasycznej FastATA,

ale producent nie przewidział żadnej możliwości przykręcenia kontrolera do slotu Zorro IV. Od razu powiem, że slot Zorro IV, w porównaniu z Zorro II czy Zorro III to zupełnie inne złącza. Tutaj niby się to trzyma, ale widać, że karta pod siłą własnego ciężaru nieco opada. Nie bardzo to widzę, gdy przyjdzie nam Amigę transportować na jakiś zlot.

Kolejny problem to Tryby PIO – w przeciwieństwie do DMA zżerają moc procesora. Jest to jednak błąd konstrukcyjny Amigi 1200, a nie samej FastATA. A1200 ma po prostu pocięte na swojej płycie kanały DMA, a co za tym idzie, nie da się wykorzystać tych trybów pod Amigą. Tego błędu nie mają karty turbo, takie jak GVP, Apollo, Blizzard i to zarówno z procesorem 68k jak i PPC. Tam kontroler SCSI posiada kanał DMA, a co za tym idzie transfer danych odbywa się bez udziału procesora. Efekt jest taki, że możemy spokojnie słuchać MP3 i nagrywać płyty.

Epilog

Kupić czy nie kupić? Jeśli jesteś, drogi czytelniku, posiadaczem slotów Zorro IV, to ja bym się na Twoim miejscu nie zastanawiał. Produkt mimo swoich wad jest udany i warty swojej ceny. Bardzo usprawnia pracę pod AmigaOS 3.x oraz AmigaOS 4.x. Zdecydowanie lepszy wybór niż klasyczna FastATA. Wyrób firmy Elbox nie dorównuje kontrolerowi SCSI zamontowanemu na karcie BlizzardPPC, ale jest jego doskonałą alternatywą. Dzięki FastATA mamy dostęp do dużych dysków twardej (większych pojemnościowo i cichszych niż dyski SCSI), nagrywarek DVD (których nie znajdziemy w technologii SCSI). Poza tym nie każdy ma taką kartę, w dodatku wraz z kontrolerem SCSI. Dodajmy też do tego fakt, że w chwili gdy piszę te słowa, SCSI Blizzarda nie jest obsługiwane przez AmigaOS 4.x. Aby było zabawniej, blizzardowe SCSI jest, w przeciwieństwie do FastATA, obsługiwane przez system MorphOS (kompromisem jest tutaj NetBSD, który obsługuje obie technologie).

Benedykt Dziubałowski

	FastATA	Blizzard SCSI
Obsługa AmigaOS 3.x	TAK	TAK
Obsługa Amiga OS 4.x	TAK	NIE
Obsługa MorphOS	NIE	TAK
Obsługa NETBSD	TAK	TAK
Maksymalny teoretyczny transfer	16,6 MB/s	10 MB/s w A1200
Maksymalny realny transfer	6,5 MB (Blizzard), 10 MB/s (Apollo)	10 MB/s w A1200
DMA	NIE	TAK
Bootowanie z płyty CD	NIE	TAK
Maksymalna ilość urządzeń na kontroler	4	7
Dodatkowe sterowniki	Wymagane do uzyskania pełnego przyspieszenia	Nie są wymagane
Nagrywanie płyt DVD	TAK	Brak nagrywarek w technologii SCSI
Obsługa napędów ZIP plus bootowanie	TAK	TAK
Dostępność	Każda 32-bitowa Amiga	Tylko karty BPPC z zamontowanym kontrolerem
Produkcja	Sprzęt aktualnie produkowany	Nie są produkowane do wielu lat

Kilka rad

- Pamiętaj, aby zwiększyć ilość buforów dla dysków. Standardowo HDToolbox daje 80 buforów. Żadna FastATA nie rozwinie na tym skrzydeł. Optymalne ustawienia to 3000-5000 buforów dla każdej z partycji. Warto więc zaopatrzyć się w większą ilość pamięci, jeśli mamy dużo partycji.
- Napęd ZIP też lubi bufor, ale z racji tego, że działa tylko w trybie PIO0 nie musi ich mieć tak dużo – 500 wystarczy z zapasem.
- MorphOS na klasyka nie lubi FastATA. Działa tylko wtedy, gdy nie korzystamy z drugiego kanału kontrolera. Dodatkowo w *startup-sequence* MorphOS-a trzeba zaznaczyć odpowiedni fix. Sterowniki, a co za tym idzie również i przyspieszenie dysków, nie działają pod systemem MorphOS. AmigaOS 4.x nie ma takich problemów.
- FastATA lubi szybkie dyski działające w trybach ATA 100 i ATA 133. Dlatego nie zapomnij wymienić starej taśmy ATA 66 na nowszą.
- Nie tylko napędy ZIP są obsługiwane przez FastATA, ale także ich konkurencja – zapomniane już dzisiaj, niemal całkowicie, dyskietki LS-120. Ta stacja obsługuje również standardowe dyskietki 3,5" (1,44 MB) z poziomu Amigi oraz własne o pojemności 120 MB.
- Jeśli Amiga ma problem z wystartowaniem, to możemy wciskać klawisze od 1-5. Zmienia to czas oczekiwania na dysk twardy. Jest to rozwiązanie żywym przeniesione z FastATA dla Amigi 3000/4000. Trzeba uważać z cyfrą 2, gdyż jak wiadomo, w Amidze 1200 wyłącza ona kartę Blizzard.



Wywiad z BSzili

autorem m. in. portu UADE 2.13 dla systemów AROS/MorphOS

Na nową wersję *UADE*, użytkownicy MorphOS-a musieli czekać aż siedem lat. Ostatni port tego genialnego odtwarzacza modułów muzycznych dla tego systemu popisał Piru w 2005 roku. W międzyczasie Fredrik Wikstrom przeportował *UADE* w wersji 2.13 dla AmigaOS 4 (tworząc ponadto wtyczkę do TuneNet). Nie trzeba było długo czekać na odpowiedź „niebieskich” i tych z kocim pazurem. Pewien młody Węgier, legitymujący się pseudonimem BSzili (w rzeczywistości to Szilárd Biró), postanowił zmierzyć się z tematem i po kilku tygodniach wyteżonej pracy przeportował nowe *UADE* na MorphOS-a i AROS-a. Pojawienie się nowej wersji programu było doskonałą okazją do nawiązania kontaktu z autorem i wypytania go o to, jakie męki przechodził w trakcie portowania tego zacnego „playerka”.

Na wstępie, dzięki, że zgodziłeś się na ten mini wywiad. A teraz przedstaw się i napisz parę słów o sobie.

Nazywam się Szilárd Biró, mam 23 lata i mieszkam na Węgrzech. Jestem zwykłym, początkującym programistą (śmiech), który nie posiadał Amigi w czasie jej największej świetności. Swoją pierwszą, skromną Amigę 500 kupiłem dopiero w 2005 roku i od razu zwariowałem na jej punkcie. Moje odkrycie „nowożytnej” sceny amigowej miało miejsce raptem kilka lat temu. W 2011 roku zainteresowałem się AROS-em, a w czerwcu 2012 kupiłem Maka Mini i licencję na MorphOS-a.

Czyli w odróżnieniu od większości z nas, jesteś świeżym nabytkiem w amigowym środowisku. Kiedy tak naprawdę zacząłeś programować pod systemy amigowe?

W 2011 roku wystartowałem z portowaniem różnego rodzaju oprogramowania na AROS-a (przede wszystkim gier), a odkąd mam już zarejestrowanego MorphOS-a, staram się portować je także na ten system.

Czemu zdecydowałeś się przeportować UADE 2.13 na MorphOS-a/AROS-a?

UADE – Unix Amiga Delitracker Emulator

Dla mniej zorientowanych w temacie, *UADE* to odtwarzacz amigowych modułów muzycznych pracujący w trybie tekstowym. Jest to tak naprawdę nieco „wykastrowany” emulator UAE, przystosowany tylko do odtwarzania muzyki. Z tego też powodu *UADE* jest dosyć procesorożerny, jednak jakość (a raczej wierność) odtwarzanych plików muzycznych jest wyjątkowo wysoka. Warto dodać, że *UADE* obsługuje ponad 150 różnych formatów muzycznych, co daje mu już na starcie przewagę nad jakimkolwiek innym odtwarzaczem modułów muzycznych na post-amigowych systemach.

<http://zakalwe.fi/uade/>

Bo na żadnym z tych systemów nie było aktualnej wersji *UADE*. Szczerze mówiąc pod AROS-em, nie było jej wcale.

Długo pracowałeś nad portem dla obu systemów?

Portowanie samego rdzenia aplikacji nie trwało długo – bazowałem na porcie *UADE 2.13* z AmigaOS 4, autorstwa Fredrika Wikstroma. Schody zaczęły się, gdy rozpocząłem pracę nad linią komend odtwarzacza. Wszystko zajęło mi mniej więcej dwa tygodnie – zacząłem z końcem sierpnia 2012.

Jakie są główne zmiany pomiędzy nową wersją UADE a poprzednią (1.03), przeportowaną przez Piru?

Przeczytaj listę zmian! ;) Generalnie, *UADE 2* oferuje znacznie większą kompatybilność w temacie odtwarzania modułów muzycznych, ich wierności odtwarzania, jakości. Zawiera też mniej błędów.

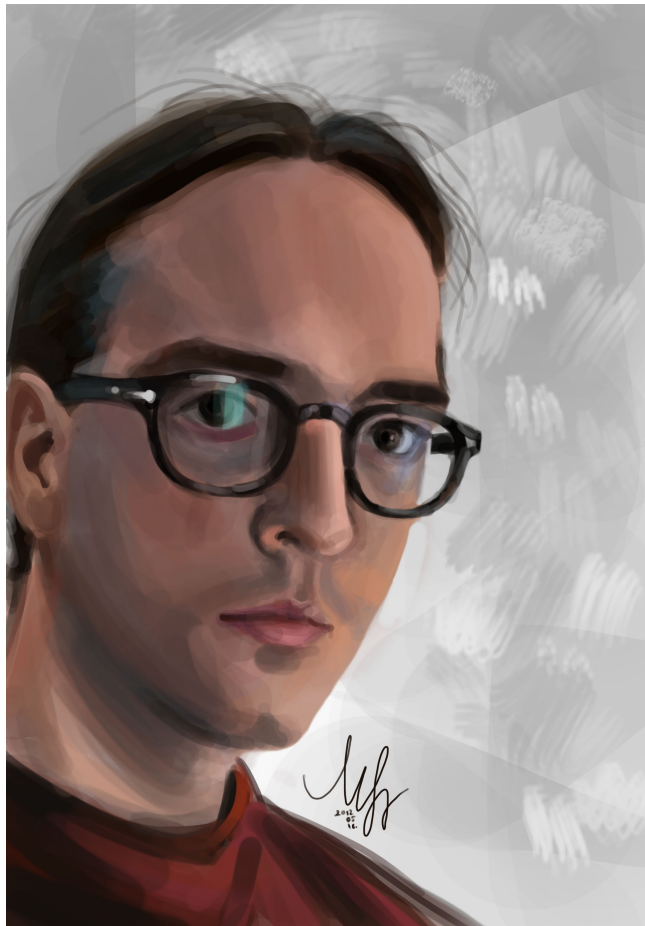
Co było największym wyzwaniem w trakcie portowania nowego UADE?

Dzięki temu, że korzystałem z portu *UADE* pochodzącego z AmigaOS 4, problem z uniksowymi ścieżkami miałem z głowy. Tak czy owak, zostało mnóstwo pracy, głównie nad tym, aby program działał poprawnie na obu wspieranych przeze mnie systemach (AROS/MorphOS). Na potrzeby wersji AROS-owej napisałem alternatywną warstwę komunikacyjną, opartą o message-porty. Z kolei wersja MorphOS-owa musiała zostać wzbogacona o obsługę AHL (bo nie mogłem znaleźć działającej biblioteki libao). Linia komend odtwarzacza też wymagała dodatkowej pracy (obsługa wejścia/wyjścia, amigowych wzorców nazw plików itp.).

Dodałeś jakieś nowe funkcjonalności do swojego portu UADE?

Niespecjalnie, linia komend *UADE* (uade123) musi być zgodny z wersją uniksową.

A czy jest jakaś funkcjonalność, z której jesteś wyjątkowo zadowolony?



Komunikacja międzyprocesowa (którą napisałem dla AROS-a) jest moim faworytem. Co prawda, jest to komunikacja synchroniczna, a przez to nieco wolniejsza niż w przy użyciu strumieni. Jednak dla zwykłego użytkownika różnica jest niezauważalna, a to już coś.

Z drugiej strony co Cię najbardziej wkurza w UADE?

To oczywiście moja osobista opinia, ale naprawdę nie lubię mechanizmu komunikacji międzyprocesowej w oryginalnym kodzie *UADE*. Z technicznego punktu widzenia jest całkiem ciekawa, ja jednak wolę bardziej konwencjonalne rozwiązania (np. wewnętrzności *UADE* w formie amigowej biblioteki).

Jakich modków słuchałeś w trakcie testowania swojego portu?

Do testów używałem załączonego do źródeł *UADE* modułu *AHX.Cruisin*, który zresztą bardzo mi się podoba. Miałem „przyjemność” odsłuchiwać go w całości setki razy (przy każdorazowym odpaleniu *UADE*), co trochę spowolniło proces testowania. Poza tym uwielbiam moduły tytułowe z „*Turricana II*” i „*Turricana III*”. One też przydawały się w trakcie testów.



Czy zostało coś jeszcze do zrobienia (jakieś błędy, niedokończone „ficzery”)?

Linia komend odtwarzacza pod MorphOS-em wymaga jeszcze dopracowania. Czasami UADE się zawiesza albo wywołuje wyjątek procesora tuż przed odtworzeniem modułu. Myślę, że namieszkałem coś w trakcie implementacji podwójnego buforowania dźwięku w AHL. Poza tym pracuję jeszcze nad wtyczką UADE dla AmiNetRadio.

A właśnie, jakie masz plany odnośnie tej wtyczki?

Na początek chcę napisać prosty plugin, który odpalałby UADE z domyślnymi ustawieniami. Kiedy to będzie gotowe, zacznę dodawać pozostałe funkcjonalności znane z UADE (filtry, normalizacja itp.).

Zaczęłeś już pracę nad wtyczką?

Praca nad wtyczką jest jeszcze w bardzo wczesnej fazie rozwoju. Jednak już na dzień dzisiejszy mogę zdradzić, że obsługa UADE z jej poziomu jest prawie skończona.

Co może być największym wyzwaniem przy

pracy nad wtyczką?

Jej współpraca z samym UADE może być trochę skomplikowana. Źródła istniejących wtyczek są bardzo pomocne, ale nie ma tam za dużo kodu, który mógłbym tak po prostu przekleić do swojego pluginu. Napisanie odtwarzacza, który w pełni poprawnie współpracuje z API AmiNetRadio wymaga naprawdę bardzo starannego przygotowania.

Zmieniając trochę temat, napisz parę słów o społeczności amigowej na Węgrzech?

Za dużo na ten temat nie mogę powiedzieć, bo nie jestem zbyt aktywnym jej uczestnikiem. Jednak z tego, co widzę, amigowanie na Węgrzech kręci się głównie wokół Amigi klasycznej, chociaż jest też wielu użytkowników MorphOS-a oraz AmigaOS 4. My też mamy amigowe portale (<http://www.amiga.hu> i <http://www.amigaspirit.hu>). Warty uwagi jest też portal stricte poświęcony systemowi MorphOS – <http://www.pegasos.hu>. Jeśli chodzi o amigowe spotkania, to portal Amiga Spirit ma swój klub w Budapeszcie. Słyszałem też o

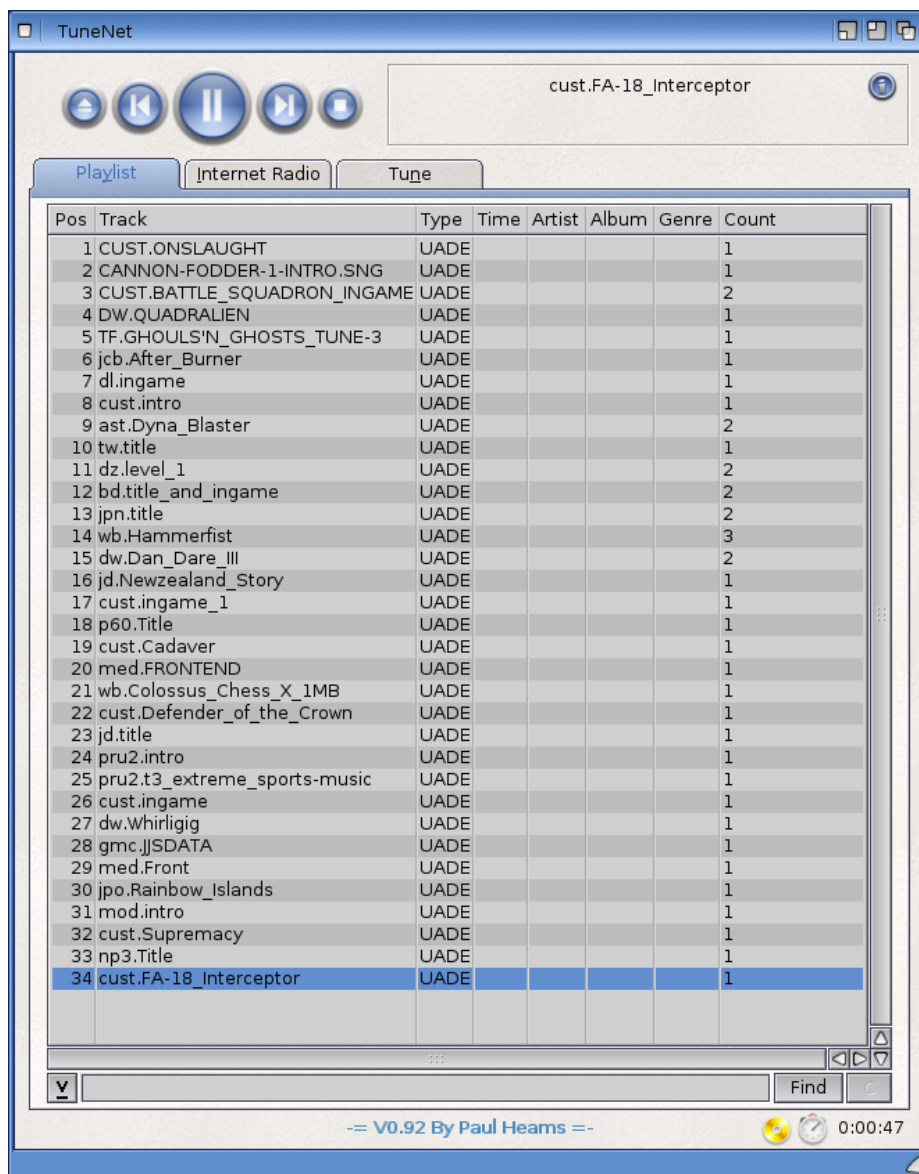
uade

kilku mniejszych meetingach, w różnych rejonach kraju.

Na koniec, czy chciałbyś powiedzieć coś użytkownikom Amigi w Polsce?

Amigowa społeczność w Polsce jest ogromna, miażdźcie! ;)

**Wywiad przeprowadził MarX
fuego@o2.pl
konsultacja merytoryczna –
Krashan**



TuneNet w wtyczką UADE

Co nowego w UADE2 (w porównaniu do wersji 1.03)?

- wierniejsza emulacja oryginalnych filtrów audio z Amigi 500 i 1200 zaowocowała jeszcze lepszą jakością generowanych dźwięków (domyślnie ustawiony jest filtr z A500)
- interfejs tekstowy odtwarzacza reaguje na polecenia wydawane w trakcie odtwarzania modułów. Z jego poziomu możemy m. in. przechodzić między subsongami, odtwarzać kolejny moduł, wyświetlać informacje o module, nanosić dodatkowe filtry i efekty dźwiękowe w czasie rzeczywistym. Pełna lista opcji dostępna po uruchomieniu *uade123*.
- dodano specjalną opcję dla miłośników słuchawek (UADE generuje delikatny przesłuch między kanałami). Aby ją uruchomić w trybie „interaktywnym”, należy najpierw uaktywnić postprocessing (klawisz **P**), a potem odpalić efekt słuchawkowy (klawisz **H**).
- pamięć przeznaczona na emulowaną Amigę może być teraz zwiększona do 8 MB (konfiguracja w pliku *uaerc*). Domyślnie pole *chipmem_size* zawiera wartość 4, (czyli 2 MB, bo pamięć obliczana jest ze wzoru: *liczba x 512 kB*). Zatem aby zwiększyć pamięć do 8 MB, należy *chipmem_size* ustawić na 16 (bo $16 \times 512 \text{ kB} = 8 \text{ MB}$).
- możemy wreszcie znormalizować głośność odtwarzanych modułów (klawisz **N**). Yupii!
- usunięto wiele poważnych wycieków pamięci (choć port UADE2 dla systemu MorphOS nie należy do wyjątkowo stabilnych).
- dodano obsługę nowych formatów modułów muzycznych m. in. Sierra AGI, Synthdream, Special FX ST, Quartet ST/PSG, KRIS, Jochen Hippsel ST, Dirk Bialluch.

Sołtys

– *Hej, Józek!* – głos niskiego, dziobatego Ferdka wpatrującego się w okno gospody za-rezonował takim echem w pustym kuflu, że siedzący przy nim listonosz skrzywił się niemilosiernie.

– *Ciszej żesz Ferdek, bo pożałujesz, zarazo jedna* – warknął i dla podkreślenia wagi swoich słów walnął wóchatym kulakiem w blat. Od razu tego pożałował, bo różowe i błękitne plamy pod powiekami zaczęły znów swój obłędny taniec, a czaszka o mało mu nie eksplodowała.

– *Józek, nie uwierzysz, Sołtys ku nam zbliża się!*

– *A bo to dziwota? Baba go z domu wygnali, szukać ma tego gacha, co sołtysównie sprzed ołtarza uciekł. Leon mu było, czy ki czort? Łazi tak biedak po wsi w tę i we wte od rana, śmieci jakieś zbiera... – ani chybi oczadziało chłopca! I jeszcze bredzi bez ustanku, że teraz to już na pewno się uda, bo może liczyć na pomoc jakiegoś Kuma... Czy Skumma?*



W ten oto sposób stali bywalcy wiejskiej klubokawiami w Poraży wprowadzili nas w zawiłości fabuły rodzimej (wydanej przez znane również z amigowego poletka Laboratorium Komputerowe AVALON) gry przygodowej pod tytułem „Sołtys”. Pojawiła się ona u schyłku świetności Amigi na polskim rynku (w roku 1995) i być może dlatego nikt nie pomyślał o tym, żeby przygotować jej wersję na ulubione przez nas komputery. Ale teraz, dzięki bohaterskim twórcom maszyny wirtualnej ScummVM (stałe poszerzającym obsługiwaną przezeń listę tytułów) i Ty, drogi czytelniku, możesz wcielić się w postać tytułowego sołtysa i spróbować przywrócić ład i porządek we wsi (a przede wszystkim święty spokój we własnym domu).

Instalacja, jak to w przypadku gier obsługiwanym przez ScummVM bywa, sprowadza się do przekopiowania zainstalowanej na PC gry na naszą Amigę (bądź „Amigę”) i wskazania programowi ścieżki dostępu. W przypadku

„Sołtysa” jesteśmy w o tyle lepszej sytuacji, że pełną wersję gry (zarówno po polsku, jak i po angielsku) można pobrać bezpośrednio ze strony projektu (adres w ramce).

Gdy żmudny etap „instalacji” mamy już za sobą, nadchodzi niezapomniana chwila odpalenia gry. I oto naszym spragnionym oczom ukazuje się... (po obowiązkowym logo wydawcy) różowiućka facjata naszego bohatera, która (jeśli pominiemy lub obejrzymy do końca listę plac) mruga do nas porozumiewawczo. Po chwili jednak przestaje być tak wesoło, gdyż: po pierwsze – nasz bohater zostaje zamasytym kopniakiem wyrzucony z domu, a po drugie – przychodzi nam po raz pierwszy zmierzyć się z interfejsem gry (o tym później) oraz grafiką, jaką przyjdzie nam oglądać, a jest ona... Hmm... Nierówna. Elementy dość przyzwoite graniczą z prawdziwymi graficznymi potworkami i dziwnymi patentami w stylu „a tu puści się blura na bezkształtną plamę i jakoś

to będzie”. Animacja trochę przypomina tę z „Kajka i Kokosza” – sołtys chodzi, jakby polknął kij od szczotki, a po przedmioty schyla się z dynamiką trenera Tai-Chi. Poza tym – im dłużej grałem w „Sołtysa”, tym bardziej miałem wrażenie, że zastosowana w grze paleta kolorów jest zbyt przesycona. Efekt jest taki, że robi się cukierkowo, niczym paczka landrynek od pani Jadzi z osiedlowego spożywczaka. W każdym razie wygląda to jakby grafik zachłystnął się po raz pierwszy w życiu paletą VGA i nie oddaje za bardzo tego przasnego, wiejskiego (wieśniackiego?) klimatu, w którym utrzymana jest rozgrywka. Szczęściem programiści pomyśleli o graczach z wrażliwymi gałkami ocznymi i zaimplementowali specjalny przycisk przełączający rozgrywkę w tryb odcielenia szarości.

Skoro już przy przydatnych przyciskach jesteśmy, to kolejny z nich służy do wyłączenia muzyki. Zasadniczo to zdanie mogłoby wystarczyć za opis jej jakości, ale ten artykuł musi mieć swoją długość, więc wyłuszczę rzecz szczegółowo: utworów jest sporo (to plus), każda lokacja (a jest ich w grze 24) posiada swoją unikalną muzyczkę. Po pierwsze jednak – napisane zostały w standardzie MIDI, przez co ranią czule uszy amigowca, a po drugie – wykonano je niedbale. Ledwo kilka z nich da się słuchać bez wspomaganie ciężkimi prochami, więc albo muzykę wyłączamy, albo konsultujemy się z lekarzem lub farmaceutą. Z dźwiękami jest nieco lepiej (z góry uprzedzam, dialogów mówionych nie ma - trzeba się wczytywać w „dymki”), choć można zwątpić słysząc na przykład „pisk myszy”. I tak nie da się ich wyłączyć - chyba że wyłączymy głośniki – więc jeśli zechcecie w „Sołtysa” zagrać, to jakoś trzeba przywyknąć.

Wspomnieliśmy o bardzo przydatnych przyciskach. Jest ich w grze trzy, a znajdują się pośrodku belki. Ten, którego do tej pory nie opisałem (czyli trzeci od góry) służy do opuszczenia gry, ale... wiem, wiem że kusi, ale jeszcze go nie wciskajmy. Po lewej stronie mamy





zagłębienie z numerkami. Początkowo jest tylko jeden, ale w miarę naszych postępów będzie ich przybywać. Służą one do przenoszenia się między poszczególnymi lokacjami (nie da się przejść do sąsiedniej po prostu podchodząc do krawędzi ekranu). Funkcja dość nietrudna, ale w sumie oszczędzająca czas. Aby nie trzeba było zapamiętywać, który numerek co oznacza, ta lokacja, nad którą zatrzymaliśmy wskaźnik myszy, pokazana jest dodatkowo w formie miniatury na szarym ekranie pośrodku belki. Po prawej zaś mamy banki przedmiotów. Ponieważ jest ich tylko osiem (a gra jest tak skonstruowana, że banków nie ma prawa nam zabraknąć), próbowanie „wszystkiego na wszystkim” będzie nieco łatwiejsze. Jeśli mówimy już o przedmiotach, to warto przybliżyć sposób sterowania. Włodarem wsi kierujemy za pomocą myszki. Lewy przycisk służy do poruszania się po lokacji (choć nasz sołtys jest nie dość, że pełen jakichś fobii, to jeszcze uparty jak osioł i czasem stwierdza kategorycznie „A właśnie, że tam nie pójdę”; niekiedy udaje się tę tendencję przezwyciężyć, niekiedy nie) oraz do wybierania przedmiotu z naszego „plecaka” – zaświeci się wówczas przy nim zielona lampka. Prawy przycisk natomiast służy do zbierania przedmiotów, jak i do interakcji z elementami scenarii i okolicznymi mieszkańcami. Z całym sposobem sterowania wiąże się jeszcze jedna niedogodność – żeby użyć danego przedmiotu w określonym miejscu, trzeba tam najpierw podejść. Inaczej dowiemy się, że „z tej odległości nic się nie da zrobić” – szalenie irytujące dla kogoś, kto przywykł, że bohater gry jest choćby średnio rozgarnięty i sam potrafi kawalek podejść.

„Oj tam, przecież to wszystko nie ma takiego znaczenia, najważniejszy i tak jest klimat” – powie ktoś i będzie miał sporo racji. Tyle tylko, że w tej materii „Sołtys” też prezentuje się przeciętnie. Całkiem niezły pomysł (czyli osadzenie gry w realiach polskiej wsi, i to wsi specyficznej, o której krąży/krążyło chyba tyle samo dowcipów, co o Chucku Norrisie) rozmienniono trochę na drobne wykonaniem. Przede wszystkim, jeśli ktoś zna kawały o Wąchocku/Poraży, to większość humorystycznych smaczków przyprawi go, co najwyżej, o napad ziewania (a pozostałe często są... hmm... niewybredne). Z drugiej strony – jeśli owych kawałów nie zna, to przejście gry może się okazać dlań nie lada wyzwaniem. Bądź tu, czło-



wieku, mądry i pisz wiersze! Kolejna sprawa to fakt, że opisywana gra, przedstawiając obraz polskiej wsi, kreśli go operując oklepanymi stereotypami – wirtualnej Poraży praktycznie wszyscy to albo świry, albo piniacze, albo alkoholicy (a najczęściej wszystko na raz). Poraża wszechobecne nieróbstwo i bylejakość, a jedynymi rozrywkami (poza wchłanianiem etanolu) wydają się filpery w knajpie i zabawa w „strip-dupniaka” z bufetową. Pomysł, który jakoś sprawdził się w przypadku „Franko” tu gdzie się rozmywa. Może to dlatego, że we „Franko” walczyliśmy z napotkanym stanem rzeczy, a tu – staramy się z wszystkich sił zachować status quo? Być może.



Wypadałoby jeszcze jakoś podsumować naszą recenzję, napiszę więc tak: W grę da się grać, da się nawet ją ukończyć. Przyjemność z tego niewielka, ale jeśli kręcą Was „swojskie” klimaty i dowcipy z brodą, to z pewnością znajdziecie tu coś dla siebie.



Na koniec dorzucę jeszcze małe ostrzeżenie: Jeśli będziecie zapisywać stan gry, to zostawcie pierwszą pozycję wolną. Gra bez uprzedzenia zapisuje autosave na tym właśnie miejscu.

Konrad Czuba

Od 26 listopada 2011 roku gra posiada status freeware i można ją legalnie pobrać:

<http://prdownloads.sourceforge.net/scummvm/soltys-pl-v1.0.zip?download>



Dla tych, co nie lubią kolorów – „Sołtys” w wersji „black and white”

- ciekawy pomysł

- słaba oprawa
- niedopracowany interfejs
- niewybredny humor



Nasza pierwsza gra

Kurs programowania z wykorzystaniem Visual Studio 2005 Express Edition i VBCC – część 4

Czwarty odcinek zaczniemy od sprzątnia. A jest co robić – w jednym folderze są upchnięte wszystkie pliki, a że jest ich już całkiem sporo, lepiej będzie to wszystko poukładać. W folderze *boxo* tworzymy trzy katalogi: *src*, *obj* i *exe*. W pierwszym z nich będzie cały kod źródłowy, w drugim obiekty pośrednie utworzone podczas kompilacji, a w ostatnim pliki: wykonywalny, grafika do gry i plansze. Do katalogu *src* przenosimy pliki z rozszerzeniem *.c* i *.h*. W folderze *obj* umieszczamy te kończące się na *.o*. W *exe* znajdują się *boxo*, *tile.pic* i *levels.bin*. W głównym folderze *boxo* zostaną nam *makefile.mak* i pliki związane z Visual Studio. Nie wystarczy tylko przenieść pliki, trzeba także zmienić *makefile* – łatwo się o tym przekonać próbując skompilować projekt. Naiwnym rozwiązaniem byłoby powpisywanie na sztywno ścieżki do plików w *makefile.mak*, lecz my ułatwimy sobie pracę i stworzymy dwie nowe zmienne: **SRCDIR = src** i **OBJDIR = obj**. Umieścimy je zaraz za definicją zmiennej **CFLAGS**. **OBJ** także trzeba zaktualizować. I tu jest dylemat, gdyż w jednej linii trzeba umieścić sporo rzeczy. Na szczęście jest możliwość łamania linii za pomocą znaku ****. Należy pamiętać, aby ten znak kończył linię. W każdym razie zmienna ta teraz wygląda tak:

```
OBJ = $(OBJDIR)\boxo.o \
      $(OBJDIR)\window.o \
      $(OBJDIR)\timer.o \
      $(OBJDIR)\input.o \
      $(OBJDIR)\level.o \
      $(OBJDIR)\tile.o \
      $(OBJDIR)\game.o \
      $(OBJDIR)\fileIO.o
```

Zauważmy, że ostatniego wpisu nie kończymy znakiem ****, gdyż właśnie tej linii już nie będziemy łamać. Warto na to zwrócić uwagę, gdy będziemy dodawać nowy wpis, aby przypadkiem nie zapomnieć dodać tego znaku. Aktualizujemy zmienną **LINKOBJ** na tej samej zasadzie co **OBJ**. Modyfikujemy także **BIN = exe\boxo**. Ostatnią rzeczą, jaka musi ulec zmianie, to reguły tworzenia obiektów pośrednich (pliki zakończone *.o*). Dodajemy na początek *nazwa.o* **\$(OBJDIR)/**, a wszędzie tam gdzie jest *nazwa.c*, dajemy na początku **\$(SRCDIR)/**. Aby wszystko było jasne, zamieszczam trzy przykładowe reguły, które już są zmienione.

```
$(OBJDIR)/boxo.o: $(SRCDIR)/boxo.c
$(CC) $(CFLAGS) $(SRCDIR)/boxo.c -o $(OBJDIR)/boxo.o
$(OBJDIR)/window.o: $(SRCDIR)/window.c
$(CC) $(CFLAGS) $(SRCDIR)/window.c -o $(OBJDIR)/window.o
$(OBJDIR)/timer.o: $(SRCDIR)/timer.c
$(CC) $(CFLAGS) $(SRCDIR)/timer.c -o $(OBJDIR)/timer.o
```

W tym miejscu warto uruchomić nasz projekt i skompilować. W przypadku błędu należy go przeanalizować i ustalić, o czym zapomnieliśmy. Dla tych, którzy już nie będą mieli sił, zamieszczam gotowe archiwum (odnośnik w ramce). Sprzątnięcie mamy za sobą, teraz zajmujemy się przeniesieniem plansz z kodu źródłowego do oddzielnego pliku. Można zapytać, co daje taki manewr poza zmniejszeniem pliku wykonywalnego? Niewątpliwie korzyścią jest to, że nie będziemy musieli po każdej zmianie planszy, kompilować naszego projektu. Przy tak małym projekcie to może mieć małe znaczenie, gdyż czas kompilacji jest krótki i okresy te są porównywalne. Przy dużych projek-

tach nie jest tak różowo. Do wad przeniesienia plansz do pliku można zaliczyć nieoczywisty sposób aktualizacji planszy, nie wspominając o przypadku, gdy chcemy dodać nową planszę. O ile w przypadku drobnej zmiany możemy posilkować się jakimś hex edytorem, to rozszerzanie bądź zwięźnianie ilości plansz wymaga większej wiedzy (znajomość formatu planszy). Te problemy rozwiązuje się przez napisanie edytora do gry bądź zmieniając format planszy na plik tekstowy. W każdym razie w obydwu przypadkach czeka nas praca. Wracając do eksportu plansz na zewnątrz, to przede wszystkim musimy zacząć od napisania procedur operujących na plikach. Spostzegawczy czytelnik z pewnością zauważył, że podczas porządków z *makefile*, pojawił się plik pośredni o nazwie *fileIO*. Właśnie w nim umieścimy procedury, które będą nam pomocne przy rzeczach związanych z plikami. Dodajmy regułę na koniec *makefile.mak*.

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

Należy nanieść też odpowiednie zmiany do zmiennych **OBJ** i **LINKOBJ**, które to już zrobiliśmy przy porządkach z wyżej wymienionym plikiem. Na sam początek potrzebujemy przynajmniej dwóch procedur: jednej do odczytu, a drugiej do zapisu. Zaczniemy od odczytu. Zastanówmy się przez chwilę, co musimy wiedzieć, aby załadować plik pod jakimś nam znany adres. Po cichu zakładam, że mamy dość miejsca na tenże plik. Na pewno musimy znać nazwę pliku i jego rozmiar. Razem z adresem, gdzie mają być umieszczone dane, wychodzą trzy parametry. Aby funkcja była elastyczna i możliwe było wczytanie nie tylko plansz, a także dowolnego pliku, to umieścimy te trzy rzeczy jako parametry naszej funkcji. Oto i ona:

```
int ReadFile(char* name, UBYTE* buffer, int size)
{
    FILE* f = fopen(name, "rb");
    if (NULL == f)
    {
        return RT_FAILED_OPEN_FILE;
    }
    fread(buffer, 1, size, f);
    fclose(f);
    return RT_OK;
}
```

Pierwszym argumentem jest wskaźnik na nazwę, drugi to wskaźnik na miejsce w pamięci, a trzeci to rozmiar. W ciele funkcji użyłem procedur, które są w standardowej bibliotece *io*. Można w tym miejscu użyć odpowiednich funkcji bibliotecznych z *dos.library*, ale myślę, że tak jest o wiele łatwiej i przejrzystej, co nie znaczy, że nie ma błędów. Jeśli przyjrzymy się bliżej, to zauważymy, że sporo rzeczy tutaj może powodować bardzo wiele zła – łącznie z zawieszeniem naszego programu. Warto zdawać sobie z tego sprawę. Nie sprawdzamy czy wskaźniki są poprawne, czyli czy są różne od **NULL**. Tak naprawdę to nie wiemy czy ktoś zaalokował wystarczającą ilość pamięci pod buffer oraz co się stanie, gdy ktoś poda niewłaściwy rozmiar pliku. Cały ten ciężar przeniosłem na zewnątrz, czyli do innych procedur bądź bloków

kodu. To ich zadaniem będzie troska o poprawność argumentów tejże funkcji. W przypadku, gdy otwarcie pliku nie powiedzie się, dostajemy bardzo lakoniczną informację, że nie udało się wczytać pliku, co też może być uznane za wadę, a może też być furtką do dalszej analizy błędu. Jako ćwiczenie proponuję dodać obsługę przypadku, gdy funkcja *read* nie przeczyta żądanej ilości informacji. Proszę pamiętać, że plik jest otwarty i wypada go zamknąć. Bardzo podobnie do *ReadFile* będzie wyglądała funkcja *WriteFile*, która to zapisuje plik i nie ma większego sensu rozpisywać się na ten temat. Oprócz odczytu i zapisu bardzo przydatną procedurą będzie pobranie rozmiaru pliku. To także można rozwiązać przy pomocy *dos.library*, ale aby być konsekwentnym, używamy biblioteki *io*, którą to dołączamy za pomocą **#include <stdio.h>** (to tak dla przypomnienia i na marginesie).

```
int GetFileSize(FILE* f)
{
    int size;
    int current = ftell(f);

    fseek(f, 0, SEEK_END);
    size = ftell(f);
    fseek(f, current,
SEEK_SET);

    return size;
}
```

Pobieranie rozmiaru pliku sprowadza się do zręcznego manipulowania funkcjami *fseek*. Na początku zapamiętujemy obecną pozycję w pliku, potem przechodzimy na koniec pliku i pytamy się za pomocą *ftell*, jak daleko jest koniec, czyli jaki jest rozmiar i na koniec przywracamy pozycję w pliku taką, jaka była na początku. Dodałem także drugą wersję pobierania pliku, gdzie argument jest nazwą pliku – różni się ona od poprzedniej tym, że otwieramy i zamykamy plik. Zanim zabierzemy się za przenosiny plansz, zmodyfikujemy naszą planszę tak, aby nasz kosmiczny statek mógł się pojawiać w dowolnym miejscu naszej planszy. Bo jak przyjrzymy się funkcji *NextLevel* w pliku *game.c*, to nasz pojazd zawsze pojawia się prawie w górnym lewym rogu niezależnie od planszy. Aby to zmienić, trzeba naszego bohatera umieścić na planszy i zaktualizować jego początkową pozycję *x* i *y*. Wartością odpowiadającą na planszy naszemu statkowi jest *TILE_SHIP* i właśnie tę wartość umieszczamy w wybranym przez nas miejscu na planszy. Warto zaznaczyć, że ten element był tylko raz na danej planszy – w przeciwnym razie gracz może być delikatnie dezorientowany. Umieszczamy statek we wszystkich planszach, by uniknąć kolejnej dziwnej sytuacji, gdy na ekranie nie ma naszego pojazdu (chyba że kosmiczni krętarce maczali tam palce). Modyfikacji ulegnie funkcja *NextLevel* i tam umieszczamy w następnej linii, zaraz za procedurą *LvlToWin*, wywołanie nowej funkcji *findShipPositionOnLevel*. Przebiegamy całą planszę w poszukiwaniu elementu *TILE_SHIP* i gdy go już odnajdziemy, to ustawiamy pozycję początkową statku. W miejsce pojazdu wstawiamy pusty klocek, po czym bezzwłocznie opuszczamy funkcję. Łatwo zauważyć, że nie ma tu żadnego sprawdzania czy rzeczywiście jest statek na planszy, co jest na pewno minusem. Poza tym nie ma też obsługi przypadku, gdy ktoś przez pomyłkę wstawi dwa



będź więcej elementów typu `TILE_SHIP`.

Mamy już procedury umożliwiające nam podstawowe operacje na plikach, to teraz zabierzmy się na poważnie za przeniesienie plansz z kodu źródłowego do pliku `levels.bin`. Użyjemy sprytniej sztuczki, aby wygenerować plik z planszami. Dodajmy kod tuż przed instrukcją `return RT_OK`; do funkcji `InitLevel` w pliku `level.c`:

```
writeFile("levels.bin", g_tabLevels, sizeof(g_tabLevels)/sizeof(g_tabLevels[0]));
return RT_OK
```

Pozostaje nam skompilować i uruchomić nasz projekt, wtedy to zostaną zapisane plansze do pliku `levels.bin`. Jeśli komuś się nie udało, to nie należy się przejmować – archiwum będzie już miało ten plik. Wykasujemy linię, gdzie zapisywaliśmy nasze plansze, bo swoje zadanie już spełniła. Teraz dodamy kod, który alokuje pamięć na plansze i załadujemy je tam. Wszystko to będzie odbywało się w `InitLevel`. Usunięcia wymaga tablica `g_tabLevels` – w jej miejsce deklarujemy zmienną `g_pAllLevels`, która jest wskaźnikiem na `UBYTE`.

Nas najbardziej interesuje fragment zaczynający się od pobrania rozmiaru pliku `levels.bin`. Po otrzymaniu rozmiaru planszy, alokujemy pamięć funkcją `malloc`, sprawdzamy czy udało nam się pozyskać ją i po czym wczytujemy nasze plansze dokładnie w to zaalokowane miejsce. Oczywiście skoro przydzieliliśmy sobie pamięć, to musimy ją także oddać, co też czynimy w `KillLevel`, w standardowy sposób używając do tego celu funkcji `free`.

W ostatniej części tego odcinka zajmiemy się `gameport.device`, czyli dodamy obsługę joysticka do naszej gry. Pewne doświadczenie z device już mamy (`timer.device`), więc nie powinno być większych problemów. Podobnie jak w `timer.device` tworzymy `MsgPort`, który będzie potrzebny IO requestowi do komunikacji z device. Zobaczmy początkowy blok kodu z funkcji `InitInput` w pliku `input.c`.

```
int InitInput(void)
{
    m_pMsgPort = CreateMsgPort();
    if (NULL == m_pMsgPort)
    {
        return RT_FAILED_GAMEPORT_MSGPORT;
    }

    m_pIO = CreateIORequest(m_pMsgPort, sizeof(struct IOStdReq));
    if (NULL == m_pIO)
    {
        return RT_FAILED_GAMEPORT_IOREQ;
    }

    m_error = openDevice("gameport.device", 1, (struct IORequest*)m_pIO, 0);
    if (0 != m_error)
    {
        return RT_FAILED_GAMEPORT_DEVICE;
    }

    g_nGamePortSignal = 1L << m_pMsgPort->mp_SigBit;
}
```

W skrócie – tworzymy `MsgPort`, tworzymy request `IOStdReq`, otwieramy device i zapamiętujemy sygnał z `MsgPortu`. Przy otwieraniu device ustawiamy unit na 1, co oznacza, że będziemy korzystali z drugiego portu – przeważnie tam mamy podłączony joystick. Przy `timer.device` to było w zasadzie wszystko, jeśli chodzi o inicjalizację, a przy `gameport` musimy wykonać jeszcze parę rzeczy. Przede wszystkim sprawdzimy czy ktoś przypadkiem nie używa już urządzenia i jeśli tak jest, to wysyłamy informację, że my będziemy go używać. Ponieważ pracujemy w multitaskingu, to na czas odpytywania i ustawiania zatrzymamy go tak, aby ktoś inny nie podebrał nam joysticka. Teraz nadszedł czas na drugą część kodu z funkcji `InitInput`:

```
Forbid();
const BYTE type = getControllerType();
if (GPCT_NOCONTROLLER == type)
{
    setControllerType(GPCT_ABSJOYSTICK);
}

Permit();
setTriggerConditions();
clearBuffer();
sendGamePortRequest();

return RT_OK;
```

Najpierw zatrzymujemy multitasking za pomocą funkcji `Forbid` z `exec.library`. Pobieramy typ kontrolera.

Jeśli nikt go nie używa, to ustawiamy joystick (`GPCT_ABSJOYSTICK`) i przywracamy multitasking funkcją `Permit`. W następnym kroku ustawiamy warunki otrzymywania informacji z urządzenia, dalej czyszcimy urządzenie i na końcu wysyłamy request z prośbą o odczytanie stanu joysticka. Przyjrzyjmy się bliżej wywoływanym funkcjom.

Funkcja `getControllerType` wysyła prośbę (`DoIO`) za pomocą komendy `GPD_ASKCTYPE` o typie kontrolera. Dane o nim zostaną zapisane do bajtu `result`, stąd rozmiar jest ustawiony na 1 (jeden bajt) i `io_Data` zawiera adres zmiennej `result`. Sama funkcja `DoIO` wykona się w tym przypadku szybko, bo w zasadzie nie korzystamy z urządzenia, a pytamy, jak się przedstawia stan rzeczy.

```
static BYTE getControllerType(void)
{
    BYTE result;

    m_pIO->io_Command = GPD_ASKCTYPE;
    m_pIO->io_Flags = IOF_QUICK;
    m_pIO->io_Data = (APTR)&result;
    m_pIO->io_Length = 1;
    DoIO((struct IORequest*)m_pIO);

    return result;
}
```

Bardzo podobna do niej jest funkcja `setControllerType`, która ustawia typ kontrolera na `GPCT_ABSJOYSTICK`. Oznacza to, że będziemy dostawać pojedynczą informację o stanie joysticka (wychylenie bądź puszczenie w danym kierunku) i przyciskach (naciśnięcie bądź puszczenie). Na przykład, gdy gracz wykona sekwencję ruchów: lewo a potem prawo i wtedy naciśnie przycisk, to urządzenie wyśle nam informację: wychylenie lewo, puszczenie lewo, wychylenie prawo, przycisk naciśnięty.

Funkcja `setTriggerConditions` odpowiada za wybór informacji/warunków, jakie będziemy otrzymywać. Innymi słowy – w tej procedurze precyzujemy, jakie i kiedy dostaniemy wiadomości o stanie joysticka i przycisków. Do tego celu używa się struktury `GamePortTrigger`. Pole `gpt_Keys` ustawiamy na `GPTF_DOWNKEYS`, co oznacza, że interesują nas wiadomości o wciśnięciu przycisku. `gpt_XDelta` i `gpt_YDelta` ustawiamy na 1, gdy chcemy otrzymywać informację lewo/prawo i góra/dół. Można oczywiście ustawić `gpt_YDelta` na zero lub też wybrać wartość różną od 1, a wtedy nie dostaniemy wiadomości o stanie góra/dół. Do `gpt_Timeout` możemy wstawić odstęp czasowy. Jednostką jest 1/50 sekundy dla PAL i 1/60 dla NTSC. Jeśli pracujemy w trybie PAL, to ustawienie 25 spowoduje, że co pół sekundy

będziemy otrzymywali wiadomości o joysticku i przykładowo, gdy gracz wychylił joystick w lewo przez 4 sekundy, to otrzymamy o tym 8 wiadomości (dla jasności – będzie to 8 wiadomości o tym samym stanie joysticka). My nie potrzebujemy takiej informacji i dlatego to pole jest ustawione na zero. Dalej w `InitLevel` wywołujemy funkcję `clearBuffer`, której zadaniem jest wyczyszczenie urządzenia, bo może się zdarzyć, że ktoś przed nami korzystał z joysticka i jakieś informacje pozostały w buforze. A na samym końcu, za pomocą `sendGamePortRequest`, wysyłamy naszą prośbę o odczytanie urządzenia, a wynik będzie zapisany w zmiennej `m_event`, która jest strukturą `InputEvent`.

Inicjalizację `gameport` mamy za sobą – czas na funkcję przetwarzającą zdarzenia z joysticka. W standardowy sposób, za pomocą sygnału, który jest przechowywany w zmiennej `g_nGamePortSignal`, będziemy odbierać wiadomości z urządzenia. Oczywiście, aby było to możliwe, dodajemy nasz sygnał do maski sygnałów w funkcji `Wait` (tak na marginesie – na kilka sygnałów to już czekamy), która jest w metodzie `loop` w pliku `boxo.c`. Po odebraniu sygnału z `gameport.device` wywołujemy funkcję przetwarzającą zdarzenia z joysticka. I w podobny sposób do funkcji `signalWindow` z pliku `window.c`, odbieramy komunikaty za pomocą `GetMsg`. Wtedy to po odebraniu komunikatu, jesteśmy pewni, że `m_event` zawiera aktualne informacje o stanie joysticka. Pole `ie_Code` zawiera informację o przyciskach. `IECODE_LBUTTON` oznacza przycisk w joysticku, a `IECODE_RBUTTON` to drugi przycisk, który notabene jest traktowany w grach po macoszemu. My robimy z niego użytek i za pomocą niego możliwe jest natychmiastowe opuszczenie gry. Pola `ie_X` i `ie_Y`, które tak na marginesie są wygodnym skrótem do `ie_position.ie_xy.ie_x` i do `ie_position.ie_xy.ie_y`, zawierają informacje o kierunkach joysticka. W przypadku `ie_X` mówimy o wychyleniu w lewo bądź w prawo (wartość 1 oznacza prawo a -1 – lewo). Oczywiście 0 oznacza, że dźwizek nie był wychylony w tychże dwóch kierunkach. Dla `ie_Y` wartość 1 to wychylenie w dół a -1 to górne wychylenie. Po analizie zdarzenia i ustawieniu odpowiednich zmiennych wysyłamy naszą prośbę o ponowne odczytanie joysticka.

Została nam jeszcze jedna funkcja – `KillInput`. Jak łatwo się domyślić jej zadaniem jest zakończenie pracy z `gameport.device`. Najpierw przerywamy poprzednio wystartowaną prośbę, informujemy `gameport.device`, że urządzenie jest wolne, ustawiając odpowiedni typ `GPCT_NOCONTROLLER`. A potem to już zamknięcie device i skasowanie naszego IORequesta i `MsgPorta`.

To wszystko w tym odcinku, zachęcam do eksperymentowania z projektem; zmieniania grafiki, plansz, kodu. W przypadku jakichkolwiek pytań proszę zadawać je na forum PPA.

Asman

Cała paczka z kodem do pobrania z poniżej zamieszczonego odnośnika.

<http://www.ppa.pl/artykuly/download/ftp/boxo.7z>

Sprzęty trOLLO – część 1

W tej serii artykułów chciałbym bliżej zaprezentować trochę rzadkich amigowych sprzętów, jak również kilka krótszych artykułów poświęconych amigowemu sprzętowi. Artykuły nie są typowymi recenzjami – skupiają się raczej na szybkim opisanie danego urządzenia i jak się ono sprawuje w działaniu z Amigą. Mam nadzieję, że kącik przypadnie Wam do gustu i z ciekawością będziecie wyczekiwać kolejnych jego części. Zaczynamy.

Amiga 600 – Fast RAM inaczej

Uwielbiam A600. Jest mała, ładna, działa na niej większość gier z A500, można w prosty sposób rozbudować ją o dysk twardy, dodatkową pamięć CHIP. Brakuje jej tylko łatwej możliwości rozbudowy pamięci FAST. Rozwiązania są dwa: kupno karty turbo do A600 (jest to jednak dosyć spory wydatek), kupno karty SRAM na PCMCIA (tutaj problemem jest dostęp do tych kart). Pierwsze rozwiązanie jest dosyć typowe, więc wybrałem to drugie – kartę SRAM na PCMCIA. Kartę 1 MB SRAM firmy SUNMAX kupiłem w nieocenionym sklepie eFuzine. Wprawdzie mówi się, że taka pamięć może być wolniejsza od „prawdziwej” pamięci FAST, ale procesor MC68000 demona prędkości też nie jest i taka pamięć równie dobrze spełnia swoje zadanie.



Montaż nie jest skomplikowany. Wkładamy kartę SRAM do gniazda PCMCIA i uruchamiamy naszą Amigę. Przy pomocy programu PREPCARD konfigurujemy kartę jako pamięć FAST. Potem reset A600 i mamy już pamięć FAST w naszej Amidzie – w moim przypadku AmigaOS 2.1 pokazuje 2 MB pamięci CHIP i 1 MB pamięci Fast.

Po uruchomieniu A600 działa zdecydowanie szybciej. Praca na mojej konfiguracji jest znacznie przyjemniejsza, odczuwalnie szybciej otwierają się okienka, system szybciej się wczytuje. Sysinfo pokazuje 0.73 MIPS – prędkość naszej A600 wzrosła o 32%. Wzrosła także prędkość odczytu z dysku z ok. 500 kb/s (bez FAST) do ponad 800 kb/s (z FAST). Czas jednak zrobić test na tym, co najważniejsze – gry i programy. DPaint IV szybciej się wczytuje, same operacje na grafice (wypełnianie, rysowanie) też działają trochę szybciej. Niektóre bardziej wymagające gry działają płynniej lub szybciej się wczytują, na przykład „Civilization” (szybsze wczytywanie, szybszy przesuw po planszy), „F/A-18 Interceptor” (płynniejsza animacja podczas gry), „Wings” (szybsze wczytywanie misji, płynniejsza animacja podczas części 3D), „UFO: Enemy Unknown” (krótszy czas oczekiwania między turami, płynniejsza animacja globu Ziemi). Uruchomiłem też demo grupy Alcatraz – „Odyssey”, któ-



Amiga 600 i SRAM na PCMCIA

re wcześniej nie chciało mi działać właśnie ze względu na brak pamięci FAST – prawie 40 minut oglądania dema!

Zalety są chyba oczywiste. Za dosyć małe pieniądze mamy pamięć FAST (jak dobrze poszukamy można znaleźć nawet karty 2 MB). Wada jest w sumie jedna – mamy zablokowany port PCMCIA. Dlatego też, aby przegrać dane z PC na A600, muszę wyjąć kartę SRAM, włożyć przelotkę PCMCIA-CF, włączyć A600, przegrać dane, wyłączyć, włożyć kartę SRAM. Obawiam się o wytrzymałość portu PCMCIA, ale na szczęście częściej żonglerki nie uprawiam.

Obecnej moja A600 posiada ROM 2.05, 1 MB Chip montowane „pod klapką”, dysk twardy 750 MB, zewnętrzną stację dyskietek oraz wspomnianą kartę 1 MB SRAM. Zabawa w takiej konfiguracji to już czysta przyjemność – system uruchamia się szybciej, dużo gier działa zdecydowanie płynniej, można pooglądać dema scenowe, które mają większe wymagania. Zdecydowanie karta warta zakupu.

Dysk SSD, czyli XXI wiek w Amidzie

Amiga to niesamowity komputer. Moja A1200 mimo 22 lat na karku bez problemu przyjmuje każdą nowinkę technologiczną, jaką jej przyniosę. Tym razem dostała nowiutki dysk SSD i stała się moim pierwszym komputerem wyposażonym jedynie w dysk SSD – wszystkie nowinki najpierw do Amigi a potem do PC. Szukając nowego dysku nie chciałem wydawać za dużo pieniędzy, ponieważ wiedziałem, iż

A1200 nie wykorzysta wszystkich zalet nowego dysku SSD. Tak więc najważniejsza była cena – im tańszy, tym lepiej. Po przejrzaniu ofert kilku większych i mniejszych sklepów, wybór padł na dysk MACH Xtreme MX-Starter 32 GB SATA II. Dysk dostarczony jest w gustownym pudełku. W środku znajduje się urządzenie i prosta karteczka informacyjna. Jako że Amiga nie posiada złącza SATA, to do podłączenia dysku użyłem adaptera SATA-IDE firmy 4World, które już było przeze mnie wcześniej testowane i sprawia najmniej problemów przy współpracy z Amigą.

Po podpięciu dysku i uruchomieniu komputera, HDToolbox (dołączony do AmigaOS 3.9) bez najmniejszych problemów odczytał dane dysku, po chwili miałem podzielony dysk na kilka partycji. Tak samo jak w przypadku karty CF trzeba zmienić ustawienia *maxtransfer* na – **0x0001FE00**, ponieważ bez zmiany tego parametru mogą pojawiać się dziwne błędy odczytu/zapisu. Po ustawieniu wszystkiego instalacja AmigaOS 3.9 przebiegła bez najmniejszych problemów. Po instalacji systemu i przegraniu reszty danych ze starego dysku

	Operacje na sekundę
Create	91
Open	131
DirScan	622
Delete	192

Wyniki uzyskane programem SysSpeed 2.6



	MB/s
Create File	4,21
Write File	3,99
Read File	5,99
Raw Read	6,12

Wyniki uzyskane programem SysSpeed 2.6

przyszedł czas na małe testy. Na początek informacja o konfiguracji Amigi, na której je przeprowadziłem: A1200, 68040 40 MHz, 64 MB pamięci, Fast-ATA MK II, karta graficzna GVP Spectrum 2 MB Zorro II, AmigaOS 3.9. Ten sam test przeprowadzony programem Sysinfo przy teście partycji SSD0 wykazał, że transfer wynosi ok 7.5 MB/s. Trochę mnie zdziwiły te wyniki, ponieważ na tradycyjnym dysku talerzowym odczyt wynosił w granicach 10 MB/s (Sysinfo i SysSpeed pokazywały mniej więcej to samo, czyli ok. 10 MB/s). Trudno powiedzieć w jakim stopniu te wyniki są prawdziwe, ponieważ wyraźnie widać przyspieszenie pracy komputera. System startuje zdecydowanie szybciej niż przy starym dysku, przeglądanie katalogów jest także szybsze, wiele programów też startuje szybciej – np. „Doom II”. Po kilku godzinach zabawy da się odczuć dziwne przyspieszenie, system jakby szybciej reaguje na wszelkie operacje dyskowe. Być może amigowe programy testujące nie dogadują się z dyskami SSD, dlatego otrzymałem takie dziwne wyniki?

Czas na małe podsumowanie. Dysk SSD Mach Extreme to chyba najwolniejszy dysk SSD, jaki widziałem obecnie w sprzedaży. W teorii jego prędkość to ok. 100 MB/s i to jest jego wada. Jednak w przypadku Amigi nie ma to znaczenia, gdyż nie ma kontrolera, który wykorzystalby pełną prędkość tego dysku. Zaletami są jego cena (można je znaleźć nawet poniżej 150 zł), ogólne przyspieszenie działania systemu Amigi oraz... cisza – po włączeniu Amigi z dyskiem SSD najgłośniejszym elementem jest wiatrak w zasilaczu. Dlatego też polecam zakup dysku SSD – dla ciszy, lansu i szpanu...

Monitor LCD do Amigi

Ciężko jest znaleźć monitor LCD, który będzie bez najmniejszych problemów działał z Amigą. Czasami monitor działa, a czasami niestety nic nie wyświetli. Na bazie moich doświadczeń stworzyłem małe zestawienie kilku monitorów, które udało mi się przetestować, a sam test był bardzo prosty – Amigę podpiąłem do monitora za pomocą gniazda D-Sub, wyświetlając sygnał PAL i DbIPAL i DbINTSC. Niektóre monitory posiadają wbudowany tuner TV, więc w takim wypadku podpiąłem też Amigę przez przelotkę RGB-Euro sprawdzając, jaki obraz pokaże monitor. Wszystko, co udało mi się w ten sposób „zbadac” zebrałem w tabelkę. Testy były przeprowadzone na Amidze 600 2 MB Chip, 1 MB Fast, AmigaOS 2.1 (test obrazu na tunerze TV) oraz Amidze 1200 4 MB Fast, AmigaOS 3.0 (test trybów DbIPAL i DbINTSC).

DbINTSC, **DbIPAL** oraz **PAL** oznacza, że monitor poprawnie wyświetlił obraz w zadanym trybie ze złącza RGB Amigi podpiętej do gniazda D-Sub monitora. **TV** oznacza poprawne wyświetlenie obrazu PAL na tunerze TV monitora – w tym wypadku Amiga została podłączona przez przelotkę RGB-Euro. Informacja



Dysk SSD i adapter 4World

TAK lub **NIE** chyba nie wymaga wyjaśnienia. Oznaczenie **SŁABO** oznacza, że monitor wyświetlał obraz, ale po chwili on gasł i pojawiał się komunikat o nieobsługiwanej sygnale. Monitory wyróżnione pochylem i wytłuszczeniem sprawdziłem na A1200 z kartą graficzną EGS Spectrum 2 MB w rozdzielczości 1024x768 – na tych monitorach obraz wyświetlił się bez problemu. Dodatkowo w przypadku wyświetlania trybów DbIPAL i DbINTSC w katalogu *DEVS/Monitors* był także sterownik *VGAOnly*.

Jak widać, z monitorami jest mały problem, gdyż większość z przetestowanych nie obsługuje poprawnie częstotliwości wyświetlanych

przez chipset ECS czy AGA. Dodatkowo pojawia się problem w przypadku jednego modelu monitora, ale pochodzącego z różnych partii: *Samsung C23A550U* z partii kwietniowej wyświetlał poprawnie obraz z trybów DbIPAL i DbINTSC, natomiast ten samo monitor, ale z partii czerwcowej już nie był w stanie wyświetlić obrazu. Dlatego też, jeżeli w tabelce jest opis **TAK/NIE**, oznacza to iż monitory z różnych partii potrafiły obraz wyświetlić lub nie. Mam nadzieję, że opis ten pomoże przy wyborze monitora.

Aleksander „trOLLO” Giedyk

Model monitora	DbINTSC	DbIPAL	PAL	TV
Acer G225HQVBD	NIE	NIE	NIE	-
LG E2251VR-BN	TAK	TAK	NIE	-
LG DM2350D-PZ	TAK	TAK	NIE	TAK
LG E2381VR-BN	SŁABO	NIE	NIE	-
LG M2450D-PZ	NIE	NIE	NIE	TAK
LG M2550D-PZ	TAK	SŁABO	NIE	TAK
LG DM2780D-PZ	NIE	NIE	NIE	TAK
PHILIPS 221TE4LB	TAK	NIE	NIE	TAK
PHILIPS 226V3LSB	TAK	TAK	NIE	-
SAMSUNG S19A100N	NIE	NIE	NIE	-
SAMSUNG LS22B150NS	TAK	SŁABO	NIE	-
SAMSUNG T22A300	NIE	TAK	NIE	-
SAMSUNG T22A350	SŁABO	NIE	NIE	TAK
SAMSUNG T24A350	SŁABO	NIE	NIE	TAK
SAMSUNG T24B301	NIE	NIE	NIE/TAK	TAK
SAMSUNG C23A550U	TAK/NIE	TAK/NIE	NIE	-

EGS Spectrum 24/28

Układy graficzne w Amigach nie należą do najwydajniejszych. Owszem, w prostych grach spełniają swoje zadanie, ale gdy mamy ochotę wykorzystywać naszą Amigę do czegoś więcej niż tylko gry, to pojawia się problem. Nagle okazuje się, że nawet najprostsze zadanie jest gigantycznym wyzwaniem dla naszego komputera mimo posiadania szybkiego procesora i dużej ilości pamięci RAM. Lekarstwem na powolne układy graficzne naszych Amig jest karta graficzna.

Amiga 1200, którą posiadam, jest wyposażona w sloty Zorro II firmy Micronik, więc wybór był prosty – karta, która działa w slotach Zorro II i nie jest droga. Warunek miałem w zasadzie jeden – karta ma bez problemu obsłużyć system minimum w rozdzielczości 800x600 przy 16 bitowej palecie kolorów.

Wybór padł na kartę EGS Spectrum 24/28. Karta oparta jest na układzie Cirrus Logic GD5426 taktowanym częstotliwością 40 MHz, wyposażona jest w 2 MB pamięci graficznej i działa pod systemami Picasso 96 i CGX od wersji 3. Sama instalacja karty nie sprawia najmniejszych problemów – po prostu wkładamy ją w wolne gniazdo Zorro II i już. Po kilku dniach testów, stwierdziłem, że karta pracuje trochę szybciej pod Picasso 96 niż pod CGX v4, dlatego też zainstalowałem pakiet Picasso 96. Instalacja pakietu Picasso 96 przeszła bez najmniejszych problemów. Podczas instalacji system sam rozpoznał zainstalowaną kartę, szybko wybrałem częstotliwość, jakie obsługuje mój monitor i po chwili mogłem cieszyć się Workbenchem w nowej rozdzielczości. Jako że posiadam kartę, która wyposażona jest w 2 MB pamięci graficznej, to mogę ustawić rozdzielczość systemu nawet na 1280x1024. Niestety w takiej rozdzielczości praca na tej karcie nie należy do najprzyjemniejszych – interlace i ogólna powolność karty dają o sobie znać. W niższych rozdzielczościach (np. 800x600 16 bit) praca w systemie to czysta przyjemność. AmigaOS 3.9 działa zdecydowanie szybciej niż na wolnych układach AGA.

Jako że Amiga to nie tylko system, ale także programy i gry, to po skończonej konfiguracji

systemu zabrałem się za testy. Pod lupę poszły programy i gry, których najczęściej używam na Amidze.

Większość programów czy gier napisanych „pod system” działa bez najmniejszych problemów pod kontrolą karty graficznej. Niektóre programy są jednak bardzo kapryśne, np. „Doom I” i „Doom II” czasami zawieszają cały komputer (nie można ustawić palety 16-bitowej). Szkoda też, że mój ulubiony program do przeglądania zdjęć FastView w ogóle nie lubi karty graficznej i nic nie wyświetla (podobnie rzecz ma się z Visage). Czasami można zaobserwować wolniejsze działanie niektórych programów w stosunku do prędkości pracy na chipsecie AGA, np. w Personal Paint 7.1 większość operacji zrobimy bardzo szybko, ale np. wypełnianie kolorem obiektu trwa dziwnie długo. Ponadto możemy go otworzyć wyłącznie na ekranie maksymalnie 8-bitowym. Zdziwiłem się także po uruchomieniu gry „Napalm” w 320x240 na karcie graficznej przewijanie planszy jest „skokowe”, natomiast w 320x256 AGA przewijanie planszy jest płynne. Trudno powiedzieć czy to wina karty, sterowników czy też gry, ponieważ „Exodus” działa zdecydowanie szybciej na karcie graficznej niż pod kontrolą układów AGA. Dużym zaskoczeniem (pozytywnym) była gra „Quake I” – nadal działa wolno, nawet w 320x240, jednak zdecydowanie szybciej niż na AGA. Teraz rozgrywka nie jest aż tak bardzo stresująca. Z powodzeniem na karcie graficznej uruchomiły się też programy AWeb, FileMaster 3.1, Wordworth 7 (w tooltypach należy ustawić PICASSO=TRUE), SysSpeed 2.6 oraz gry „Heretic”, „Gloom Deluxe”, „Genetic Species”, „Myst”, „Payback”.

Dużym i przyjemnym zaskoczeniem była prędkość odtwarzania filmików AVI/MOV z bardzo starych płyt CD (dla ciekawych – CD z nieistniejącego pisma Kawaii) – w systemie pracującym pod kontrolą AGA animacje nie były wyświetlane płynnie, obraz i dźwięk strasznie przycinały. Teraz można spokojnie odtwarzać te animacje, nawet w oknie na pulpicie systemu bez większego obciążenia procesora (średnie obciążenie 35-40%).

Niektóre gry lub programy można zmusić do działania na karcie graficznej – do tego celu używam programu NewMode. Dzięki niemu mogę czasami uruchomić gry, które nie mają wyboru rozdzielczości i teoretycznie nie można ich ustawić, aby wyświetlały obraz na karcie graficznej. Dobrym przykładem jest „UFO: Enemy Unknown” – po zastosowaniu programu NewMode, wybrałem rozdzielczość 320x240 i mogłem normalnie uruchomić tę grę. Wspomniały wcześniej FastView (a także



Moovid pod AmigaOS 3.9 na recenzowanej karcie graficznej

Visage oraz SysInfo) jest niestety odporny na wszelkie zabiegi i wyświetla tylko czarny ekran.

Mimo iż EGS Spectrum 24/28 to jedna z wolniejszych kart dostępnych dla Amigi posiadającej sloty Zorro, to praca na komputerze wyposażonym w tę kartę to czysta przyjemność. System w 800x600 16 bit wygląda bardzo ładnie, działa szybko i przyjemnie. Większość programów i gier działa zdecydowanie szybciej niż na konfiguracji z AGA. Polecam każdemu posiadaczowi Amigi kupno karty graficznej.

Wszystkie testy były przeprowadzone na mojej Amidze w konfiguracji: Amiga 1200 E/Box, 68040 40 MHz, 64 MB Ram, FAST-ATA MK II, GVP Spectrum24/28 2 MB, dysk 32 GB SSD SATA II, DVD-RW LG SATA II

Aleksander „trOLLO” Giedyk

Rozdzielczość	Głębina kolorów (w bitach)	Uwagi
1280x1024	8	Wyświetlanie z przeplotem
1152x900	8	Wyświetlanie z przeplotem
1024x768	8	
800x600	8, 16	
640x480	8, 16, 24	

Rozdzielczości możliwe do uzyskania w EGS Spectrum 24

- szybsza praca systemu
- wyższa rozdzielczość systemu
- większa ilość kolorów w systemie
- współpraca z monitorami LCD

- niektóre gry działają wolniej na karcie w stosunku do AGA
- część programów nie działa pod kontrolą P96/CGX



Karta sieciowa D-LINK DUB E100 USB w Amidze klasycznej

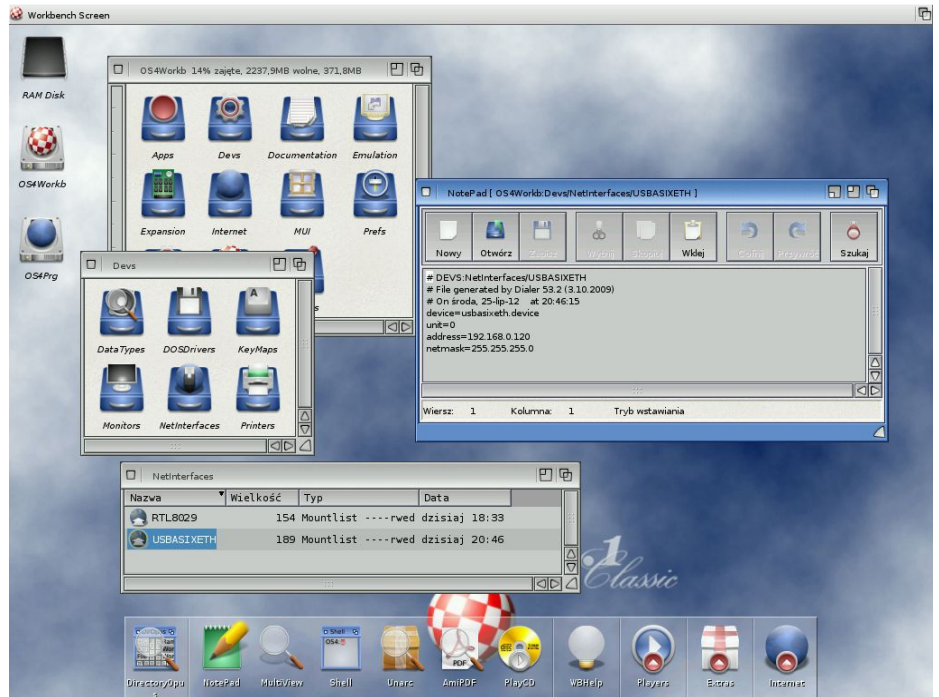
Jako posiadacz Amigi 4000 z Mediatorem i kartą CSPPC604e 233/60/50 MHz SCSI i 128 MB pamięci bardzo ucieszyłem się z możliwości korzystania z AmigaOS 4.1 na swoim sprzęcie. Niestety jedna rzecz nie dawała mi spokoju – firma Hyperion Entertainment uraczyła nas nowym AmigaOS na klasyka w wersji 4.0, a później 4.1, lecz, niestety, przy tej okazji dołączyła jedynie sterowniki do kart sieciowych Ethernet 10 Mbps (Elbox stworzył sterownik do kart FastEthernet 100 Mbps, ale niestety tylko dla AmigaOS 3.9). Komfort korzystania z sieci (ze szczególnym uwzględnieniem internetu) pozostawiał wiele do życzenia. Przykładowo – próby pobrania większych plików kończyły się u mnie z reguły niepowodzeniem (połączenie lubilo się zerwać). Postanowiłem więc poszukać sieciówki 100 Mbps działającej pod wyżej wymienionym systemem. A więc „Google” w dłoń i zaczynamy! Generalnie wszystkie fora internetowe (oczywiście łącznie z PPA) poinformowały mnie o działaniu karty D-Link DUB-E100 USB pod AmigaOS 4.1 na klasyku. Temat wydał mi się interesujący, więc postanowiłem się nim bliżej zainteresować.

Sprzęt

Potrzebna będzie Amiga wyposażona w kontroler USB. Jako szczęśliwy posiadacz Amigi 4000 z kontrolerem USB DENEb działającym pod AmigaOS 4.1 ten problem akurat miałem z głowy. Tutaj małe dygresja. Gdy kilka lat temu zakupiłem kontroler, jako jedną z pierwszych rzeczy, którą zrobiłem, było umieszczenie w pamięci flash karty wszystkich plików koniecznych do działania stosu USB Poseidon. Nie muszę więc mieć żadnych pików Poseidona na dysku twardym, a sprzęt działa od razu po włożeniu. Dlatego po instalacji systemu AmigaOS 4.1 na Amidze, USB zadziałało mi „od pierwszego kopa”. Po podjęciu decyzji o zakupie sieciówki przejrzałem, co oferuje znany aukcyjny portal internetowy oraz internetowe sklepy. Ceny w granicach 120–140 zł nie wyglądały zachęcająco, ale udało się znaleźć sprzedawcę, który zadowolili się kwotą 60 zł.

Instalacja

Po podpięciu sieciówki od razu wyskoczył komunikat o podłączeniu urządzenia D-Link DUB-E100. Po podpięciu kabla od internetu



AmigaOS 4.1 w działaniu

obie zielone diodki zaczęły się świecić. Teraz wystarczy uruchomić kreator połączeń internetowych (`SYS:/Internet/NewConnection`). Sygnał podaje Amidze za pośrednictwem routera dzielącego sygnał z modemu od kablówki (25 Mbps). Opcja automatycznej konfiguracji z DHCP spowodowała bezproblemowe odpalenie nowego połączenia (jedynie musiałem wybrać kartę, której chcę używać: RTL8029 PCI czy też Axis USB Ethernet Adapter), ale po sieci zaczęły się kłopoty. Ustawienie statycznego adresu IP załatwiło sprawę. Postanowiłem zatem zrobić to samo pod AmigaOS 3.9 zainstalowanym na tej samej Amidze. I tu zaczęły się pierwsze poważne schody. Każdy chyba wie, jakie automatyczne konfigurowanie i DHCP posiada stos Genesis. Wziąłem się więc za konfigurację ręczną i od razu porażka.

Jaki sterownik należy podać dla urządzenia Sana-II? Wróciłem więc do AmigaOS 4.1 i wyedytowałem plik z `SYS:/DEVs/Netinterfaces/`. Plik nazywa się `USBASIXETH` i takż jest sterownik: `usbasixeth.device`. Teraz pod AmigaOS 3.9 poszło już z górki. Jako urządzenie Sana-II wpisałem „z palca” `usbasixeth.device` (bez podawania jakiegokolwiek ścieżki dostępu) i już! Drugim sposobem jest wyedytowanie starego połączenia sieciowego pod działającą z dwoma systemami sieciówką RTL8029 (na której wcześniej pracowałem): `SYS:/Internet/Genesis/db/Genesis.conf` i zmiana sterownika na ten dla DUB-E100.

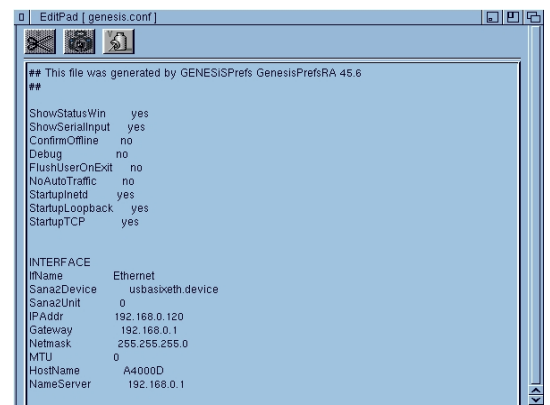
Podsumowanie

Uważam, że operacja była warta tych wydanych 60 zł. Komfort korzystania z internetu wzrósł niewspómiernie, praktycznie też zakończyły się zwiisy przy pobieraniu większych plików. No i przede wszystkim urządzenie działa na obu używanych przeze mnie amigowych systemach.

P.S. Moje odczucia z poprawy prędkości są czysto subiektywne, ale pod poniższym odnośnikiem niejaki Darren Eveland przeprowadził odpowiednie testy:

<http://blog.hyperion-entertainment.biz/?cat=518>

Paweł „Kitech” Kita





Poradnik dla początkującego posiadacza Amigi – karty procesorowe dla A1200

Amiga przez wiele lat była kojarzona z Amigą A500 i o ile duże Amigi posiadały dyski twarde, sloty Zorro i slot procesora, to w zasadzie jedyną rozbudową przewidzianą przez konstruktorów Amigi 500 były rozszerzenia pamięci. Również twórcy gier na Amigę z zawziętym zapalem pilnowali, aby wymagania gry nie przekroczyły standardu OCS/ECS i 1 MB RAM. Co więcej, gry często odwoływały się bezpośrednio do sprzętu z pominięciem systemowych procedur, co powodowało kłopoty na rozbudowanych konfiguracjach, dodatkowo zniechęcając do rozbudowy komputera. Stąd też kart takich było niewiele, a istniejące tzw. karty turbo były kłopotliwe w montażu i często wymagały rozkręcenia całej Amigi. Wraz z pojawieniem się następcy A500 – modelu A1200 – twórcy wreszcie zauważyli, że dysk twardy i możliwość dodania czegoś więcej niż tylko dodatkowej pamięci w domowym komputerze, to nie tylko przywilej, ale absolutna podstawa. W nowych komputerach uwzględniono wreszcie możliwość rozbudowy o karty z szybszym procesorem, stąd też pojawienie się slotu „Local Bus Slot”, który ułatwił montaż tychże kart. Instalacja systemu operacyjnego na dysku twardym i uruchomienie czegokolwiek szybko uświadomiła użytkownikom, że do komfortowej pracy 2 MB pamięci to za mało. Co więcej, do komfortowej pracy z poważnymi programami standardowy procesor też nie jest demonem wydajności. Szybko zatem pojawiały się coraz to nowe karty rozbudowujące Amigę początkowo o dodatkowy RAM oraz z procesorami MC68030, a nieco później bardziej zaawansowane z jeszcze szybszymi. Obecnie, poza nielicznymi wyjątkami, karty turbo są to prawie wyłącznie kilkunastoletnie konstrukcje, jednak o dziwo wciąż świetnie działające. Należy jednak zwrócić uwagę, że niektóre są dość kapryśne – szczególnie że po latach jakość sygnałów naszej Amigi jest dużo gorsza niż kiedyś.

Montaż

Wszystkie dostępne dla Amigi A1200 akceleratory są montowane pod klapkę w slotcie procesora. Montaż jest bardzo prosty i składa się zazwyczaj z trzech kroków: zdjęcia klapki, wsunięcia karty na slot oraz zamknięcia klapki. Slot jest skonstruowany w ten sposób, że nie jest możliwe odwrotnie założenie karty. Po montażu funkcja „autoconfig” powinna automatycznie rozpoznać dodatkowe urządzenie i z założenia żadna dodatkowa ingerencja nie powinna być wymagana. W praktyce jednak niektóre bardziej rozbudowane karty mogą mieć zworki umożliwiające konfigurację – zazwyczaj dotyczące ilości pamięci bądź obecności kontrolera SCSI. Konfiguracja każdej karty jest banalnie prosta i szczegółowo opisana w dołączonej instrukcji. Warto także wspomnieć, że niektóre karty tzw. „high end” mogą wymagać obudowy tower ze względu na wydzielane ciepło bądź rozmiary (na przykład drugi SIMM lub duży radiator). Warto zwrócić również uwagę, że niektóre karty grzeją się do

tego stopnia, że zamknięcie klapki może powodować ich przegrzanie i okazyjne zawieszanie się systemu. Inne wręcz uniemożliwiają jej zamknięcie.

Oprogramowanie

Dodatkowa pamięć oraz karty oparte na procesorach MC68020/030 nie wymagają instalacji żadnego dodatkowego oprogramowania i system wykorzystuje je zaraz po zainstalowaniu. Karty oparte o procesory MC68040/060 to znacznie bardziej skomplikowane konstrukcje i w tym przypadku należy skopiować bibliotekę *68040.library* (lub *68060.library*) do katalogu *Libs*. Warto też nadmienić, że w przypadku niektórych kart istnieje możliwość zainstalowania specjalnego oprogramowania oferującego dostęp do dodatkowych możliwości konkretnego modelu karty (na przykład *Blizkick* dla kart Blizzard lub *RemApollo* dla kart Apollo). Dodatkowo istnieją też dedykowane wersje oprogramowania specjalnie zoptymalizowane pod konkretne modele procesorów, które można używać w celu bardziej komfortowej pracy (jednak nie jest to wymagane). Często też oprogramowanie przy instalacji wykrywa typ procesora i proponuje prawidłową wersję. Znacznie bardziej skomplikowana jest sprawa z kartami PowerPC. Aby z nich skorzystać pod AmigaOS 3.x, należy zainstalować system wykorzystujący ten procesor oraz dedykowane oprogramowanie.

Karty FastRAM

Przyspieszenie	około 2x
Pamięć	do 9 MB
CPU	NIE
RTC	TAK
FPU	podstawka pod montaż

Karty FastRAM to najbardziej podstawowe karty przyspieszające Amigę. Zazwyczaj oferują rozbudowę pamięci operacyjnej o 4 lub 8 MB, choć istnieją rozwiązania mogące rozbudować Amigę nawet o 9 MB. Dzięki zastosowaniu szybkiej pamięci Fast RAM nasza Amiga 1200 przyspiesza prawie dwukrotnie. Dzięki odciążeniu pamięci Chip, oprócz samego procesora przyspieszają również pozostałe układy specjalizowane, a więc także i operacje graficzne. Pamięć na kartach może być przylutowana lub też występować w postaci modułów SIMM lub ZIP. Zazwyczaj karty dodatkowo posiadają RTC (z ang. Real-Time Clock – zegar czasu rzeczy-

wistego) oraz miejsce dla FPU (z ang. Floating Point Unit – koprocesor). Dzięki zwiększeniu ilości dostępnej pamięci na Amidze 1200 można używać najnowszych systemów w wersji 3.5 i 3.9, choć komfort pracy tak naprawdę zapewniają właściwie tylko z systemem w wersji 3.0 i 3.1.

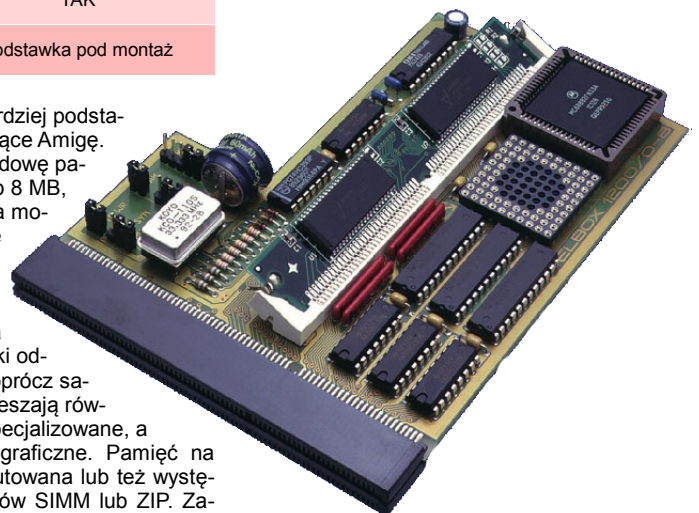
Oprócz kart istnieją również karty rozszerzenia pamięci pod PCMCIA. Taka pamięć jest jednak wolniejsza i gorzej wygląda sprawa z jej dostępnością. Istnieje również możliwość zakupu slotów Zorro II z podstawkami SIMM. Takie rozwiązanie wymaga jednak przełożenia Amigi 1200 do obudowy tower. Ponadto ten sposób rozszerzenia lubi sprawiać problemy i jest rzadko spotykany. Ze względu na ograniczoną, 24-bitową przestrzeń adresową standardowego procesora, w ten sposób możemy maksymalnie osiągnąć do 8 MB dodatkowej pamięci. I jeszcze drobna uwaga – karty z 8 MB pamięci mogą blokować port PCMCIA. Wynika to z ograniczeń 24-bitowej szyny adresującej procesora Amigi MC68EC020. Karta jest adresowana w przedziale 200000-9ffff, który jest jednocześnie przypisany PCMCIA, stąd po wypełnieniu ponad 4 MB wystąpią problemy w dołączonym urządzeniu.

Najpopularniejsze w Polsce modele to:

- **Elbox 1204** – zamontowane na stałe 4 MB pamięci, podstawka pod FPU, RTC.
- **Elbox 1208/Elsat 1208** – gniazdo na RAM SIMM 72 pin do 8 MB, podstawka pod FPU, RTC.

Spotyka się również modele produkowane przez firmę M-TEC.

Temat kart FastRAM jest znacznie szerzej eksploatowany na poletku Amigi 500/600 z racji ograniczonych możliwości rozbudowy.



Elbox 1208

Zdjęcie: Amiga Hardware Database
<http://amiga.resource.cx>



Karty budżetowe

Przyspieszenie	około 6x
Pamięć	do 8 MB
CPU	68020, 68030
RTC	TAK
FPU	podstawka pod montaż

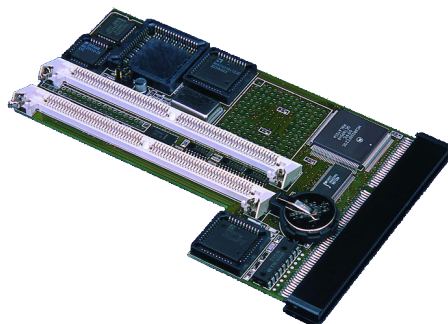
Budżetowe konstrukcje rozbudowują Amigę o procesor MC68020 lub MC68030. Zwykle mają ograniczenie pamięci do 8 MB, która występuje w postaci modułów SIMM, wyjątkowo ZIP. W przeciwieństwie do rozszerzeń RAM otrzymujemy również większą prędkość działania ze względu na nowy procesor. Warto przy tym zauważyć, że zegar MC68030 posiada podobną wydajność co MC68020. Przewaga MC68030 to jednak szybszy dostęp do pamięci, tryb BURST, pamięć cache zarówno dla instrukcji, jak i danych oraz praca ze znacznie większymi częstotliwościami (dla 020 do 28 MHz, dla 030 do 50 MHz). Ponadto na dokładkę pozostaje nam jeszcze pole do popisu przy OC. Pomimo nowego procesora karty wciąż posiadają problem z zaadresowaniem więcej niż 8 MB pamięci, co wynika z budowy Amigi 1200, która w normalnych warunkach adresuje tylko 8 MB na złączu CPU. Warto zauważyć, że MC68030 występuje też w wersji EC, różniącą się brakiem jednostki MMU.

Najpopularniejsze w Polsce modele to:

- **ELBOX 1230-LC** – maksymalnie 8 MB pamięci, procesor MC68030/33MHz przetaktowany do 40 MHz, RTC, opcjonalnie koprocessor.
- **M-tec 1230 MC68030/28-56 MHz** – maksymalnie 8 MB pamięci, RTC, opcjonalnie koprocessor, opcjonalnie SCSI.

Klasa średnia

Przyspieszenie	około 7x
Pamięć	do 128 MB
CPU	68030
RTC	opcjonalnie
FPU	podstawka pod montaż lub TAK



Elbox 1230 Turbo MK3

Zdjęcia: Amiga Hardware Database
http://amiga.resource.cx



Blizzard 1230-IV

Karty oparte o procesor MC68030 zazwyczaj są taktowane częstotliwością 40-50 MHz. Karty posiadają MMU oraz miejsce na koprocessor. Standardowe 72-pinowe gniazda SIMM umożliwiają tanią rozbudowę. Uwaga jednak na karty GVP – posiadają specjalne, właściwie niedostępne kości pamięci. Warto zaznaczyć, że karty posiadają pełną, 32-bitową przestrzeń danych, co oznacza, że nie blokują złącza PCMCIA. Niektóre karty posiadają możliwość montażu kontrolerów SCSI. Amiga była projektowana do współpracy właśnie z kartami turbo MC68030, zatem karty te mają najszybszy dostęp do płyty głównej, dzięki czemu pracuje z nimi znacznie szybciej kontroler IDE oraz chipset AGA oferując największą kompatybilność. Karty szczególnie polecane do wykorzystania w połączeniu z WHDLoad. Rozbudowa do 128 MB pamięci robi duże wrażenie, jednak programy, które przyzwyciężają na tych kartach, nie wykorzystują więcej niż 32 MB. Nowsze konstrukcje, czyli karty ACA, dzięki zastosowaniu bardzo szybkiej pamięci RAM przy tym samym taktowaniu są około 20% szybsze od swoich starszych odpowiedników.

Najpopularniejsze modele to:

- **ELBOX 1230-II EC030/40 MHz** – maksymalnie 64 MB pamięci w dwóch slotach, procesor MC68030/33 MHz przetaktowany do 40 MHz, opcjonalnie koprocessor, RTC.
- **Blizzard 1230 MK-IV** – MC68030/50 MHz, złącze dla modułu Fast-SCSI-II, 128 MB pamięci (256 MB z modułem SCSI), koprocessor 68882/50 MHz, RTC.
- **ACA 1230** – procesor EC68030/28 MHz lub 42 MHz, bardzo szybka pamięć 64 MB, opcjonalny RTC.

Górna półka

Przyspieszenie	Od 25 do 60x
Pamięć	do 256 MB
CPU	68040, 68060
RTC	TAK
FPU	podstawka pod montaż lub TAK



Apollo 1240

Karty z procesorami MC68040 oraz MC68060 są najszybszymi kartami do Amigi Classic opartymi o architekturę MC680x0. Ze względu na brak niektórych rozkazów w tych procesorach, karty te są nieco mniej kompatybilne niż na przykład te wyposażone w MC68030. Do poprawnego działania wymagana jest biblioteka *68040.library* lub *68060.library*. Ponadto Amigi nie były projektowane pod te procesory, w związku z czym karty wymagają układów pośredniczących między Amigą i procesorem, co wprowadza dodatkowe opóźnienie. W efekcie procesor ma nieco wolniejszy dostęp do zasobów płyty głównej. Ważną zmianą w tych procesorach jest wbudowanie nowego, znacznie wydajniejszego od poprzedników koprocessora. Karty polecane głównie użytkownikom, którzy mają już pewne doświadczenie z Amigą. Wszelkie swoje niedogodności karty kompensują znacznie większą prędkością działania. Dodatkowym atutem jest sensowna możliwość korzystania z emulatorów Macintosh, a tym samym szerokiej bazy gier i programów wydanych na tę platformę, będących trochę bardziej w czasie. Wreszcie dopiero na tych kartach można uruchomić najbardziej zaawansowane gry i programy na Amigę. Podobnie jak poprzednie karty, również tutaj często dodawane są złącza na kontroler SCSI. Procesor MC68060 jest obecnie najszybszym procesorem z rodziny MC680x0. Karty z tym procesorem taktowane są częstotliwością 50 MHz, jednak można spotkać wersje podkręcone do 80 MHz, a nawet 100 MHz. Warto również nadmienić, że karty te mogą wymagać mocniejszego niż standardowy zasilacza.

Najpopularniejsze modele to:

- **Blizzard 1240/1260** – procesor 68040/40 MHz lub 68060/50 MHz, złącze dla modułu Fast-SCSI-II, 128 MB pamięci (256 MB z modułem SCSI).
- **Apollo 1240/1260** – procesor 68040/25-40 MHz lub 68060/50 MHz, moduł SCSI, 32 MB pamięci (64 MB z drugim slotem).



Karty PowerPC

Przyspieszenie	dla PowerPC około 350x
Pamięć	do 256 MB
CPU	68040 lub 68060 i PowerPC 603e 160-240 MHz
RTC	TAK
FPU	podstawka pod montaż lub TAK

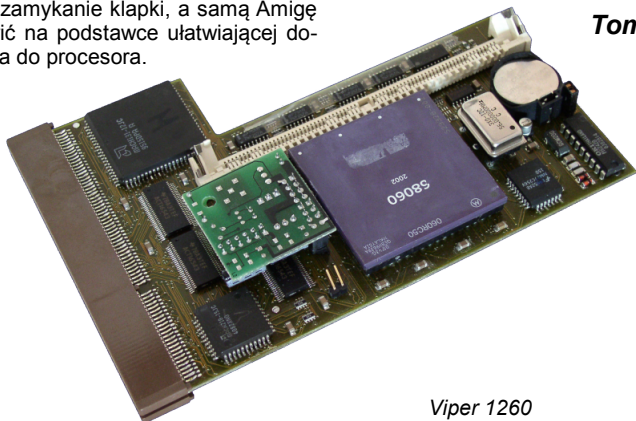
Są to najszybsze karty, jakie wyprodukowano do Amigi Classic. Ta dwuprocessorowa konstrukcja to istne dzieło sztuki, gdzie na małym PCB umieszczono procesor Motorola MC680x0 oraz procesor z serii PowerPC 603e. Do tego upchano dwa sloty SIMM, kontroler SCSI oraz szybkie złącze karty graficznej. Dzięki procesorowi PowerPC na kartach uruchomimy system operacyjny AmigaOS 4 Classic oraz MorphOS. Procesor PPC w wersji dla Amigi 1200 cierpi na wydajności ze względu na zastosowanie pamięci typu SIMM i szynę tylko 32-bitową. Dodatkowym obciążeniem dla i tak już wolnej pamięci jest przełączanie się procesorów w trakcie dostępu do pamięci. W internecie można spotkać wersje z procesorem MC68060 podkręconym do 66 MHz i 603e podkręconym nawet ponad 300 MHz.

Dla Amigi 1200 występuje tylko jeden model:

- **Blizzard PPC 603e** – procesor MC68040/25 MHz lub MC68060/50 MHz i PowerPC 603e/160, 200 lub 240 MHz, pamięć maksymalnie 256 MB, opcjonalnie slot Fast-SCSI-II (w tak zwanej wersji 603e Plus).

Która karta dla mnie?

Zakładając, że przeciętny gracz retro będzie używał pakietu *WHDLoad*, wówczas do grania wystarczy dowolna karta posiadająca 8 MB pamięci (plus karta CF). Szybsze karty są wymagane w przypadku, gdy spodoba nam się nowsza wersja systemu operacyjnego czy też zaczniemy upiększać nasz pulpit. Szybko wówczas odkrywamy, że komfort pracy zaczyna się od kart MC68030 i 16 MB pamięci. Jeśli Amigi chcemy również używać do emulacji Macintosha (a warto) oraz oglądania możliwie wszystkich produkcji scenowych, dobrym procesorem będzie MC68040/40 MHz lub nawet MC68060. Kupując taką kartę warto też zwrócić uwagę czy zmieści się w obudowie desktop oraz, czy zbyt bardzo się nie grzeje. W tym ostatnim przypadku pewnym rozwiązaniem może być niezamykanie klapki, a samą Amigę można ustawić na podstawce ułatwiającej dopływ powietrza do procesora.



Viper 1260



Blizzard PPC 603e

Problemy z kartami, czyli co robić, gdy nie działa?

1. Sprawdzamy kontakt karty ze złączem krańcowym, czyścimy złącze, na przykład alkoholem izopropylowym, odginamy nóżki na karcie tak, aby na pewno kontaktowały.
2. Pamięć RAM może być niekompatybilna. Sprawdzamy kartę bez modułów pamięci.
3. Mierzmy napięcia zasilania karty. Karty turbo do Amigi pobierają napięcie z linii 5V, dlatego należy zadbać, aby napięcie mieściło się w granicach 5V +/- 0.25V.
4. W prototypach chipsetów AGA był problem z timingami, który wymagał fixa. Chipsety usprawniono, ale fix pozostał, pogarszając jakość sygnału na wyjściu. Efektem mogą być niestabilne działania kart turbo zwłaszcza Apollo 1240/1260. Aby go usunąć, należy z płyty głównej wylutować dwa kondensatory SMD: E123C i E125C.
5. Kolejnym problemem Amigi są stare kondensatory. Kondensatory elektrolityczne po kilku latach zaczynają wysychać i tracić właściwości. Może to prowadzić do kłopotów ze stabilnością, a w skrajnych przypadkach wycieku elektrolitu z kondensatorów na płytę główną i tym samym korozję ścieżek. Kondensatory najlepiej zastąpić kondensatorami tantalowymi lub zwykłymi, dobrej jakości, o podwyższonej żywotności. W Amidze nie są wymagane kondensatory LRS.
6. Czerwony ekran sygnalizuje problem z ROM-em. Należy go docisnąć, w razie potrzeby przeczyścić złącze spirytusem i ponownie umieścić we właściwym miejscu.

Tomasz Magosa

Słowniczek pojęć

DMA – Direct Memory Access – bezpośredni dostęp urządzenia do pamięci z pominięciem procesora. Dzięki obsłudze DMA urządzenie przy korzystaniu z pamięci nie obciąża procesora.

FPU – Floating Point Unit – jednostka zmiennoprzecinkowa, w zasadzie w systemie i grach na Amidzie nie jest używana. Moc obliczeniową FPU można jednak wykorzystać dzięki specjalnym bibliotekom (np. *HSMATHLibs* – pakiet zamienników systemowych bibliotek matematycznych). W przypadku procesorów 68020 i 68030 należy wyposażyć się w osobny koprocesor 68881 lub 68882. Procesory 68040 i 68060 mają wbudowane koprocesory wiele razy wydajniejsze od wspomnianych wcześniej (istnieją też wersje LC040 i LC060 bez koprocesorów). Na płycie głównej Amigi A1200 jest miejsce na wlotowanie koprocesora MC68881/MC68882 w obudowie PLCC, jednak na razie nikt się jeszcze takim wyczynem nie pochwalili.

MMU – Memory Management Unit – jednostka zarządzania pamięcią wymagana przy używaniu pamięci wirtualnej, czyli w zasadzie niewykorzystywana przez Amigę. Z punktu widzenia użytkownika może być przydatna jedynie przy mapowaniu Kickstartu. Ponoć *WHDLoad* wykorzystuje MMU, jednak równie dobrze działa bez. MMU jest wymagane w przypadku instalacji Linuksa na Amidzie. Na płycie głównej Amigi 1200 jest miejsce na zewnętrzny moduł MMU MC68851, jednak z tego, co wiem, nikt nie podjął się jego zamontowania.

Pamięć Chip – pamięć graficzna Amigi. Jest wolna (i zawsze jej brakuje), wykorzystywana głównie przez chipset AGA. Amiga 1200 posiada 2 MB pamięci Chip i nie da się jej rozbudować.

Pamięć Fast – pamięć na kartach rozszerzeń Amigi, zazwyczaj od kilku do kilkadziesiąt razy szybsza od pamięci Chip. Dodanie takiej pamięci automatycznie przyspiesza nawet zwykłą Amigę 1200 około 2 razy. W Amigach Classic, poza nielicznymi wyjątkami, stosuje się pamięci typu SIMM. Najlepszym wyborem jest SIMM typu FPM, natomiast pamięci typu EDO mogą stwarzać problemy z niektórymi kartami. W kartach turbo zaleca się pamięć o czasie dostępu 60ns. Jeśli podkręcamy kartę, konieczne mogą okazać się rzadko spotykane pamięci 50ns (tylko EDO). Szybsze pamięci można spotkać tylko w kartach graficznych.

RTC – Real Time Clock – zegar czasu rzeczywistego, który podtrzymuje ustawiony czas po wyłączeniu Amigi. Na płycie głównej Amigi 1200 jest przewidziane miejsce na RTC, jednak ponownie, nie znam nikogo, kto podjął się jego zamontowania. Uwaga – stare akumulatory Ni-Cd podtrzymujące zasilanie RTC lubiły przeciekać, powodując przy okazji korozję ścieżek na PCB i niszcząc karty. Obecnie stosuje się baterie 3V.

SCSI – standard obsługi dysków, podobny do IDE, jednak dedykowany do profesjonalnego sprzętu. SCSI umożliwia obsługę dysków, napędów optycznych, a nawet skanerów. Zaletą SCSI jest większa prędkość od odpowiedników IDE oraz sprzętowy kontroler, który odciąża procesor przy przesyłaniu danych i korzysta z DMA. Standard ten wymaga dedykowanych urządzeń SCSI, obecnie trudno dostępnych i od dawna nieprodukowanych. Najczęściej w kartach turbo stosowany jest kontroler SCSI II/Fast SCSI II.



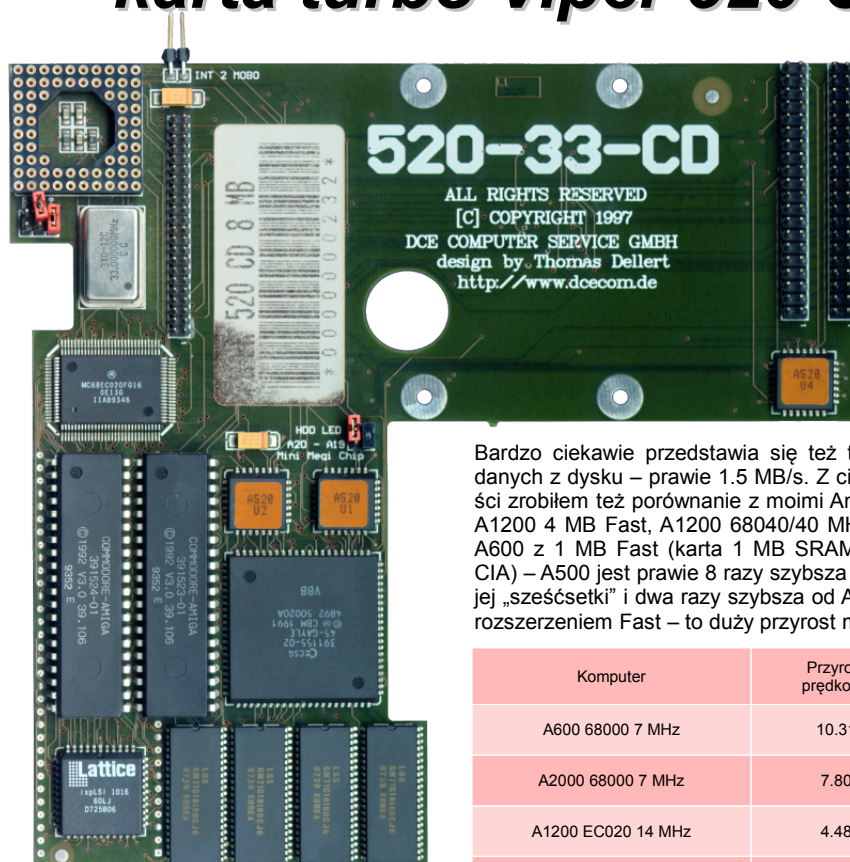
Turbo „Pięcsetka” – karta turbo Viper 520 CD

Amiga 500 to wspaniały komputer. Do tego modelu powstała ogromna ilość rozszerzeń, lecz niestety z racji wieku komputera niewiele z nich dotrwało do dzisiejszych czasów. Czasami jednak można trafić na rarytas. Udało mi się znaleźć i przetestować kartę turbo do Amigi 500 – Viper 520 CD.

Konfiguracja karty

Karta wyposażona jest w procesor MC68020 – 33 MHz (jest to wersja 16 MHz, fabrycznie przetaktowana do 33 MHz), koprocesor MC68881 – 33 MHz (także tutaj producent zdecydował się na przetaktowanie z 20 MHz na 33 MHz). Karta posiada na stałe wlotową pamięć Fast 8 MB, ROM 3.0 oraz dwa gniazda IDE 3.5" i jedno 2.5". Mimo przetaktowania CPU i FPU, karta pracuje bardzo stabilnie – nie się nie wiesza, a same układy po kilku godzinach ciężkiej pracy są tylko delikatnie nagrzane. Sam montaż karty nie jest skomplikowany – wkładamy ją w miejsce procesora MC68000. Po zamontowaniu karty podpiąłem jeszcze do gniazda IDE 2,5" kartę CF 2 GB, na której został zainstalowany AmigaOS 2.1.

CPU	MC68020 33.3 MHz
FPU	MC68881
MMU	brak
Custom graphic chip	ECS Denise 8373 (rev 0)
Custom animation chip	ECS PAL Fatter Agnus 8372a, 1 MB
Other custom chips	Paula 8364 (rev 0), Gayle (rev 13)
Graphics system	ECS
RTC	tak
ROM	39.106 (Kickstart 3.0)
Fast RAM	8 MB



Testy, testy, testy

Pierwszy „test” to samo uruchomienie tak rozbudowanej A500. Już podczas bootowania widać przyspieszenie Amigi – system wczytuje się kilka sekund. Praca w systemie też jest zdecydowanie szybsza – nie trzeba czekać na narysowanie okien, także otwieranie katalogów jest niemal błyskawiczne. Po wgraniu systemu pierwszą aplikacją, jaką uruchomiłem był program *WhichAmiga*. Trochę dziwne, że program nie podał 2 MB pamięci Chip, ponieważ A500 miała zamontowane rozszerzenie 1 MB Chip „pod klapkę”, ale systemowy program *Show-Config* poprawnie rozpoznał konfigurację Amigi, podając 2 MB Chip, 8 MB Fast i procesor 68020. Następnie przyszedł czas na pierwszy prawdziwy test, który przeprowadziłem za pomocą nieśmiertelnego programu – *SysInfo* 3.24. Zostałem poinformowany, że posiadam A500 cztery razy szybszą od gołej A1200 (!!!) – najważniejsze jednak iż tak rozbudowana A500 jest wg *SysInfo* szybsza od swojej dużej siostry, Amigi 3000, aż o 17%.

Bardzo ciekawie przedstawia się też transfer danych z dysku – prawie 1.5 MB/s. Z ciekawości zrobiłem też porównanie z moimi Amigami: A1200 4 MB Fast, A1200 68040/40 MHz oraz A600 z 1 MB Fast (karta 1 MB SRAM PCMCIA) – A500 jest prawie 8 razy szybsza od mojej „sześćsetki” i dwa razy szybsza od A1200 z rozszerzeniem Fast – to duży przyrost mocy.

Komputer	Przyrost prędkości
A600 68000 7 MHz	10.31
A2000 68000 7 MHz	7.80
A1200 EC020 14 MHz	4.48
A2500 68020 14 MHz	2.65
A3000 68030 25 MHz	1.17
A4000 68040 25 MHz	0.29
A600 68000 7 Mhz (1 MB Fast)	7.79
A1200 68020 14 MHz (4 MB Fast)	2.02
A1200 68040 40 MHz	0.12

Po „suchym” teście w programie *SysInfo* przyszedł czas na najlepsze – gry. „*Doom I*” – gra legenda, która nigdy nie miała zaistnieć na Amidze (a przynajmniej tak twierdził jej twórca). Niestety, Amiga 500 z kartą Viper nie przejmowała się tym, co powiedział Carmack i uruchomiła tę grę. Gdy ustawimy maksymalne okno w rozdzielczości 320x256 EHB to widać, że animacja nie jest płynna. Wystarczy jednak ustawić rozmiar 60% pełnego okna i już mamy płynną rozgrywkę. Oczywiście z powodu zastosowania kości ECS w A500, „*Doom*” nie jest tak kolorowy, jak na A1200 lub karcie graficznej, niemniej jednak zabawa i tak jest wspaniała, jak się pomyśli, że to przecież Amiga 500, która powstała w czasach prostych platformówek. Po krótkiej zabawie „*Doomem*” przyszedł czas na bardziej amigową grę – „*Gloom Deluxe*”. W tej grze też nie było problemów – instalacja przebiegła szybko i po kilku sekundach ukazał się ekran tytułowy gry, a zaraz potem można było grać. Tak samo jak w przypadku gry „*Doom*”, tutaj też trzeba było ustawić jakieś trzy wielkości okna, aby gra była płynna. Wy-



Trudno w to uwierzyć – „Doom” na Amidze 500



daje mi się też, że na kościach ECS „Gloom Deluxe” wydaje się trochę ładniejszy od „Dooma”. Pracuje również szybciej na A500 z kartą Viper niż na A1200 z pamięcią Fast. Po zabawie z „Gloomem” przyszedł czas na gry WHDLoad. Tutaj też nie ma najmniejszych problemów. Wszystkie testowane gry („S.W.I.V.”, „Lotus”, „Test Drive”, „Chuck Rock 2”, „Golden Axe”, „Railroad Tycoon”) uruchamiają się błyskawicznie, a część gier, które wymagają szybszego procesora, działa zdecydowanie płynniej. Mój ulubiony „Railroad Tycoon” na mojej A600 działał bardzo opornie - początkowe animacje, generowanie mapy czy późniejsza gra, gdy mam już mocno rozbudowane imperium jest... wolna (delikatnie mówiąc). Przy procesorze 020-33 MHz i 8 MB Fast ta gra wręcz śmiga – teraz zabawa to czysta przyjemność.

Po dłuższej zabawie grami przyszedł czas na użytki. Filemaster działa błyskawicznie (kopiowanie, otwieranie katalogów, przeglądanie katalogów itd.), DOPUS także śmiga. Z ciekawości uruchomiłem DPaint IV i obrazek w HAM6 – wczytanie obrazka było błyskawiczne, praca na takim obrazku też. W porównaniu do gołej A500 lub do mojej A600 z Fast, DPaint IV chodził przynajmniej kilkanaście razy szybciej. Podczas testów karty przypadkowo nadarzyła się możliwość przeprowadzenia jeszcze jednego sprawdzianu – naprawy dysku za pomocą programu DiskSalv. Naprawa walidacji na partycji 1.2 GB, gdzie nagrane było około 15000 plików, trwała około 20 minut – myślę, że to bardzo dobry wynik.

Podsumowanie

Karta zrobiła na mnie ogromne wrażenie. Jest stabilna, szybka, wyposażona w gniazda IDE, ROM 3.0. Wadą karty jest niestety wlutowana na stałe pamięć i ROM, przez co nie ma możliwości wymiany kości na ROM 3.1, ale poza tą drobną wadą, karta posiada same zalety. Praca i zabawa na tak dopalanej Amidze 500 jest naprawdę bardzo przyjemna, a „Doom” uruchomiony na tym komputerze to niesamowite przeżycie.



„Deluxe Paint”

Przepraszam za jakość zdjęć, ale niestety wszystkie musiały zostać zrobione aparatem, gdyż z niewiadomych przyczyn ScreenGrab czy Grabber podczas próby zrobienia screenshota wieszały cały system.

Na zakończenie, chciałbym serdecznie i gorąco podziękować Basi znanej na PPA jako BA-CUS10 za wypożyczenie karty do testów.

Aleksander „trOLLO” Giedyk



„Gloom Deluxe”



A500 z Viperem na pokładzie



- procesor MC68EC020 – 33 MHz
- miejsce na koprocesor MC68881
- 8 MB pamięci Fast
- dwa gniazda IDE 3.5"
- gniazdo IDE 2.5"
- kickstart 3.0
- prosty montaż



- pamięć Fast wlutowana na stałe
- kickstart wlutowany na stałe, przez co nie można go wymienić na np. 3.1
- montaż wymaga wyjęcia MC 68000
- niektóre gry nie lubią MC68EC020



HxC SD Floppy Emulator rev F

Od około dwóch lat na rynku dostępny jest *HxC SD Floppy Emulator* – urządzenie pozwalające na zastąpienie góry dyskietek jedną kartą SD. Podłączane jest ono w miejsce oryginalnej stacji dyskietek. Interfejs urządzenia umieszczony od strony komputera jest w 100% kompatybilny z amigowym interfejsem stacji dyskietek. Recenzję oryginalnej wersji *HxC SD Floppy Emulator* („rev B”) przeczytać można na portalu PPA (polecam jej przeczytanie przed przystąpieniem do lektury tego artykułu).

W połowie tego roku w ofercie producenta pojawił się nowy model, oznaczony jako „rev F”. Posiada on wszystkie usprawnienia funkcjonalne, które pojawiły się w oryginalnym *HxC SD Floppy Emulator*. Wśród tych usprawnień najistotniejsza jest możliwość zapisu na amigowe obrazy dyskietek (dodana w „rev C”). Ponadto „rev F” korzysta z dokładnie tego samego firmware, który używany jest w „rev C”, więc możliwości urządzenia są w zasadzie identyczne.

Co wyjątkowego jest w nowej wersji „rev F”? Rozmiar. Na pierwszy rzut oka, nie jest to dramatyczna zmiana, ale w przeciwieństwie do oryginalnej wersji, „rev F” przeznaczona jest do instalacji w standardowej zatoce 3,5 cala i posiada estetyczny panel przedni tego samego rozmiaru. Nowa wersja została opracowana, aby stanowić idealny zamiennik zwykłej stacji dyskietek. To jest coś, na co przede wszystkim czekali użytkownicy Amig w obudowach tower oraz desktop (A2000, A3000T, A4000, A4000T). W większości wypadków wystarczy wyjąć oryginalną stację i na jej miejsce podłączyć *HxC Floppy Emulator rev F*. Odpada konieczność „rzeźbienia” w obudowie komputera, co niestety często było konieczne przy instalacji poprzedniej wersji. Niestety, w niektórych Amigach (np. A500, A600, A1200, A3000), o ile sama zatoka stacji jest standardową zatoką 3,5 cala, to niestety panel przedni jest nietypowy. Problem ten można rozwiązać poprzez wycięcie części obudowy, gdzie znajduje się panel. Niektórzy użytkownicy będą jednak

mieli zrozumiałe obiekcje przed cięciem obudowy zabytkowego komputera. W tej sytuacji rozsądnym wyjściem wydaje się zakup zewnętrznej stacji dyskietek oraz wymiana w niej samego napędu na emulator. Niektóre Amigi mają nawet zworkę, która pozwala na używanie stacji zewnętrznej jako DF0.

To, co rzuca się w oczy po wyjęciu emulatora „rev F” z opakowania, to jak solidnie jest on wykonany. Jego podstawa oraz panel przedni wykonane są z jednego kawałka metalu, więc jest on dość ciężki (nie ma nawet zbyt dużej różnicy w wadze między stacją dyskietek a emulatorem). Sama płytk drukowana także wygląda profesjonalnie i nowocześnie. W pudełku oprócz emulatora dostajemy dwie naklejki na panel przedni (w kolorze czarnym i białym), a także zestaw zwork.

Instalacja emulatora w obudowach, które posiadają standardowe zatoki i panele 3,5 jest trywialna. Polega na odłączeniu taśmy oraz zasilania od oryginalnej stacji, wyjęciu jej, włożeniu na jej miejsce emulatora oraz podłączeniu taśmy i zasilania do emulatora. Na panelu przednim znajduje się wyświetlacz LCD, na którym prezentowane są informacje o stanie urządzenia (m. in. obecnie załadowanym obrazie dyskietki), dwie diody LED, slot na karty SD oraz trzy przyciski do sterowania pracą emulatora (zmiany załadowanych obrazów). Obsługa oraz sposób działania są identyczne z wersją „rev B” i „rev C”, więc nie można tu wiele dodać.



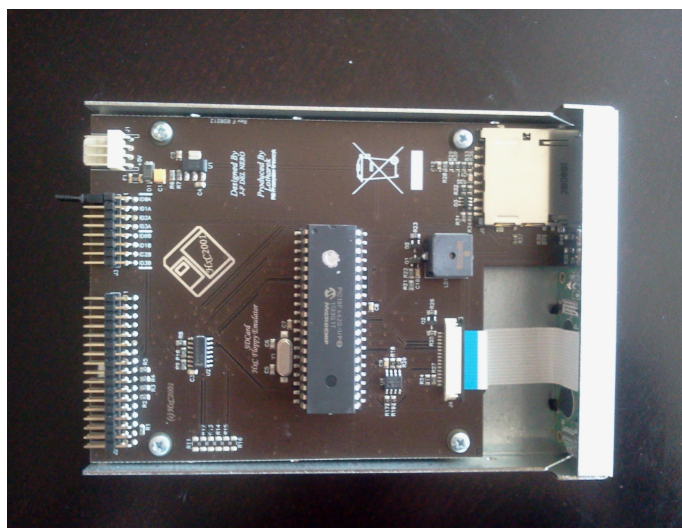
ku gdy Amiga nie posiada dysku twardego oraz rozszerzenia pamięci.

Osoby zainteresowane emulatorem powinny odwiedzić stronę producenta. Przejrzeć tam można pełną specyfikację urządzenia oraz dokonać zakupu. W momencie pisania tego artykułu cena wynosiła 110 euro (za wersję „rev F”).

Radosław "strim" Kujawa

HxC SD Floppy Emulator rev F

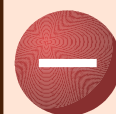
<http://www.lotharek.pl>



Według mnie *HxC SD Floppy Emulator* jest urządzeniem, którego zakup powinien rozważyć każdy, kto aktywnie używa Amigi klasycznej. Jak wiadomo, z każdym dniem trudniej zakupić jest nową paczkę dyskietek. Z czystym sumieniem mogę polecić *HxC Floppy Emulator rev F* jako zamiennik stacji dyskietek. Należy dodać, że jest to zamiennik znacznie bardziej komfortowy. *HxC Floppy Emulator* może też stanowić alternatywę dla pakietu WHDIload, w przypad-



- 100% kompatybilny zamiennik stacji dyskietek
- pasuje idealnie do zatok 3,5 cala
- solidna a zarazem estetyczna konstrukcja
- posiada wszystkie funkcje *HxC Floppy Emulator rev C*



- Instalacja wewnątrz małych Amig wymaga modyfikacji obudowy



Sqrzx (Remake)

Gra o tak dziwnym tytule to platformówka, w której liczy się refleks oraz szybkość kojarzenia faktów i zdarzeń. Jej autorami są Marcus Vesterlund oraz John Holmvall, którzy w 1996 roku stworzyli grę w wersji dla DOS oraz Windows. Po wielu latach tytuł został przepisany w bardziej „otwarte” środowisko z wykorzystaniem biblioteki SDL. Pieczę nad kodem sprawuje obecnie Rodolphe Boixel, który przygotował „Sqrzx” już w trzeciej odsłonie. Gra dostępna jest chyba dla wszystkich możliwych platform systemowych, nie zapominając także o wersjach dla AmigaOS 4, AROS-a oraz MorphOS-a, które zostały przeprotowane m. in. dzięki Frankowi Wille. Ta recenzja z kolei dotyczy jeszcze innej wersji tej gry – mianowicie „Sqrzx (Remake)”, będącej konwersją oryginalnej gry, napisanej na podstawie kodu Rodolphe Boixela, przeznaczoną dla Amig klasycznych. Tak, to nie pomyłka – gra nie jest portem (configure & make) o niewyobrażalnych wymaganiach sprzętowych, lecz konwersją dostosowaną do amigowych realiów i możliwości.

Co to za zwierz?

Dosłownie i w przenośni – królik. Długouchy i skoczny musi pokonać szesnaście etapów zbierając napotkane żółte kwiatki. Po drodze musi unikać przeszkadzajek w postaci blobów, języków czy choćby piranii. Niektóre z nich, takie jak właśnie wspomniane bloby, może zabić lub wykorzystać jako trampolinę, zyskując możliwość doskoczenia do wyżej położonych platform. Większość szwędających się „bestii” pozostanie niestety nieugięta naporowi skoków zająca, doprowadzając go do niechybnej śmierci. Ruchome potworki na naszej drodze to jednak nie wszystko, z czym przyjdzie się zmierzyć. Zagrożenie stanowią również pułapki w postaci kółców (widocznych i niewidocznych) czy opadających krat blokujących wyjście z korytarza (które jednak można wykorzystać jako element taktyki na zablokowanie drogi, na przykład jeżowi). Pamiętajmy też, że to, co raz zniknęło za ekranem, nieodwracalnie tam pozostaje – nie ma możliwości powrotu w odwiedzone miejsca. Na pokonanie etapu autorzy wyznaczyli określoną ilość czasu, która obrazowana jest w postaci czerwonej kreski w dolnej części ekranu. Drugi wskaźnik, koloru niebieskiego, to tlen, który ubywa wraz z naszym przebywaniem pod wodą – kolejne utrudnienie (lub uatrakcyjnienie zabawy – jak kto woli).

Gra może wydawać się bardzo prosta i szybka



do ukończenia. W teorii – tak, w praktyce – nie do końca. Szesnaście etapów to bardzo mało jak na grę, której większość elementów daje się opisać w jednym paragrafie, ale cała zabawa polega na tym, że etapów trzeba się uczyć, czasami wielokrotnie próbując do nich podchodzić, starając się zapamiętać schemat poruszania przeciwników lub obmyślając taktykę przejścia. Czasami musimy to wszystko pogodzić z refleksem i sprawnością naszych palców, nie zapominając, że goni nas upływający czas. To sprawia, że ta krótka gra, może stanowić całkiem pokazną ilość dobrej zabawy. Co prawda można się przyczepić i stwierdzić, że gry można się nauczyć, ukończyć i zapomnieć, ale podobne zdanie można mieć o 80% (jak nie więcej) wydanych gier.

Co widać i słyhać?

Wbrew temu, co się może wydawać, gra wygląda schludnie. Może warto było zatrudnić kogoś lepszego grafika, u którego obiekty nie wyglądałyby rodem z gier z lat 80-tych, ale może właśnie taki był zamysł autorów – retro w każdym calu i gdzie tylko się da? Dowód na to można znaleźć choćby w stronie muzycznej – 18 utworów towarzyszących zabawie to tzw. moduły typu chip (dla każdego etapu inny). Ma to swoje plusy – dźwiękowo nie czujemy znużenia. Minusem może być to, że nie każdy preferuje ten typ muzyki. Mnie się akurat podoba, zwłaszcza że aranżacja utworów wpada w ucho. Od strony animacji muszę przyznać, że gra jest szybka, a ta szybkość właśnie jest kluczem do pokonania etapu. I nie dlatego, że chodzi tutaj o upływający czas, lecz występujące na ekranie zdarzenia (patrz paragraf wyżej). Mam jednak nieodparte wrażenie, że płynność i szybkość jest dopasowana do możliwości A1200 i na wolniejszych komputerach gra może trochę „przycinać”. Nie sprawdziłem jednak tego, więc tutaj mogę się mylić.

No to grać czy nie grać?

Odpowiadając na tak postawione pytanie, ja bym zagrał. I dlatego, że takich gier wychodzi ostatnimi czasy niewiele, i dlatego że gra nie jest kolejnym potworkiem na SDL, i dlatego że gra nie straszy wykonaniem, i dlatego że można spędzić przy niej kilka chwil (nawet na szybko próbując przebiec kilka etapów), i dlatego, że ma w sobie jakiegoś ducha retro. Wiadomo – czasy już nie te, my również nie jesteśmy mentalnie tymi samymi ludźmi i nasze wspomnienia minionego okresu zawsze będą lepsze niż to, co ktoś próbuje obecnie serwować jako retro. Niemniej jednak warto docenić czyjś trud i wysiłek. Frank Wille we współpracy z grupą Retroguru pracowali nad grą trzy miesiące. Jak mogli starali się zachować ducha „minionej epoki” i chyba trochę im się udało. Ze względu na ten właśnie aspekt i z tej właśnie perspektywy należy na grę spojrzeć. W każdym calu autorzy składają hołd retro w ekstremalnym wydaniu: muzyka chip, wymagania minimalne (kickstart 1.2, chipset OCS, 512 kB pamięci Chip, 512 kB pamięci Fast). Dodajmy do tego, że gra dostępna jest na jednej dyskietce typu NDOS, co dla niektórych może wydać się śmieszne, ale moim zdaniem jest przypieczętowaniem upamiętnienia stylu. A gdyby komuś jeszcze było mało, aby to zrozumieć, napomknę, że gra w postaci dyskietkowej miała swoją premierę podczas scenowej imprezy Demo Days, która odbyła się we wrześniu 2012 roku w Szwajcarii, gdzie zwyciężyła w „Interactive compo”. Autorzy sprzedawali ją w limitowanej edycji, na specjalnie przygotowanych dyskietkach po 2 euro za sztukę. Zarobili na tym 80 euro. I niech to będzie zwieńczeniem tego tekstu i zarazem rekomendacją recenzowanej gry.

Sebastian Rosa