AMIGA od kuchni: PAULA

Kolejnym układem specjalizowanym zamontowanym w Amidze jest Paula. Jej głównym zadaniem jest kontrola operacji we/wy. Paula obsługuje takie urządzenia jak stacja dysków, łącze szeregowe RS-232, ponadto jest wykorzystana do obsługi generacji dźwięku oraz odczytu tzw. wiosełek (paddle). Oprócz tego Paula obsługuje cały system przerwań.

Z czternastu możliwych źródeł Paula generuje sygnały przerwań dla procesora. Przerwania na poziomach 1–6 są wytwarzane na linii IPL procesora. Paula daje programiście możliwość zezwolenia lub zabronienia przerwań wszystkich źródeł.

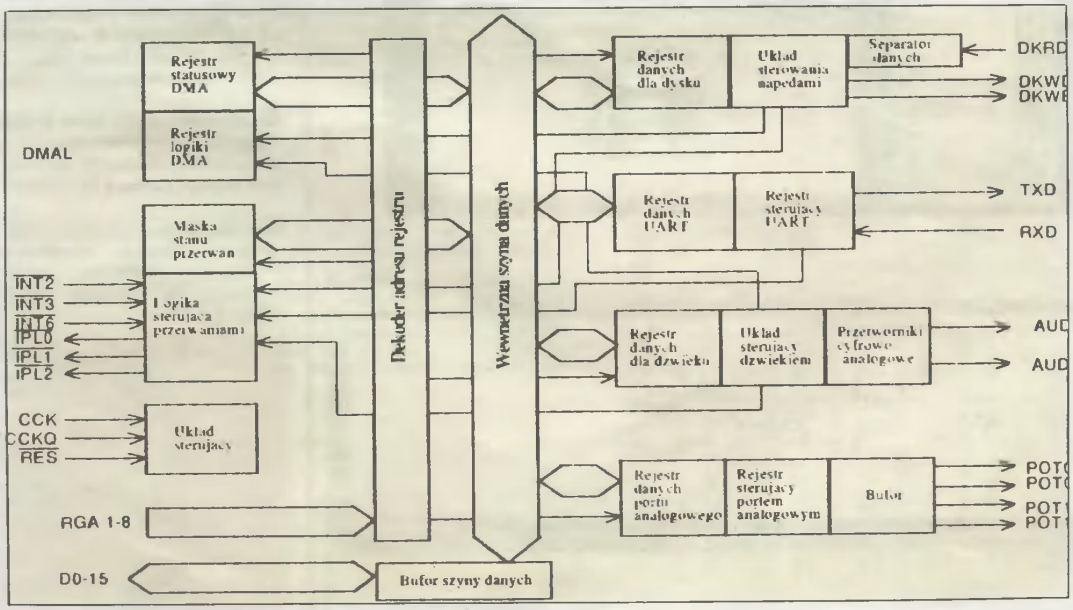
Transmisja danych do i ze stacji dysków jak również generacja dźwięku przebiegają z udziałem DMA. W czasie odczytu z dyskietki nie zawsze jest możliwe sprawdzenie czy następne słowo jest gotowe do transmisji DMA za pośrednictwem układu Agnus. Powodem tego jest zmienna prędkość odczytu. Również przy tworzeniu dźwięku Agnus nie może przewidzieć odpowiedniego momentu dla transmisji kolejnych danych. Aby uzyskać płynną transmisję DMA wyposażono Paulę w linię DMAL informującą Agnus o konieczności DMA.

Układy dla transmisji szeregowej są zawarte wewnątrz bloku o nazwie UART. Jest to skrót od angielskiej nazwy Universal Asynchronous Receive Transmit, co można przetłumaczyć jako uniwersalny, asynchroniczny nadajnik/odbiornik.

Znaczenie poszczególnych linii:

- Szyna danych D0–D15
  Tak jak w innych układach Amigi jest podłączona do szyny danych CHIP RAM.
- Szyna adresowa rejestrów RGA 1–8
  Ta szyna jest tylko wejściem. Wybiera wewnętrzny rejestr przy pomocy dekodera adresu rejestru.
- Sygnały zegarowe CCK i CCKQ
  Znaczenie identyczne jak w układzie Agnus (patrz „C&A" 06/92).
- Sygnał RES
  Sygnał ustawiający Paulę we wcześniej zdefiniowany stan początkowy.
- Sygnały AUDL i AUDR
  Wyjścia dźwięku dla kanału lewego i prawego. Do linii AUDL są podłączone kanały 0 i 3, do AUDR — 1 i 2.
- Linie TXD i RXD
  Sygnały interfejsu szeregowego RS-232. RXD jest to linia odbiornika (wejście), TXD — nadajnika (wyjście). Linie te pracują w standardzie TTL, jednak między Paulą i gniazdem łącza szeregowego umieszczono konwerter napięć do poziomu spotykanego w RS-232.
- Wejścia analogowe POT0X, POT0Y, POT1X i POT1Y
  Sygnały POT0X i POT0Y są podłączone do gniazda joysticka/myszy o numerze 0, natomiast POT1X i POT1Y — o numerze 1. Do tych gniazd możemy podłączać wcześniej wymienione wiosełka oraz joysticki analogowe, zbudowane w oparciu o potencjometry. Paula odczytuje wartość rezystancji danego potencjometru i wartość tę zapisuje w rejestrze odpowiadającym danemu potencjometrowi. Wszystkie cztery wejścia analogowe mogą być programowo ustawione jako wyjścia.
- Linie DKRD, DKWD i DKWE
  Poprzez linię DKRD Paula odczytuje dane z dysku, poprzez DKWD zapisuje, natomiast linia DKWE służy do przełączenia stacji dysków z odczytu na zapis.
- Linie przerwań INT2, INT3, INT6 oraz IPL0, IPL1, IPL2
  Trzy linie INT informują Paulę o poziomie przerwania. Linia INT2 jest normalnie podłączona do układu CIA-A. Jest także podłączona do interfejsu RS-232 oraz szyny rozszerzeń (Expansion Port). Linia INT3 jest podłączona do wyjścia układu Agnus o tej samej nazwie, natomiast linia INT6 do układu CIA-B oraz szyny rozszerzeń. Linie o pozostałych numerach są wykorzystane do obsługi urządzeń we/wy sterowanych przez Paulę i nie są wyprowadzone na zewnątrz układu. Linie IPL0—IPL2 są podłączone bezpośrednio do odpowiednich linii procesora i służą do generacji przerwania procesora na odpowiednim poziomie.

DUD



Struktura układu Paula

Rejestry układu PAULA

Poniżej przedstawiam wykaz rejestrów znajdujących się w układzie PAULA. Jeżeli chcemy obliczyć adres efektywny interesującego nas rejestru, musimy do adresu z tabeli dodać adres bazowy obszaru rejestrów, który wynosi $DFF000. Adresy w tabeli zostały podane jako liczby szesnastkowe. Wszystkie rejestry możemy podzielić na cztery typy:

R (Read) — rejestr przeznaczony tylko do odczytu.
W (Write) — rejestr przeznaczony tylko do zapisu.
S (Strobe) — wpisanie danych do rejestru spowoduje jednorazowe zadziałanie układu.
ER (Early Read) — rejestr wyjściowy DMA, dostęp do niego ma tylko kontroler DMA.

Bezpośrednio po nazwie rejestru podaję w nawiasie jego adres względny (przesunięcie od adresu bazowego) oraz typ.

DMACONR (002) R
Odczyt sterownika DMA (ma do niego dostęp również AGNUS).

DSKDATR (008) ER
Odczyt danych z dysku do RAM.

ADKCONR (010) R
Odczyt sterownika dźwięku/dysku.

POT0DAT (012) R
Odczyt potencjometru w porcie 0.

POT1DAT (014) R
Odczyt potencjometru w porcie 1.

POTGOR (016) R
Odczyt danych portu potencjometru.

SERDATR (018) R
Odczyt portu szeregowego i statusu.

DSKBYTR (01A) R
Odczyt bajtu danych dyskowych i statusu.

INTENAR (01C) R
Odczyt zezwolenia przerwania (interrupt enable).

INTREQR (01E) R
Odczyt żądania przerwania (interrupt request).

DSKLEN (024) W
Długość bloku DMA dysku.

DSKDAT (026) W
Zapis danych z RAM na dysk.

SERDAT (030) W
Zapis danych (za pośrednictwem portu szeregowego) oraz bitów stopu.

SERPER (032) W
Sterowanie portem szeregowym i prędkością transmisji.

POTGO (034) W
Zapis danych portu potencjometru i bit startu.

STRHOR (03C) S
Sygnał synchronizacji poziomej (do tego rejestru ma również dostęp układu DENISE).

DMACON (096) W
Rejestr sterujący DMA (mają do niego dostęp układy: AGNUS, PAULA, DENISE).

INTENA (09A) W
Zezwolenie przerwania.

INTREQ (09C) W
Żądanie przerwania.

ADKCON (09E) W
Sterowanie dźwiękiem, dyskiem i UART.

Opisane poniżej rejestry i ich zastosowanie są identyczne dla czterech kanałów dźwiękowych; stąd też podam jedynie adresy rejestrów przydzielonych pozostałym kanałom.

AUD0LEN (0A4) W
Długość danych dźwiękowych kanału 0. Pozostałym kanałom przydzielono rejestry: AUD1LEN (0B4) — kanał 1, AUD2LEN (0C4) — kanał 2, AUD3LEN (0D4) — kanał 3.

AUD0PER (0A6) W
Okres próbkowania kanału 0. Pozostałym kanałom przydzielono rejestry: AUD1PER (0B6) — kanał 1, AUD2PER (0C6) — kanał 2, AUD3PER (0D6) — kanał 3.

AUD0VOL (0A8) W
Głośność kanału 0. Pozostałym kanałom przydzielono rejestry: AUD1VOL (0B8) — kanał 1 AUD2VOL (0C8) — kanał 2, AUD3VOL (0D8) — kanał 3.

AUD0DAT (0AA) W
Dane dźwiękowe kanału 0 (do przetwornika C/A). Pozostałym kanałom przydzielono rejestry: AUD1DAT (0BA) — kanał 1, AUD2DAT (0CA) — kanał 2, AUD3DAT (0DA) — kanał 3.

Rejestr DMACON służy do zamykania i otwierania poszczególnych kanałów DMA. Bity w tym rejestrze mogą być albo ustawione albo wyzerowane; decyduje o tym bit 15. Jeśli jest on w stanie logicznym 1 (ustawiony), to wszystkie ustawione bity słowa danych zostaną również ustawione. Jeżeli natomiast bit 15 jest wyzerowany, to wszystkie ustawione bity w słowie danych zostaną w DMACON wyzerowane.

Bit 15 (SET/CLR) — decyduje o ustawieniu lub wyzerowaniu bitów
Bit 14 (BBUSY) — blitter zajęty (tylko odczyt).
Bit 13 (BZERO — wynik operacji blittera =0 (tylko odczyt).
Bity 12–11 — niewykorzystane.
Bit 10 (BLTPRI — DMA blittera ma pierwszeństwo przed procesorem.
Bit 9 (DMAEN — uaktywnienie ustawionych kanałów DMA (dla bitów 0–8).
Bit 8 (BPLEN) — DMA dla płaszczyzn bitowych (bit planes).
Bit 7 (COPEN) — DMA dla Coppera.
Bit 6 (BLTEN) — DMA dla Blittera.
Bit 5 (SPREN) — DMA dla duszków (sprites).
Bit 4 (DSKEN) — DMA dla dysku.
Bity 3–0 (AUDxEN) — DMA dla dźwięku dla kanału x (numer bitu odpowiada numerowi kanału)

Bit 9 (DMAEN) pełni rolę głównego przełącznika. Jeśli jest on wyzerowany, to wszystkie kanały DMA, bez względu na bity 0–8, będą nieaktywne. Jeżeli chcemy uaktywnić jakikolwiek kanał DMA, to oprócz ustawienia jego bitu (bity 0–8) musimy ustawić także bit 9 (DMAEN).

Zawartość rejestru DMACON możemy odczytać pod adresem $DFF002 (rejestr DMACONR). Stan bitu 13 i bitu 14 może poinformować nas czy blitter wykonuje jeszcze jakąś operację, oraz czy wynik wszystkich operacji blittera wyniósł 0.

Za pomocą PAULI możemy zarządzać poszczególnymi źródłami przerwań i generować sygnały przerwań dla procesora. Mamy do dyspozycji dwa rejestry: żądania przerwania (INTREQ) i maski przerwania (INTENA). Przeznaczenie bitów w obu rejestrach jest takie same. Poniżej podano kolejno: numer bitu, nazwę, adres autowektora, poziom oraz funkcję.

Bit 15 SET/CLR, —, —: Tak jak w DMACON.

Bit 14, INTEN, ($78), (6): Uaktywnienie przerwania.

Bit 13, EXTER, $78, 6: Przerwanie z CIA-B lub portu rozszerzeń.

Bit 12, DSKSYN, $74, 5: Rozpoznanie wartości synchronizacji dysku.

Bit 11, RBF, $74, 5: Bufor odbiorczy danych szeregowych zapełniony.

Bit 10, AUD3, $70, 4: Wyjście danych dźwiękowych kanału 3.

Bit 9, AUD2, $70, 4: Wyjście danych dźwiękowych kanału 2.

Bit 8, AUD1, $70, 4: Wyjście danych dźwiękowych kanału 1.

Bit 7, AUD0, $70, 4: Wyjście danych dźwiękowych kanału 0.

Bit 6, BLIT, $6C 3: Blitter gotowy.

Bit 5, VERTB, $6C, 3: Osiągnięto początek wygaszania pionowego.

Bit 4, COPER, $6C, 3: Przerwanie Coppera.

Bit 3, PORTS, $68, 2: Przerwanie z CIA-A lub portu rozszerzeń.

Bit 2, SOFT, $64, 1: Przerwania programowe.

Bit 1, DSKBLK, $64, 1: Wykonany transfer przez DMA dysku.

Bit 0, TBE, $64, 1: Bufor nadawczy danych szeregowych jest pusty.

Bit 14 (INTEN) w rejestrze INTENA pełni rolę głównego przełącznika. Przerwanie będzie wygenerowane jeśli bit INTEN i dwa odpowiednie bity w rejestrze INTENA i INTREQ zostaną ustawione. Przy ustawianiu bitów obowiązuje taka sama zasada jak w przypadku omawianego wcześniej rejestru DMACON.

Jeżeli chcemy wygenerować przerwanie wystarczy do rejestru INTREQ wpisać odpowiednią wartość (np. za pomocą rozkazu MOVE). Bit 14 w rejestrze INTREQ nie ma określonej funkcji, tak jak w INTENA, lecz jeżeli bit INTEN w INTENA jest ustawiony i ustawimy bit 14 w INTREQ, to wygenerowane zostanie przerwanie poziomu 6.

Należy pamiętać aby w rejestrze INTREQ wyzerować bit, który spowodował wytworzenie przerwania. Zawartość rejestrów INTREQ i INTENA możemy odczytać pod adresami $DFF01E (INTREQR) i $DFF01C (INTENAR).

BARTOSZ SMAGA