

6581 SOUND INTERFACE DEVICE

Na początku był SID. Potem dołączono do niego procesor. Następnie układ VIC, jeszcze trochę elektroniki i klawiaturę. Na koniec sprzęgnięto to wszystko razem i tak rozpoczęła się historia skomputeryzowanej muzyki.

Nawet twórcy kości SID nie zdawali sobie sprawy z tego, co stworzyli. To było cudowne, nie zastarzało się przez 10 lat i w dalszym ciągu odtwarza coraz to nowsze i lepsze wydźwięki.

SID jest niezależnym układem, który poza główną czynnością — generowaniem dźwięku — służy także do obsługi wiosełek (paddle) oraz zajmuje się generacją liczb pseudolosowych. Dużo można napisać na temat tego układu...

SID potrafi generować 4 rodzaje fal dźwiękowych: piłokształtną, prostokątną (z regulowanym wypełnieniem), trójkątną i szum. Pierwsze trzy fale można ze sobą mieszać. SID umożliwia także definiowanie obwiedni, głośności dźwięku, ustawianie filtrów górno- i dolnoprzepustowych oraz pasmowych, modulację wybranych oscylatorów, modulację okrężną, regulację głośności, a także przetwarzanie dźwięku z zewnątrz (zmiana głośności + filtry).

W sumie SID jest bardzo podobny do układu dźwiękowego Amigi (zbliżone możliwości). Gdyby C-64 miał nieco więcej pamięci na sample (8-bitowe), to naprawdę byłibyśmy blisko Amigi.

A teraz parę słów o rejestrach tego układu:

54272/54273 (\$D400/\$D401) FRELO1/FREHI1 SID+0/SID+1

Szesnastobitowy rejestr częstotliwości (wysokości dźwięku dla kanału pierwszego. Pozwala na uzyskanie pełnych 8 oktaf o częstotliwości od 0 Hz (\$0000) do około 4000 Hz (\$FFFF). Aby przeliczyć wartość rejestru na częstotliwość dźwięku, można skorzystać ze wzoru:

$$\text{Częstotliwość} = ((\text{Zawartość rejestru}) * 985250 / 16777216) \text{ Hz}$$

Dla komputerów pracujących w systemie NTSC (U.S.A.) wartość 985259 należy zastąpić liczbą 1022730.

54274/54275 (\$D402/\$D403) PWLO1/PWHI1 SID+2/SID+3

Dwunastobitowy rejestr szerokości impulsu fali prostokątnej dla kanału pierwszego (4 najstarsze bity rejestru PWHI1 są nie wykorzystane). Wypełnienie fali prostokątnej można wyliczyć ze wzoru: Szerokość impulsu = ((Zawartość rejestru)/40.95) % Szerokość impulsu można regulować od 0% (0) do 100% (4095).

54276 (\$D404) VCREG1 SID+4

Rejestr kontrolny. Znaczenie bitów:

bit 0:1 = rozpoczęcie fazy narastania/opadania/wybrzmiewania dźwięku dla kanału

pierwszego (muszą być ustawione rejestry ATDCY1/SUREL1, FRELO1/FREHI1 i wybrany kształt fali)

0 = rozpoczęcie fazy wyniku dźwięku dla kanału pierwszego

bit 1:1 = synchronizacja częstotliwości oscylatora kanału pierwszego z oscylatorem kanału trzeciego. Oscylator 3 musi być włączony (FRELO3/FREHI3), inne parametry kanału trzeciego nie wpływają na dźwięk

0 = synchronizacja wyłączona

bit 2:1 = modulacja dookreżna oscylatora 1 i 3. Można dzięki temu uzyskać efekty zbliżone do gongu

bit 3:1 = wyłącza kanał pierwszy

bit 4:1 = włącza falę trójkątną na kanale pierwszym

bit 5:1 = włącza falę piłokształtną na kanale pierwszym

bit 6:1 = włącza falę prostokątną na kanale pierwszym (PWLO1/PWHI1 musi być ustawiony)

bit 7:1 = włącza szum na kanale pierwszym.

Możliwe jest łączenie ze sobą fal piłokształtnej, trójkątnej i prostokątnej (logiczne AND kształtu fal). Nie ma sensu łączyć szumu z inną falą. Fala prostokątna to po prostu na przemian wartości 0 i 255, szybkość zmian zależy od częstotliwości dźwięku i szerokości impulsu. Fala piłokształtna obejmuje wartości od 0 do 255 z krokiem 1. Fala trójkątna to wartości od 0 do 255 z krokiem 1 i od 255 do 0 z krokiem -1. Szum jest zbiorem wartości losowych. Program 1 obrazuje przebieg fali trójkątnej.

54277 (\$D405) ATDCY1 SID+5

Rejestry faz narastania/opadania dźwięku dla kanału pierwszego.

bity 0-3: opadanie (0-15);
bity 4-7: narastanie (0-15);

Wartość rejestru narastania określa czas, w którym głośność będzie podnosić się od 0 do szczytu amplitudy głośności. Poniższe wartości określają czas:

0 — 2 milisekundy
1 — 4 milisekundy
2 — 16 milisekund
3 — 24 milisekund
4 — 38 milisekund
5 — 56 milisekund
6 — 68 milisekund
7 — 80 milisekund

8 — 100 milisekund
9 — 250 milisekund
10 — 500 milisekund
11 — 800 milisekund
12 — 1 sekunda
13 — 3 sekundy
14 — 5 sekund
15 — 8 sekund

Czasy dla fazy opadania:

0 — 6 milisekund
1 — 24 milisekundy
2 — 48 milisekund
3 — 72 milisekundy
4 — 114 milisekund
5 — 168 milisekund
6 — 204 milisekundy
7 — 240 milisekund
8 — 300 milisekund
9 — 750 milisekund
10 — 1.5 sekundy
11 — 2.4 sekundy
12 — 3 sekundy
13 — 9 sekund
14 — 15 sekund
15 — 24 sekundy

54278 (\$D406) SUREL1 SID+6

Rejestr wybrzmiewania/zaniku dźwięku dla kanału pierwszego.

bity 0-3: zanik dźwięku (0-15)
bity 4-7: wybrzmiewanie dźwięku (0-15)

Dla wybrzmiewania dźwięku obowiązuje tabela podana dla fazy narastania. Analogicznie, dla opadania stosuje się tabelę dla fazy zaniku. Program 2 pokazuje cykl narastania, opadania, wybrzmiewania i zaniku dźwięku.

54279/54280 (\$D407/\$D408) FRELO2/FREHI2 SID+7/SID+8

Częstotliwość dźwięku dla kanału drugiego.

54281/54282 (\$D407/\$D40A) PWLO2/PWHI2 SID+9/SID+10

Szerokość impulsu fali prostokątnej dla kanału drugiego.

54283 (\$D40B) VCREG2 SID+11

Rejestr kontrolny. Bity u,3,4,5,6,7 mają takie same funkcje jak w VCREG1, ale w odniesieniu do kanału drugiego.

bit 1:1 = synchronizacja częstotliwości oscylatora kanału drugiego i pierwszego
bit 2:1 = modulacja dookreżna oscylatorów 2 i 1

54284 (\$D40C) ATDCY2 SID+12

Rejestr narastania i opadania dźwięku dla kanału drugiego.

54285 (\$D40D) SUREL2 SID+13

Rejestr wybrzmiewania i zaniku dźwięku dla kanału drugiego.

54286/54287 (\$D40E/\$D40F) FRELO3/FREH3 SID+14/SID+15

Częstotliwość dźwięku dla kanału trzeciego.

54288/54289 (\$D410/\$D411) PWLO3/PWHI3 SID+16/SID+17

Szerokość pulsu fali prostokątnej dla kanału trzeciego.

54290 (\$D412) VCREG3 SID+18

Rejestr kontrolny. Bity 0,3,4,5,6,7 — mają takie samo znaczenie, jak w VCREG1 i VCREG2, ale dla kanału trzeciego.

- bit 1:1 = synchronizacja częstotliwości oscylatora trzeciego = drugim
- bit 2:1 = modulacja dookreślna oscylatorów 3 i 2.

54291 (\$D413) ATDCY3 SID+19

Rejestr narastania i opadania dźwięku dla kanału trzeciego.

54292 (\$D414) SUREL3 SID+20

Rejestr wybrzmiewania i zaniku dźwięku dla kanału trzeciego.

54293/54294 (\$D415/\$D416) CUTLO/CUTHI SID+21/SID+22

Jedenastobitowy rejestr częstotliwości granicznej filtrów (dolnoprzepustowy, pasmowy i górnoprzepustowy). W rejestrze CUTLO wykorzystane są tylko 3 najmłodsze bity (0-2), pozostałe są nie wykorzystane. Częstotliwość graniczna filtru może być określona w zakresie od 30 Hz do około 12 000 Hz. Można ją wyliczyć z następującego wzoru:

Częstotliwość graniczna = (Zawartość rejestru * 5.8) Hz+30Hz

Filtr dolnoprzepustowy tłumi częstotliwości powyżej określonego przez CUTLO/CUTHI poziomu. Filtr górnoprzepustowy działa na odwrót. Filtr pasmowy redukuje głośność częstotliwości dźwięku poniżej i powyżej zadanej częstotliwości granicznej. Wszystkie trzy rodzaje filtrów można ze sobą łączyć — co prowadzi często do drastycznych zmian w porównaniu z dźwiękiem niefiltrowanym. Filtry dolnoprzepustowy i górnoprzepustowy działają z dynamiką 12 db na oktawę, filtr pasmowy — 6 db na oktawę.

54295 (\$D417) RESON SID+23

Rejestr kontrolny filtrowania.

- bit 0:1 = filtrowanie kanału 1
- bit 1:1 = filtrowanie kanału 2
- bit 2:1 = filtrowanie kanału 3
- bit 3:1 = filtrowanie sygnału przychodzącego z zewnątrz;

Jak wiadomo, możliwa jest obróbka dźwięku z zewnątrz (wprowadzonego przez linię AUDIO IN — nóżka 5 gniazda AUDIO/VIDEO). Można go poddać filtracji i/lub zmienić jego głośność. Poddany obróbce sygnał zostaje następnie wyprowadzony na nóżkę 3 gniazda AUDIO/VIDEO (AUDIO OUT). Niestety, nie jest to przetwarzanie analogowo-cyfrowe dźwięku, więc tego sposobu nie da się wykorzystać do digitalizacji dźwięku.

Uwaga! Nie polecam zabawy z przetwarzaniem dźwięku z zewnątrz — SID jest bardzo delikatnym układem

bity 4-7: charakterystyka filtrów (0-15);

Bity te pozwalają na „podbicie” częstotliwości leżących blisko częstotliwości granicznej filtru (0-min., 15-max.).

54296 (\$D418) SIGVOL SID+24

Rejestr głośności i doboru filtrów dla wszystkich głosów

- bity 0-3: wybór głośności (0-min, 15-max)
- bit 4:1 = filtr dolnoprzepustowy włączony;
- bit 5:1 = filtr pasmowy włączony;
- bit 6:1 = filtr górnoprzepustowy włączony;
- bit 7:1 = kanał trzeci wyłączony;

Bit 7 jest przydatny wtedy, gdy korzystasz z kanału trzeciego do modulowania częstotliwości innych kanałów lub do generacji liczb pseudolosowych

LISTING 1

```

10 POKE 54286,1:POKE 54287,0
20 POKE 54290,2^4+1
30 PRINT PEEK(54299)
40 GOTO 30

```

LISTING 2

```

200 POKE 54280,111:POKE54287,0
210 POKE 54291,15+15*16
220 POKE 54292,15+15*16
230 POKE 54290,1+2^4
240 PRINT "NARASTANIE/OPADANIE
/WYBRZMIEWANIE"
250 FOR A=0 TO 255
260 :PRINT PEEK(54300)
270 NEXT A
280 POKE 54290,0+2^4
290 PRINT "ZANIK"
300 FOR A=0 TO 255
310 :PRINT PEEK(54300)
320 NEXT A

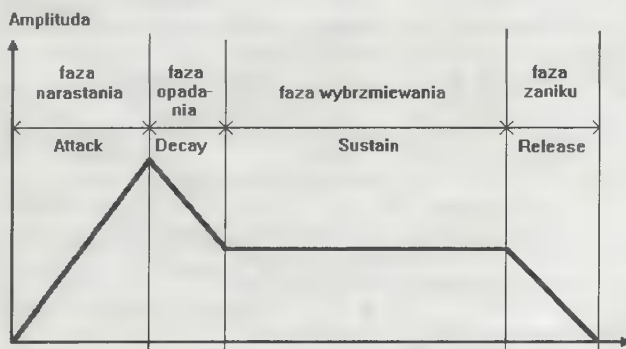
```

LISTING 3

```

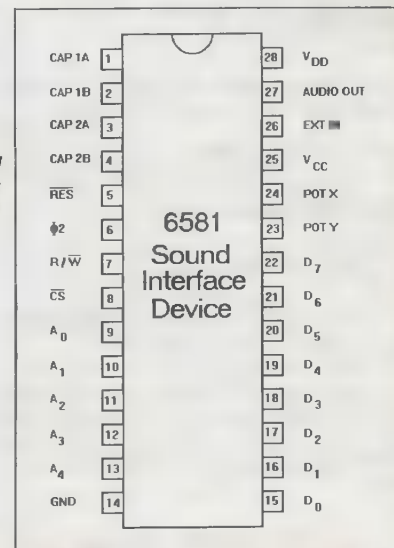
10 FOR A=54272 TO 54296
20 :POKE A,111
30 :FOR B=1 TO 20:NEXT B
40 :PRINT PEEK(A)
50 NEXT A

```



Rys. 2. Fazy powstawania dźwięku (ADSR)

Rys. 1. Rozkład wyprowadzeń układu 6581



(rejestr RANDOM), czy podobnych celów. Ustawiając bit 7 (stan logiczny 1) eliminujesz wszystkie zbędne efekty dźwiękowe.

Do wszystkich rejestrów opisanych powyżej dane możesz jedynie wpisać. Próba odczytu ich zawartości da w wyniku 0, chyba że odczyt wykona się wystarczająco szybko, natychmiast po wpisaniu danych do rejestrów. Ciekawy efekt daje program 3 — do wszystkich powyższych rejestrów wpisywana jest

wartość 11,1, ale wartości odczytane z nich są różne. Proponuję poeksperymentować z długością pętli B.

**54297/54298 (\$D419/\$D41A) POTX/POTY
SID+25/SID+26**

Rejestry wiosełek (paddle). Służą do odczytu położenia wiosełek. Do C-64 można podłączyć 4 takie potencjometry. Rejestr POTX odczytuje stan 1 lub 3 wiosełka, POTY odczytuje wiosełko 2 lub 4. Ich odczyt następuje na bardzo podobnych zasadach jak odczyt klawiatury — za pośrednictwem komórki 56320 (\$DCOO) czyli CIA#1. Ze względu na małą popularność wiosełek pominięto dokładniejszy opis rejestrów POTX i POTY. Standardowo w POTX/POTY znajduje się wartość 255.

54299 (\$D41B) RANDOM SID+27

Odczyt oscylatora 3, generator liczb pseudolosowych. Trzeci z kolei rejestr tylko do odczytu. Wartości, które się w nim pojawiają, to górne 8 bitów fali oscylatora 3, czyli po prostu kształt fali. Szybkość zmiany zależy od wysokości dźwięku kanału 3 (FRELO3/FREHI3). Program 1 odczytuje kształt fali trójkątnej. Ciekawe efekty daje połączenie dwóch fal (logiczne AND kształtu). Zastosowań tego rejestru jest wiele np. modulacja innych kanałów, efekt vibrato, dynamiczna zmiana częstotliwości dźwięku i filtrów, generacja liczb pseudolosowych, i tak dalej.

54300 (\$D41C) ENV3 SID+28

Głośność kanału 3. Ostatni z 4 rejestrów tylko do odczytu. Obrazuje kształt narastania opadania wybrzmiewania i zaniku fali oscylatora 3. Zastosowania — takie same, jak dla RANDOM.

**54301—54303 (\$D41D/\$D41E/
/\$D41F) SID+29/SID+30/
/SID+31**

Rejestry nie wykorzystane. Zgodnie z logiką binarną powinny ich być 32, stąd też trzy komórki o adresach 54301, 54302 i 54303 zawsze zawierają wartość 255 (nie są podłączone do układu). Wpisywanie do nich czegokolwiek nie da żadnego efektu.

54304—55295 (\$D420—\$D7FF)

W pamięci dla układu SID przydzielono 1024 bajtów. Jednakże SID korzysta jedynie z 32 bajtów (a ściślej z 29). Pozostały obszar to lustrzane odbicia tych pierwszych 32 rejestrów. Tak więc komórka 54272 zawiera dokładnie to samo, co 54272+32*k, gdzie k = 1,2,...,31, i tak dalej.

Ot i cały SID. Z tego układu można naprawić wycisnąć wiele i gdyby C-64 miał 0.5 MB RAM i nieco szybszy zegar, to mielibyśmy obecnie TRZY-KANAŁOWĄ wersję programu PROTRACKER 1.1B (Amiga).

BARTŁOMIEJ DRAMCZYK

ZEGAR TOD (Time of Day)

Przesyłam Wam program, który po uruchomieniu spowoduje pojawienie się w prawym górnym rogu ekranu zegara, pokazującego z dokładnością do dziesiętnej części sekundy. Program wykorzystuje rejestry wewnętrznego zegara komputera umieszczonego w układzie CIA 1 (time of day). Zegar działa niezależnie od wewnętrznego systemu przerwan IRQ (są tu one wykorzystane do wyświetlania cyfr czasu na ekranie) oraz niezależnie od czynności wykonywanej aktualnie przez komputer.

Również transmisją danych z magnetofonem bądź stacją dysków nie spowoduje zatrzymania tego zegara, odróżnieniu od programowego zegara systemu operacyjnego dostępnego za pomocą instrukcji PRINT TI lub PRINT TIS. Oznacza to, że możemy w tym samym czasie wpisywać, uruchamiać, poprawiać czy korzystać z dowolnego programu (pod warunkiem pozostawienia bez zmian wektora przerwan IRQ w komórkach 788—789); przez cały czas będziemy mieli jednocześnie podawany aktualny czas.

Po wczytaniu programu i sprawdzeniu sumy kontrolnej (zmienna SK) program zapyta o aktualny czas. Ze względu na konstrukcję rejestrów TIME OF DAY, do zapisu godzin stosowana jest notacja angielska, tzn. liczby z zakresu 0—11 oraz litery PM i AM określające odpowiednio czas PO lub PRZED przed południem. Gdy wciśniemy klawisz RETURN po podaniu liczby minut, zegar rozpocznie pracę, w najwyższej linii ekranu po prawej stronie pojawi się wskazanie zegara.

Po wciśnięciu klawiszy RUN/STOP i RESTORE zegar znika z ekranu. Nie znaczy to jednak, że przestał on działać. W każdej chwili możesz spowodować jego ponowne pojawienie się za pomocą instrukcji SYS 49333. Możliwe jest również wpisanie nowego czasu w postaci:

**POKE 49316,6:REM:GODZINA 0600
POKE 49318,4:REM:MINUT 01**

W analogiczny sposób od komórki 49230 wpisać możemy liczbę sekund, jeśli ma być ona różna od zera. W przypadku gdy chodzi o czas popołudniowy, do zmiennej GG należy dodać 128. Po wykonaniu powyższych operacji uruchamiamy program instrukcją SYS 49299.

J.W. DOMAŃSKI

```
20 rem * commodore
30 rem * time-of-day
40 rem * j.w.domanski
60 :
110 for i=49214 to 49345
120 read q:sk=sk+q:poke i,q
130 next i
140 if sk < > 15004 then print "blad w danych":end
150 print "tod data ok."
160 input "podaj godzine (0-11):"g
170 input "przed czy po poludniu (am/pm):"a$
180 input "podaj minute (0-59):"m
190 gg=int(g/10)*16+g-int(g/10)*10
200 mm=int(m/10)*16+m-int(m/10)*10
210 if a$="p" then gg=gg+128
```

```
poke 49316,gg:poke 49318,mm:sys 49299
49214 data 162,000,160,003,185,008,220,141,194,192,041
49225 data 112,074,074,074,074,032,137,192,173,194,192
49236 data 041,015,032,137,192,169,010,032,137,192,136
49247 data 208,225,173,011,220,141,194,192,173,008,220
49258 data 041,015,032,137,192,169,240,032,137,192,169
49269 data 209,044,194,192,016,003,024,105,015,032,137
49280 data 192,169,221,032,137,192,076,049,234,024,105
49291 data 048,009,128,157,027,004,232,096,173,014,220
49302 data 009,128,141,014,220,173,015,220,041,127,141
49313 data 015,220,162,000,160,000,169,000,142,011,220
49324 data 140,010,220,141,009,220,141,008,220,162,062
49335 data 160,192,120,142,020,003,140,021,003,088,096
```

EDIT dla C-64

Program ten dostarcza kilku ciekawych funkcji, pomocnych przy pisaniu programów. Każde działanie uruchamia się kombinacją klawiszy CTRL i klawisza literowego. Podręczny opis funkcji uzyskać można na ekranie w postaci „okna” po naciśnięciu klawisza F7. „Okno” z informacją zniknie odtwarzając poprzednią zawartość ekranu po naciśnięciu klawisza C=.

Program zmienia wektor IRQ. Jeżeli użyjemy kombinacji klawiszy RUN/STOP i RESTORE, możemy odtworzyć działanie programu używając nowo-wprowadzonego po uruchomieniu programu zlecenia „@”. Program zmniejsza ilość dostępnej pamięci dla programów BASIC. Jeżeli w komputerze działają już jakieś programy zmieniające wektor IRQ, np. 64 CLOCK, EDIT wpisuje się „nad” ten program nie zmieniając jego działania (co nie zawsze niestety jest możliwe — przyp. redakcji). Jednakże EDIT 64 musi być uruchomiony dopiero po tym programie. Poniżej przedstawiam nowe instrukcje:

- R** usunięcie znaków w linii programowej do końca w prawo
- L** usunięcie znaków w linii programowej do końca w lewo
 - usunięcie znaków w linii w prawo do separatora " : "
 - usunięcie znaków w linii w lewo do separatora " : "
- SPACE** usunięcie wszystkich znaków w linii

- K** usunięcie wszystkich znaków między separatorami " : "
- A** ustawia ekran nr 1 normalny (początek od \$0400)
- D** ustawia ekran nr 2 (początek od \$8C00); po pierwszym włączeniu trzeba go wyczyścić. Dwa ekrany łatwo przełączalne można wykorzystywać jako okna jedno do edycji, drugie do uruchamiania programu (jak w Turbo BASIC na IBM).
- X** skok kursora do środka linii
- Y** skok kursora do środkowej linii na ekranie
- Z** skok kursora do lewego dolnego rogu ekranu
- P** skok na koniec linii programu
- J** włącza/wyłącza inwersję linii programu dla uwidocznienia interesującego miejsca
- U** zamiana miejsca znaku pod kursorem ze znakiem na prawo od kursora
- C** zamiana kolorów ekranu i ramki na czarno, tekst na biało
- O** włącza/wyłącza tryb wstawiania INSERT. Można też nacisnąć RETURN, by powrócić do normalnego trybu wpisywania liter
- @** usunięcie zbędnych spacji zawartych w linii programu. Nie usuwa spacji zawartych pod instrukcją wydruku
- B** włącza i wyłącza tryb powtarzania znaków przy wpisywaniu.

JAROSŁAW SĄKOL