GRAJĄCY PECET

Szczęśliwi (albo i nie – to zależy od punktu widzenia) posiadacze komputerów zgodnych ze standardem IBM PC wiedzą, że wbudowane w komputer mechanizmy wydawania dźwięku są więcej niż ubogie. Jeden mały głośniczek sterowany dwupoziomowo (czyli sygnałem prostokątnym) jest dobry do wydawania dźwięków określanych mianem „pipnięcia", ale nie do tworzenia przyzwoitej oprawy dźwiękowej. Nic więc dziwnego, że skonfrontowani z właścicielami innych komputerów (np. Amiga...) posiadacze PC-tów niewiele mają do powiedzenia. I nawet tłumaczenia, że jest to komputer przeznaczony do pracy, a nie do zabawy, nie mogą zachwiać wewnętrznego przekonania, że czasem by można posłuchać dobrej muzyki z komputera... lub choćby, żeby to bezduszne bydlę przemówiło do właściciela jakim takim głosem.

Oczywiście – można kupić specjalizowaną kartę i zacząć w wolnych chwilach bawić się w tworzenie komputerowego świata dźwięków, tyle że cena takiej zabawki stanowi dosyć znaczący procent ceny całego komputera.

Na rynku programów shareware i public domain dostępne są programy służące do generacji dźwięku przez komputer. Można je zdobyć poprzez tzw. BBS-y (zainteresowanym polecamy zapoznanie się z katalogami hurtowni programów shareware i public domain – przypominamy jednak, że za programy shareware trzeba płacić ich autorom!). W dołączonej do programu dokumentacji (w postaci jednego z plików programu) można znaleźć dokładny opis obsługi programu i przykładowe pliki danych, zawierających np. różne utwory muzyczne.

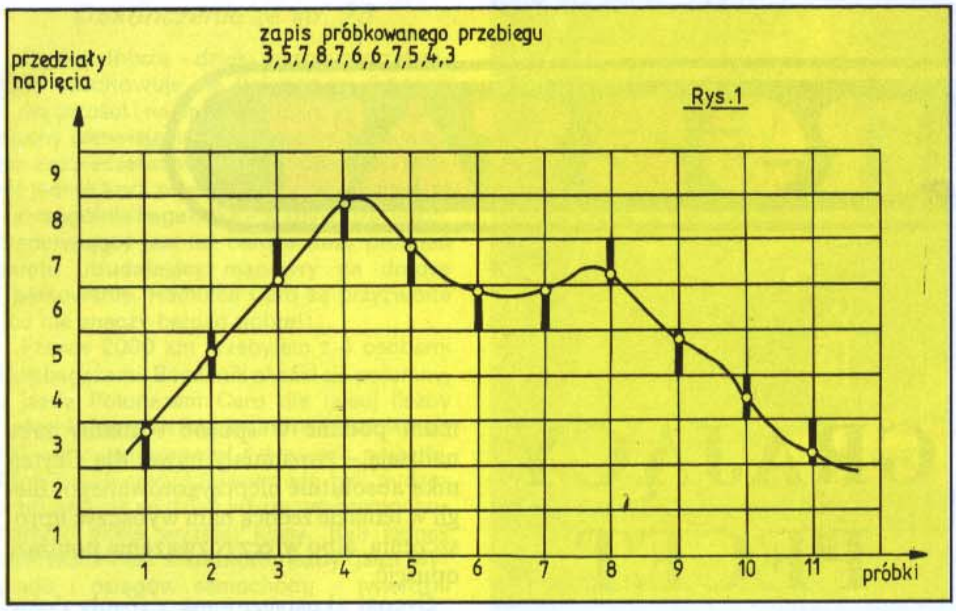
Programy te mogą pracować wykorzystując jako źródło dźwięku także i wewnętrzny głośniczek komputera, ale efekty – powiedzmy to szczerze – są opłakane. O wiele lepiej jest używać zewnętrznego wzmacniacza i porządnych kolumn głośnikowych. Tylko jak podłączyć wzmacniacz do komputera?

W niniejszym artykule opisany jest pierwszy krok w tym kierunku – prosta przystawka stanowiąca pośrednik między komputerem i zewnętrznym wzmacniaczem akustycznym. Urządzenia takie są dostępne na rozmaitych giełdach, gdzie noszą dumną (może zbyt dumną w stosunku do możliwości) nazwę Covox. W rzeczywistości jest to prosty przetwornik cyfrowo-analogowy pobierający dane wejściowe ze złącza drukarki. Proponowane rozwiązanie charakteryzuje się prostą konstrukcją i niską ceną, za co – niestety – przyszło zapłacić nie najlepszą jakością przetwarzania (jakość ta jest jednak o wiele wyższa niż osiągana z wbudowanego w komputer głośniczka).

Aby móc zrozumieć działanie Covoxa niezbędne jest trochę teorii sygnałów. Poniższy tekst zawiera absolutne minimum podane w sposób – mamy taką nadzieję – zrozumiały nawet dla Czytelnika absolutnie nieprzygotowanego. Biegli w temacie zechcą nam wybaczyć uproszczenia, albo wręcz rozważania poniższe opuścić.

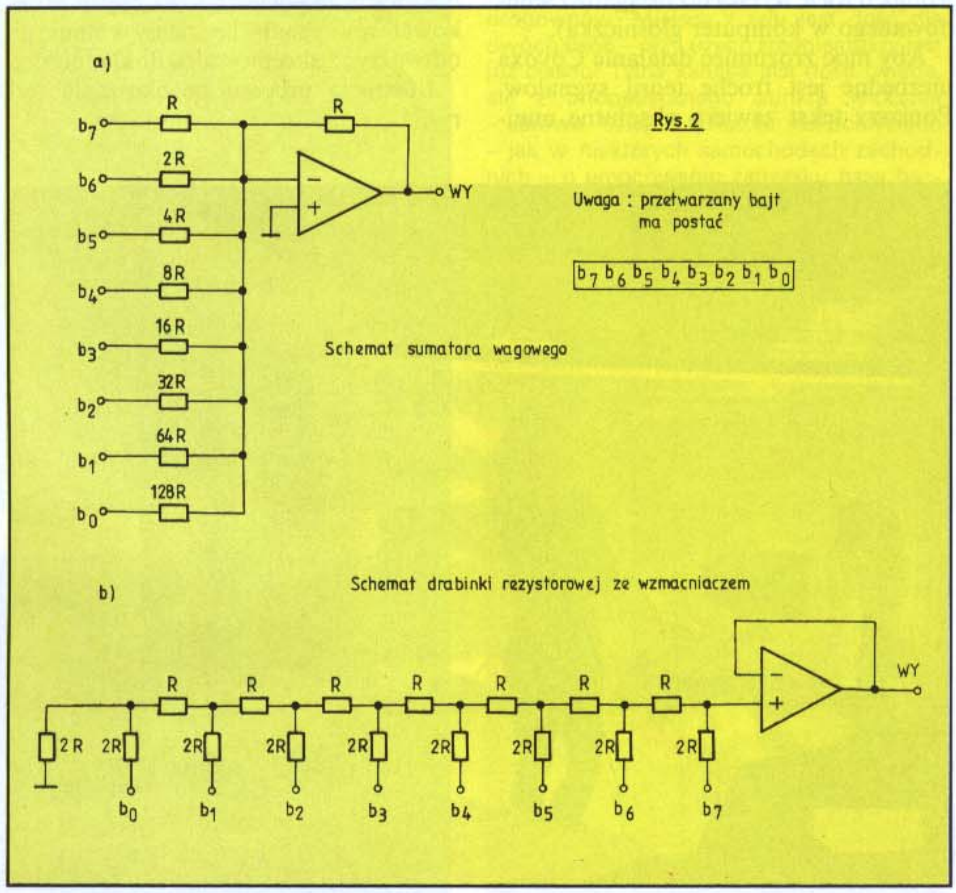
Sygnał akustyczny ma z reguły dosyć nieregularny, poszarpany kształt. Jest on daleki od przebiegu sinusoidalnego, który łatwo można opisać częstotliwością (czyli czasem, w którym przebieg się okresowo powtarza) i amplitudą (czyli maksymalną wartością przebiegu). Jedną z metod opisania przebiegu nieregularnego jest tzw. próbkowanie. W najprostszym przypadku po prostu co pewien (pewien, ale zawsze ten sam!) czas patrzymy, jaką wartość ma sygnał (jest to tzw. wartość chwilowa sygnału) i zapamiętujemy kolejne wartości próbek. Proces ten nazywa się naukowo próbkowaniem. Jeżeli będziemy to robić odpowiednio często, to – wiedząc, jak często próbkowaliśmy sygnał – będziemy w stanie go odtworzyć z akceptowalną dokładnością.

Ilustracją procesu próbkowania jest rys. 1.



W tym miejscu warto się chwilę zastanowić nad doborem dwóch wartości. Po pierwsze – jaka jest najniższa częstotliwość próbkowania, przy której można jeszcze sygnał odtworzyć (czyli mówiąc bardziej przejrzyście: chodzi o to, aby odstęp pomiędzy poszczególnymi próbkami był jak największy, ze względu na to, że im wolniej próbkujemy, tym łatwiej jest taki proces zrealizować, ale nie może on jednocześnie być za długi, bo nie będziemy w stanie odtworzyć kształtu sygnału).

Będący za pan brat z akustyką wiedzą, że sygnał akustyczny zajmuje pewne pasmo, tzn. jest sumą przebiegów sinusoidalnych o rozmaitych częstotliwościach (harmonicznych). Jaki jest wpływ poszczególnych harmonicznych na brzmienie sygnału możemy się przekonać zmieniając nastawy regulatorów barwy dźwięku we wzmacniaczu akustycznym – polega to właśnie na zmianie udziału poszczególnych harmonicznych w sygnale. Otóż okazuje się, że aby móc odtworzyć sygnał, wystarczy próbkować dwa razy częściej, niż wynosi częstotliwość największej harmonicznej sygnału (jest to tzw. twierdzenie Shannona).



Drugą wartością wymagającą doboru jest dokładność, z jaką próbkujemy wartość chwilową. Problem jest analogiczny do doboru częstotliwości próbkowania – im dokładniej mierzymy, tym dokładniej jesteśmy w stanie odtworzyć sygnał, ale tym trudniejsza (droższa) jest realizacja praktyczna tego przedsięwzięcia. W praktyce sprawdza się, czy wartość chwilowa przebiegu mieści się w pewnych granicach (w pewnym przedziale) i zapamiętuje się numer tego przedziału.

Tak więc zapamiętany przebieg akustyczny będzie po prostu ciągiem liczb – no, może niezupełnie, dla pełnej informacji musimy jeszcze znać czas próbkowania, czyli odstęp między kolejnymi próbkami, i maksymalną wartość, którą mógł osiągnąć sygnał podczas próbkowania. (Zwróćmy uwagę, że w prawidłowo zarejestrowanym sygnale wartości próbek nigdy nie mogą być większe od tej drugiej wartości).

Mamy więc zapisany sygnał w postaci ciągu liczb – pozostaje go teraz przetworzyć znowu na sygnał akustyczny. Tym w zasadzie zajmie się głośnik, nam pozostaje zamiana go na sygnał elektryczny, czyli przebieg o zmiennym w czasie napięciu.

W celu uściślenia problemu przypomnijmy, jak przechowywane są dane liczbowe wewnątrz komputera.

Abstrahując od szczegółów technicznych każda liczba w komputerze przedstawiana jest w postaci ciągu bitów, czyli elementów składowych mogących przyjmować tylko dwie wartości – nazwane umownie zerem logicznym (0) i jedynką logiczną (1). Przyjęło się, że podstawowym ciągiem jest ciąg o długości ośmiu bitów, czyli bajt.

Krótki test z kombinatoryki: jeżeli mamy osiem pudełek, a w każdym z pudełek może (lub nie) być kulka, to na ile sposobów możemy rozłożyć kulki w pudełkach? Oczywiście – wariacje z powtórzeniami, znany wzór $2^8 = 256$. Czyli, przez analogię, posiłkując się jednym bajtem możemy wyrazić 256 różnych wartości, a mówiąc inaczej – zapisać w nim informację o przynależności wartości chwilowej sygnału do jednego z 256 przedziałów.

Wartość dziesiętną takiej liczby binarnej (tak ją nazywamy, bo opiera się ona na systemie dwójkowym) wyznacza się następująco:
jeżeli mamy liczbę binarną o postaci
$b_7b_6b_5b_4b_3b_2b_1b_0$
gdzie b_i jest bitem, czyli może przyjmować wartości '0' i '1',

to wartość dziesiętna tej liczby może być wyliczona ze wzoru

$2^7 \cdot b_7 + 2^6 \cdot b_6 + 2^5 \cdot b_5 + 2^4 \cdot b_4 + 2^3 \cdot b_3 + + 2^2 \cdot b_2 + 2^1 \cdot b_1 + 2^0 \cdot b_0$

Przykładowo, binarna liczba 00010001 to dziesiętne 17, a binarne 11111111 to dziesiętne 255.

Tak więc sposób kodowania już mamy – jak teraz jednak odwzorować go w wartości napięcia?

Zastanówmy się chwilę, jakimi własnościami powinien odznaczać się układ zdolny do odtworzenia wartości chwilowej sygnału na podstawie próbki – powiedzmy – zapisanej w pojedynczym bajcie? Oczywiście powinien mieć osiem wejść dwustanowych (tzn. takich, które dopuszczają tylko dwa poziomy sygnału odpowiadające logicznym '0' i '1') i wyjście, na którym powinno pojawiać się napięcie. Funkcja uzależniająca napięcie wyjściowe od stanu poszczególnych wejść (transmitancja) musi mieć postać taką, jak podany wyżej wzór przeliczeniowy.

Układ taki nazywa się przetwornikiem cyfrowo-analogowym, a zrealizować go można w układach (na przykład) sumatora wagowego lub drabinki rezystorowej.

Sumator wagowy (rys. 2a) jest układem, który na swoim wyjściu wytwarza sygnał będący sumą sygnałów wejściowych pomnożonych przez współczynniki (wartości tych współczynników zależą od wartości elementów w układzie). Jeżeli dobierzemy te współczynniki tak, że będą one kolejnymi potęgami dwójki, to otrzymamy dokładnie to, o co nam chodziło. A ponieważ, jak powiedzieliśmy, wartości współczynników zależą od wartości elementów, tak więc realizacja układu staje się bardzo prosta.

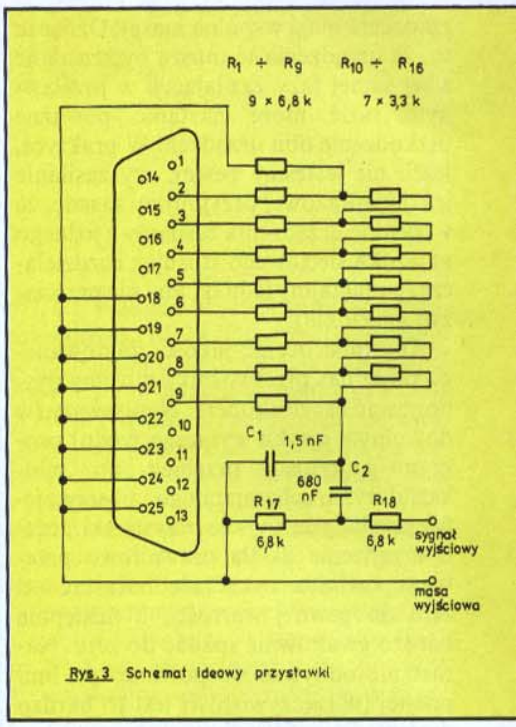
Drabinka rezystorowa (zwana drabinką R/2R – rys. 2b) jest bardzo sprytnie pomyślaną siecią rezystorów, której zasada działania (w uproszczeniu) polega na tym, że zastępcza rezystancja układu widziana z pozycji węzłów (czyli miejsc, gdzie łączą się rezystory o wartościach R i 2R) jest stała, i wynosi 2R. Dlatego drabinka taka może być rozpatrywana jako zespół dzielników we współczynniku podziału 2, połączonych kaskadowo (dwa rezystory o wartości 2R każdy, z których pierwszy łączy wejście z węzłem, a drugi – to zastępcza rezystancja drabinki do masy z pozycji węzła wynosząca też 2R). Jeżeli teraz na np. trzecie z kolei wejście drabinki podamy sygnał – logiczne '1' (a przecież logiczne '1' to też napięcie!), to na skutek działania kaskadowo połączonych dzielników do wyjścia dotrze napięcie logicznej '1' podzielone przez 2, i jeszcze raz przez 2, i jeszcze raz przez 2, czyli w sumie przez 23. A ponieważ analogicznie będzie z innymi wejściami, więc teraz wyraźnie już widać, że układ realizuje to, o co nam chodziło.

Zauważmy tu jeden „haczyk": napięcie logicznej '1' jest tu traktowane jako sygnał napięciowy o pewnej (stałej) wartości, a napięcie logicznego '0' – jako napięcie o wartości 0 woltów, czyli zwarcie z masą. A ponieważ napięcia poziomów logicznych są określane z pewną tolerancją, więc dla dokładnego działania układu niezbędne byłoby zastosowanie układów formujących, przetwarzających wartość logiczną ('0' lub '1') w ściśle określone wartości napięcia.

Przystawkę zrealizować można zarówno jako sumator wagowy, jak i drabinki R/2R. Zanim jednak rozpatrzymy za i przeciw każdego z wariantów zastanówmy się, którędy wyprowadzany jest sygnał cyfrowy.

Jako standard przyjęto, że ciąg kolejnych próbek przebiegu wyprowadzany jest na złącze drukarki. Wyjście to widziane jest przez komputer jako pewna lokacja adresowa w polu adresowym urządzeń peryferyjnych, dla nas nie jest to jednak ważne – tym zajmie się mikroprocesor i program. Standard ten nakłada jednak na nas pewne poważne ograniczenie – mamy dostęp jedynie do ośmiu końcówek poszczególnych bitów i masy sygnałowej, nie dysponujemy za to zasilaniem (drukarka ma własne zasilanie i nie było potrzeby wyprowadzania go na złącze). Dlatego też, jeżeli chcemy uniknąć dodatkowego zasilacza, musimy skonstruować tzw. urządzenie bierne.

W proponowanym rozwiązaniu zdecydowano się na wykorzystanie układu drabinki R/2R. Co prawda zawiera ona dwa razy więcej rezystorów niż układ sumatora wagowego, ale rezystory te mają tylko dwie wartości, i w ten sposób uproszczony został problem doboru elementów. Wykorzystano standardowe wartości elementów o tolerancji 5%, dla których stosunek wartości 2:1 nie jest dokładnie zachowany, ale i tak otrzymane wyniki były bardzo dobre. Oczywiście jeżeli istnieje możliwość zastosowania elementów dokładniejszych, a tym samym lepszy dobór tej proporcji, zaleca się skorzystanie z tej możliwości. Dodatkowym czynnikiem przemawiającym za stosowaniem rozwiązania w postaci drabinki rezystorowej R/2R jest bardziej równomierny rozkład obciążeń wprowadzanych przez poszczególne wejścia układu, co wiąże się z widzianą od strony wejścia jego rezystancją zastępczą. Pamiętajmy, że stan logicznej '1' na wejściu układu traktowany jest jako sygnał napięciowy o określonej wartości; równomierność obciążeń zapewni mniejsze spadki napięć na opornościach wewnętrznych wyjść i tym samym zwiększy dokładność przetwarzania.



Rys. 3 Schemat ideowy przystawki

Schemat ideowy przystawki przedstawiono na rys. 3. Rezystory R_1–R_{16} tworzą drabinkę R/2R, kondensator C_1 służy do filtracji przebiegu napięciowego, kondensator C_2 zapewnia separację sygnału stałego, a R_{17} i R_{18} tworzą układ wyjściowy. Poziom napięcia otrzymywanego na wyjściu dobrano tak, aby zapewnić poprawną pracę z typowym wzmacniaczem akustycznym. Ze względu na prostotę konstrukcji urządzenie jest jednak czułe na oporność obciążającego je wejścia – przy nietypowym wejściu wzmacniacza może zajść konieczność lepszego dopasowania go do warunków pracy poprzez zmianę wartości rezystora R_{18}.

Egzemplarz modelowy zmontowano bezpośrednio na wyprowadzeniach złącza szufladowego (tzw. złącze „męskie") stosując „montaż powietrzny" i układając elementy tak, że zmieściły się w plastikowej osłonie wtyczki. Podczas montażu należy zwrócić uwagę na możliwość powstania zwarć – przy wykonywaniu dłuższych połączeń za pomocą końcówek elementów należy na nie nasuwać kawałki koszulki izolacyjnej. Wyjście wyprowadzono kablem ekranowanym zakończonym wtykiem – w zależności od posiadanego sprzętu będzie to albo wtyk magnetofonowy (tzw. diodowy), albo wtyk typu cinch.

Uwaga: tak wykonana przystawka nie zapewnia separacji galwanicznej komputera i sprzętu elektroakustycznego (u-

rządzenia mają wspólną masę). Oznacza to, że urządzenia te muszą być zasilane z tej samej fazy zasilającej, w przeciwnym razie może nastąpić poważne uszkodzenie obu urządzeń! W praktyce, jeżeli nie jesteśmy pewni, czy zasilanie jest jednofazowe, przyjmijmy zasadę, że wszystkie urządzenia zasilamy z jednego gniazdka sieciowego stosując rozdzielacze (pamiętajmy jednak, aby nie przeciążyć gniazdka!).

Aby móc ocenić jakość zbudowanego przez nas przetwornika musimy dysponować oscyloskopem. Napisawszy (w dowolnym języku wyższego rzędu) program generujący przebieg, np. piłokształtny, uruchamiamy go i obserwujemy przebieg na wyjściu przystawki. Jeżeli urządzenie działa prawidłowo przebieg powinien cyklicznie narastać od zera do pewnej wartości, a następnie bardzo gwałtownie spadać do zera. Narastanie odbywać się powinno po linii prostej (w rzeczywistości jest to bardzo drobna linia schodkowa) bez „uskoków" i „wypłaszczeń". Jeżeli takowe występują, nie pomyliliśmy się w montażu i port drukarki jest na pewno sprawny, oznaczać to może, że albo źle dolutowaliśmy któryś z rezystorów, albo wartości niektórych elementów drabinki znacznie odbiegają od założonych.

Przykładowy program generujący przebieg piłokształtny na złączu LPT1 napisany w Turbo Pascalu:

```
program covox;
label 1;
const lpt1 = $3bc;
var i: byte;
begin
1:
  for i:=0 to 255 do port [lpt1]:=i;
  goto 1
end.
```

„Niełady" sposób organizacji pętli (instrukcja goto) wynika z tego, że czas realizacji tej instrukcji jest krótki, a co za tym idzie – przebieg wyjściowy jest mniej odkształcony od wzorcowej „piły". Program generuje przebieg o okresie zależnym od częstotliwości zegara komputera – ogólnie okres ma czas trwania rzędu pojedynczych milisekund. Aby zatrzymać program należy nacisnąć kombinację klawiszy Ctrl/C.

Program obsługuje port drukarki o adresie 3bch – możliwości konfiguracyjne komputerów PC dopuszczają inną lokalizację tego portu. W przypadku niepowodzenia radzimy spróbować adresów 378h i 278h, co w notacji pascalowej będzie miało postać: $378 i $278.

Życzymy miłych wrażeń przy słuchaniu muzyki z komputera.

Dariusz Adam Przygoda