

Cyfrowy efekt gitarowy

Technologia DSP dla muzyków



AVT 5215

Starsi elektronicy zapewne pamiętają czasy, gdy w czasopismach elektronicznych królowały różnego rodzaju przystawki do instrumentów muzycznych. Wśród nich można wymienić najbardziej popularne, wykonywane z przeznaczeniem do gitary, takie jak vibrato, tremolo i fuzz. Współczesna technika cyfrowego przetwarzania sygnałów umożliwia nie tylko generowanie tych prostych efektów, ale również tworzenie zupełnie innych, bardziej wyrafinowanych. Przykładem takiego rozwiązania jest prezentowane niżej urządzenie. Ma ono wbudowany specjalizowany procesor DSP, który tworzy efekty na bazie prostego do wykonania programu. Oprócz tego firma udostępnia wiele predefiniowanych nastaw, dzięki którym można tworzyć niepowtarzalne efekty dźwiękowe.

Rekomendacje: muzycy z żylką elektronika niezbędną do budowy układu i z zacięciem do eksperymentowania

Nie jestem gitarzystą. Nie potrafię nawet zagrać przysłowiowej ogniskowej piosenki na trzy chwyt. Skąd zatem pomysł na wykonanie cyfrowego efektu gitarowego? Zrodził się on przy okazji poszukiwania układów przeznaczonych do cyfrowej obróbki sygnałów analogowych. Do takich celów stosuje się specjalizowane procesory z wbudowanymi unikalnymi układami peryferyjnymi nazywanymi silnikami DSP (*DSP engine*). Cyfrowe operacje na zdigitalizowanym sygnale analogowym wymagają głównie operacji dodawania i mnożenia liczb wielobitowych (najczęściej w praktyce są to liczby 16-, 24- lub 32-bitowe). O ile dodawanie nie jest problemem, o tyle mnożenie na przykład 2 liczb 24-bitowych „zwykłym” CPU zajmuje sporo

czasu. W systemach DSP do obróbki sygnału audio tego czasu nie ma, ponieważ zwykle przetwarzanie wykonywane jest w czasie rzeczywistym. Dlatego mikroprocesory DSP są wyposażane w sprzętowe moduły mnożące, własną wielobitową jednostkę ALU i zestaw specjalizowanych instrukcji wspomagających typowe operacje na sygnałach.

Na rynku jest wiele rodzin mikroprocesorów DSP. Bardzo znana jest np. TMS320 z Texas Instruments. Na kawałek tortu z tego rynku ma również ochotę Microchip, a dowodem są mikrokontrolery DsPIC30 i DsPIC33.

Programowanie procesorów DSP nie jest rzeczą prostą i wymaga sporej wiedzy oraz jeszcze większego doświadczenia. Producenci robią, co mogą aby zachęcić do zrobienia

AVT-5215 w ofercie AVT:

AVT-5215A – płytką drukowaną
AVT-5215B – płytką drukowaną + elementy

Podstawowe informacje:

- Napięcie zasilania 6...12 VDC (wewnętrzny stabilizator 3,3 V)
- Płytką drukowaną: wyświetlacz 22×25 mm, efekt 46×160 mm
- Przeznaczony do współpracy z gitarą
- Wybór 1 z 8 efektów zaprogramowanych przez użytkownika lub predefiniowanych przez firmę
- Sterowanie: PIC16F628A firmy Microchip, generowanie efektów: FV-1 Spin Semiconductor

Dodatkowe materiały na CD i FTP:

- host: ep.com.pl, user: 12235, pass: 60u61c5y
- przykłady nagrań efektów
 - wzory płytek PCB
 - program
 - karty katalogowe i noty aplikacyjne elementów oznaczonych na Wykazie Elementów kolorem czerwonym

Projekty pokrewne na CD i FTP:

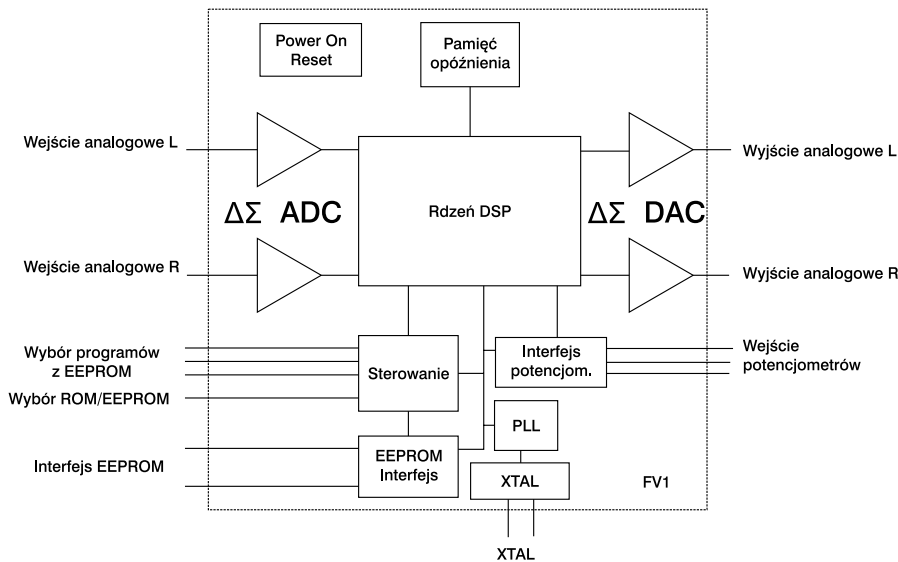
(wymienione artykuły są w całości dostępne na CD)

- AVT-302 Kompresor do gitary i basu (EP5/1996)
- AVT-303 Przystawka do gitary „Distortion” (EP6/1996)
- AVT-304 Gitarowa bramka szumów (EP7/1996)
- AVT-306 Chorus gitarowy (EP10/1996)
- AVT-313 Gitarowa kaczka (EP11/1996)
- AVT-314 Efekt tremolo – vibrato (EP12/1996)
- AVT-435 Prosty wzmacniacz do ćwiczeń gry na gitarze (EP7/2005)

przysłowiowego, pierwszego, najtrudniejszego kroku. Oferują coraz tańsze zestawy startowe oraz ewaluacyjne i edukacyjne wersje kompilatorów języka C/C++. Dodatkowo wraz z kompilatorem dostarczane są biblioteki zawierające typowo stosowane funkcje np. transformatę Fouriera i gotowe filtry cyfrowe.

Mimo wszystko zbudowanie kompletnego systemu do obróbki sygnałów akustycznych nie jest proste. Nawet jak uporamy się z programem, to w układzie oprócz mikrokontrolera trzeba zastosować odpowiedni kodek audio i obsługiwać w czasie rzeczywistym przepływ danych pomiędzy przetwornikami a mikrokontrolerem. To wyzwanie nie tylko dla programisty, ale również konstruktora.

To stosunkowo skomplikowane przedsięwzięcie prawdopodobnie spowodowało, że **ktos wpadł na pomysł, aby zaoferować łatwy do zastosowania, specjalizowany układ DSP przeznaczony tylko do obróbki sygnałów analogowych audio.** Dla użytkownika najlepiej byłoby, gdyby taki układ miał wbudowany wewnątrz kodek i wszystkie niezbędne układy, w tym pamięć RAM i pamięć programu.



Rys. 1. Schemat blokowy FV-1

Takim układem o unikalnych właściwościach połączonych z prostotą aplikacji jest procesor sygnałowy FV-1 firmy Spin Semiconductor.

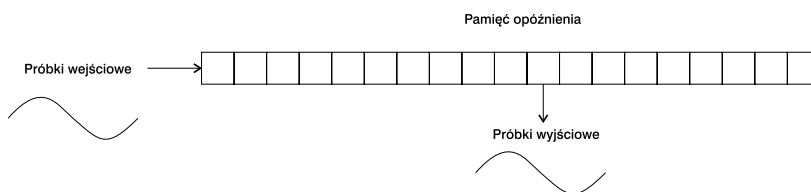
Procesor audio FV-1

O tym układzie pisaliśmy już w EP 11/2009. Przede wszystkim, zawiera on w swojej strukturze wszystkie niezbędne bloki funkcyjne i do działania potrzebuje tylko kilku elementów zewnętrznych, w tym łatwo dostępnej pamięci EEPROM 24LCxx z interfejsem I²C. Pamięć ta przeznaczona jest do zapamiętania programu. Co ważne, układ jest tak skonstruowany, że pamięć EEPROM można zaprogramować w systemie.

FV-1 ma wbudowany stereofoniczny, 24-bitowy kodek audio z przetwornikami sigma-delta. Przetworniki te mają niezłe parametry: poziom szumów (A-ważony) typowo -97 dB i zniekształcenia 0,015% dla przetworników A/C i C/A. Pasma przenoszenia jest ograniczane do 15 kHz (nierównomierność ±0,5 dB) dla standardowej częstotliwości taktowania układu kwarcem zegarkowym o częstotliwości 32,768 kHz. Podniesienie częstotliwości taktowania do 48 kHz rozsze-

rza pasmo przenoszenia do 20 kHz. Wszystkie niezbędne sygnały zegarowe są generowane przez wewnętrzny układ PLL.

Schemat blokowy FV-1 pokazano na rys. 1. Układ jest zasilany pojedynczym napięciem +3,3 V. Wejściowe sygnały analogowe są konwertowane przez przetworniki A/C Σ-Δ. Po konwersji, rdzeń DSP odczytuje dane z wyjść przetworników i poddaje dalszej obróbce. Układ może wykonywać 16 różnych programów wybieranych sprzętowo, stanami wymuszonymi na trzech wejściach sterujących. Podczas produkcji w pamięci ROM układu zapisywanych jest osiem fabrycznych programów. Ta przestrzeń nie może być wykorzystywana przez programistę, chyba że będą to programy zapisane na zamówienie przez producenta w procesie produkcji. Układ pozwala nie tylko na użycie gotowych funkcji, ale również na tworzenie własnych. Podanie stanu wysokiego na wejście sterujące T0 oznaczone na rys. 1 jako wejście wyboru ROM/EEPROM powoduje, że rdzeń będzie wykonywał jeden z ośmiu programów użytkownika, zapisanych w zewnętrznej pamięci EEPROM.



Rys. 2. Generowanie stałego opóźnienia

Adres	Nazwa	R/W	Opis
0	SIN0_RATE	W	SIN0 współczynnik częstotliwości
1	SIN0_RANGE	W	SIN0 zakres
2	SIN1_RATE	W	SIN1 współczynnik częstotliwości
3	SIN1_RANGE	W	SIN1 zakres
4	RMPO_RATE	W	RMPO współczynnik częstotliwości
5	RMPO_RANGE	W	RMPO zakres
6	RMP1_RATE	W	RMP1 współczynnik częstotliwości
7	RMP1_RANGE	W	RMP1 zakres

W strukturze układu jest umieszczony blok interfejsu potencjometrów analogowych, niespotykany w innych tego typu konstrukcjach. Do trzech doprowadzeń układu można podłączyć wyjście potencjometru dzielącego napięcie zasilania. Napięcie z suwaka jest przekształcane na odpowiednią liczbę 10-bitową i zapisywana w rejestrze w obszarze SFR. Odczytana z potencjometru wartość napięcia zmienia się w zakresie 0...0,99 i może być użyta w programie jako współczynnik ustalający wartość nastawianego parametru. Użycie klasycznego potencjometru obrotowego jest zwykle wygodniejsze niż zastosowanie klawiszy góra/dół.

Pamięć opóźnienia ma pojemność 32 k-słów 14-bitowych. W pamięci zapisywane są liczby 24-bitowe w skompresowanym formacie zmiennoprzecinkowym S23. W czasie zapisu jest przeprowadzana kompresja danych, a przy odczycie dekompresja.

Prawdziwa, sygnałowa natura układu wynika z zastosowania zestawu instrukcji assemblerowych DSP. Instrukcje te umożliwiają programowe tworzenie efektów akustycznych wykorzystywanych między innymi w typowych efektach gitarowych, do tej pory wykonywanych w technice analogowej.

Jednym z podstawowych bloków wykorzystywanych przy konstrukcji efektów gitarowych jest generator LFO (*Low Frequency Oscillator*). W układzie cyfrowym funkcja LFO jest realizowana programowo z wykorzystaniem pamięci opóźnienia. Do dyspozycji są cztery generatory: dwa sygnały sinusoidalnego i dwa sygnały piłokształtnego (*Ramp Generator*). Z generatorami jest związanych osiem rejestrów obszaru SFR. Wymieniono je w **tab. 1**.

Generowanie sygnału LFO omówimy, zaczynając od prostej realizacji opóźnienia. Do pamięci opóźnienia wpisywane są próbki wejściowe z częstotliwością próbkowania. Żeby uzyskać opóźnienie na przykład 20 próbek, trzeba ustawić wskaźnik adresowania przesunięty o 20 w stosunku do adresu zapisu. W ten sposób są zawsze odczytywane próbki zapisane wcześniej o czas równy 20 okresom częstotliwości próbkowania. Sygnał na wyjściu ma niezmienny kształt i częstotliwość, ale jest opóźniony w stosunku do źródła. Jeżeli teraz próbki napływają do pamięci ze stałą prędkością, a wskaźnik odczytu pamięci jest przesuwany z jakąś prędkością w kierunku miejsca zapisu nowych próbek, to odczytane z pamięci próbki reprezentują częstotliwość większą niż częstotliwość sygnału wejściowego (**rys. 3**).

Jak się łatwo domyślić, przesuwanie wskaźnika odczytu w drugim kierunku spowoduje zmniejszanie częstotliwości odczytywanych próbek. Prędkość przesuwania wskaźnika może być liniowa lub zmieniać się zgodnie z funkcją sinus (**rys. 4**). Sygnał wyjściowy jest modulowany częstotliwo-

ściowo. Zakres zmiany częstotliwości jest określany zakresem zmian wskaźnika, czyli amplitudą modulacji. Częstotliwość modulacji jest proporcjonalna do szybkości zmian stanu wskaźnika pamięci.

Aby opisywaną wyżej funkcję zrealizować z użyciem FV-1, należy najpierw zdefiniować obszar w pamięci używany do odmierzenia opóźnień. Zarezerwowana na opóźnienie pamięć musi mieć rozmiar proporcjonalny do żądanej amplitudy modulacji. Na przykład, jeżeli modulacja przebiegiem sinusoidalnym będzie miała amplitudę 1024 próbek, to obszar zarezerwowany na opóźnienie musi mieć rozmiar 2049 próbek (± 1024 oraz liczba 0). W implementacji sinusoidalnego generatora LFO modulującego częstotliwość wejściową z amplitudą 1024 próbek wskaźnik ustawiany jest na adres względny 1024. Mamy wtedy pewność, że w czasie modulacji nie zostaną pobrane próbki spoza obszaru zapisywanego w tej implementacji LFO. Teraz w tak zdefiniowanej pamięci trzeba zapisywać próbki kolejno nadchodzące z przetwornika A/C. Musi być też możliwość ustalania i modyfikowania wskaźnika dostępu do pamięci. W FV-1 jest to realizowane w ten sposób, że do adresu zawartego w instrukcji dostępu do pamięci dodawana jest wartość licznika zliczającego w dół i dekrementowanego przy każdym cyklu próbkowania. Pozwala to na zapisywanie próbek z niższym adresem i odczytywanie z wyższym. Na przykład, dla opóźnienia równego 20 próbek są one zapisywane pod adres 0, a odczytywane spod adresu 20 (dziesiątynie).

W praktyce, do wyliczenia amplitudy generatora sinusoidalnego LFO używa się liniowej interpolacji funkcji sinus opartej na dwóch współczynnikach, zapisywanych w obszarze pamięci SFR (tab. 1). Pierwszy współczynnik (*SIN_RATE*) określa częstotliwość LFO zgodnie z zależnością:

$$K_f = 2^{17} \cdot \frac{2\pi f}{R}$$

gdzie: f – żądana częstotliwość LFO w Hz,
 R – częstotliwość próbkowania w Hz.

Dopuszczalny zakres zmian współczynnika K_f to 0...511. Pozwala to na pracę LFO z częstotliwością w zakresie 0...20 Hz przy częstotliwości próbkowania wynoszącej 32,768 kHz.

Współczynnik określający amplitudę zmian (*SIN_RANGE*) jest wyliczany z zależności:

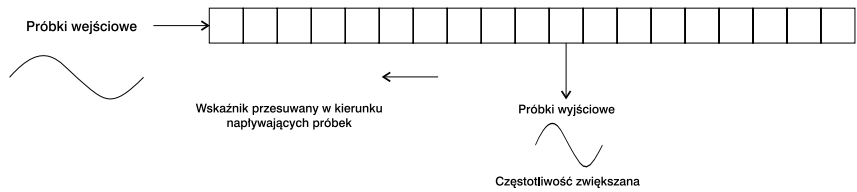
$$K_a = \frac{N \cdot 32767}{16385}$$

gdzie N – liczba próbek opóźnienia.

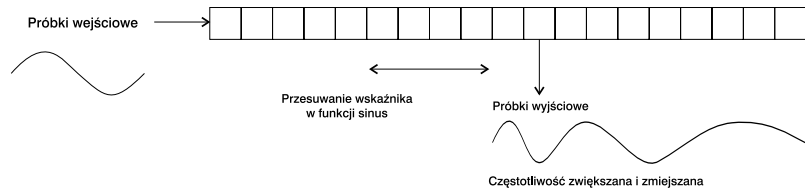
Dopuszczalny zakres zmian K_a to 0...32767.

Zaprogramowanie LFO jest bardzo proste. Wystarczy wyliczyć żądane współczynniki i zapisać rejestry rozkazem WLDS (rys. 5).

Bardzo podobnie działa *LFO Ramp Generator* umożliwiający modulację przebiegiem piłokształtnym. W odróżnieniu od modulacji



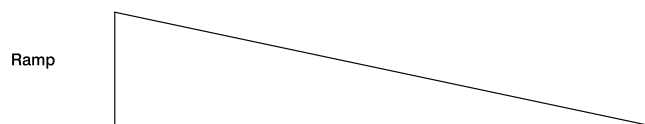
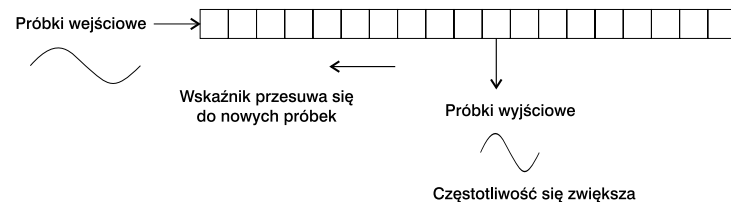
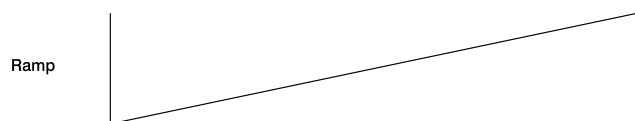
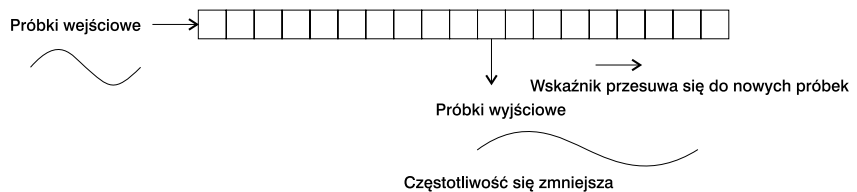
Rys. 3. Zwiększanie częstotliwości sygnału wejściowego



Rys. 4. Modulowanie częstotliwości sygnału wyjściowego

Mnemonik	Kod rozkazu	
WLDS	00NFFFFFFFAAAAAAAAAAAAAA10010	
Nazwa	Długość	Opis
N	1 bitów	Wybór SIN LFO(0, 1)
F	9 bitów	Zakres 0...511
A	15 bitów	Zakres 0...32767

Rys. 5. Rozkaz WLDS



Rys. 6. Zasada działania generatora LFO ramp

sinus, tutaj wskaźnik będzie zawsze poruszał się w jednym kierunku: albo od próbki zerowej do wartości maksymalnej (amplituda modulacji) dla zmniejszenia częstotliwości, albo od maksymalnej wartości do próbki zerowej przy zwiększaniu częstotliwości (rys. 6).

Tworzenie efektów dźwiękowych

Generatory LFO są podstawą efektów gitarowych. Jednym z najprostszych jest *tremolo*. Polega on na modulacji amplitudy

sygnału sinusoidalnym przebiegiem wolnozmennym. W materiałach firmowych można znaleźć kod źródłowy kompletnego efektu tremolo, którego prędkość jest regulowana położeniem potencjometru Pot1, a poziom wyjściowy potencjometrem Pot2. Fragmenty tego programu pokazano na list. 1. W części inicjalizacyjnej jest definiowany generator sinusoidalny LFO1 ze współczynnikami $K_f=0$ i $K_a=16383$. Potem, na podstawie wartości odczytanej z potencjometru Pot1 modyfikowana jest wartość K_f .

List. 1. Fragment programu efektu tremolo

```

;Pot1 = Tremelo rate
;pot2 = Effect level

...
;fragment inicjalizacji efektu
wlds sinl,0,16383 ;inicjalizacja LFO1 SIN F=0, A=16383
;wykonanie efektu tremolo z kontrola LFO SIN1 potencjometrem Pot1
rdax pot1,1 ;odczytaj wartość z potencjometru Pot1 (rejestr
potencjometru)
mulx pot1 ;podnieś odczytaną wartość do kwadratu
sof 0.4,0.1 ;skalowanie prędkości efektu do rozsądnych granic
wrax sinl rate,0 ;zapisanie wartości do rejestru współczynnika
częstotliwości ;SIN1_RATE
cho rdal,sinl odczytanie próbki z pamięci opóźnienia
i przeskalowanie przez 0,5
;i przesłanie do akumulatora
sof 1,0.5 ;dodanie do akumulatora stałej 0,5
wrax trem,0 ;przesłanie do zmiennej odpowiedzialnej za modulację
sygnału

rdax mono,0.5 ;przeskalowanie sygnału wejściowego
mulx trem ;ACC=mono*trem
sof 0.75,0 ;ACC=ACC*0,75
rdax mono,-0.5 ;ACC=ACC+mono*-0,5
mulx pot2 ;ACC=ACC*pot2
rdax mono,0.5 ;ACC=ACC+mono*0,5
sof 1.999,0 ;ACC=ACC*1,999 +0
rdax revout,1 ;ACC=ACC+revout*1
sof 1,0.2 ACC=ACC+0,2
wrax dacl,1 ACC->DACL
sof 1,-0.4
wrax dacr,0 ACC->DACR

```

Wykaz elementów**Rezystory:** (SMD, 1206)

R5: 100 Ω
R9, R10, R13, R14, R16, R17: 150 Ω
R1 R2 R3 R4 R6, R8, R11 R12: 1 kΩ 1206
R7, R9, R10: 10 kΩ
P1, P2, P3: potencjometry 10...50 kΩ
liniowe (A), do druku

Kondensatory:

C4: 10 pF (ceramiczny, 1206)
C7, C10: 2,2 nF (ceramiczny, 1206)
C2, C3, C6, C13, C14: 100 nF (1206)
C8, C9: 1 μF/16 V (SMD)
C1, C5, C11: 10 μF/16 V (tantalowy, SMD)

Półprzewodniki:

U1: FV-1
U2, U3: pamięć EEPROM 24LC128
zaprogramowana (opis w tekście)
U4: SPX1117-3,3V (TO-252)
U5: PIC16F628A zaprogramowany

Inne:

Kwarc zegarkowy 32,768 kHz
Listwy goldpinów
Wyświetlacz 7-segmentowy LED o wspólnej
anodzie

Program nie jest zbyt skomplikowany. Pisanie programów dla FV-1 różni się od tego dla klasycznych mikrokontrolerów. Nie ma tu pętli, w której klasyczny program musi się wykonywać. Program dla FV-1 kończy się tak po prostu, bez żadnego rozkazu skoku. Jest zapętłony sprzętowo i wykonywany cyklicznie przy każdym pobraniu próbki z prze-

tworników analogowo-cyfrowych ADC. Inaczej mówiąc, program jest wywoływany z częstotliwością próbkowania.

Z sekwencyjnym wykonywaniem programu jest związana flaga *RUN*. Może ona być testowana sprzętowo przez rozkaz ominięcia wykonywania następnego rozkazu lub grupy rozkazów. Po załadowaniu programu do

wykonania z zewnętrznej pamięci EEPROM do pamięci FV-1, flaga *RUN* jest ustawiana, a po pierwszym wykonaniu programu jest zerowana. Można to wykorzystać w czasie inicjacji generatorów LFO czy zmiennych programu. Podczas pierwszego przebiegu programu wykonywane są instrukcje inicjalizacji dla *RUN*=1, a w czasie następnych

R E K L A M A



Zdrowych i pogodnych Świąt Bożego Narodzenia.

Kolorowej i pachnącej choinki,

wymarzonych prezentów,

radości i spokoju.

A w Nowym 2010 Roku

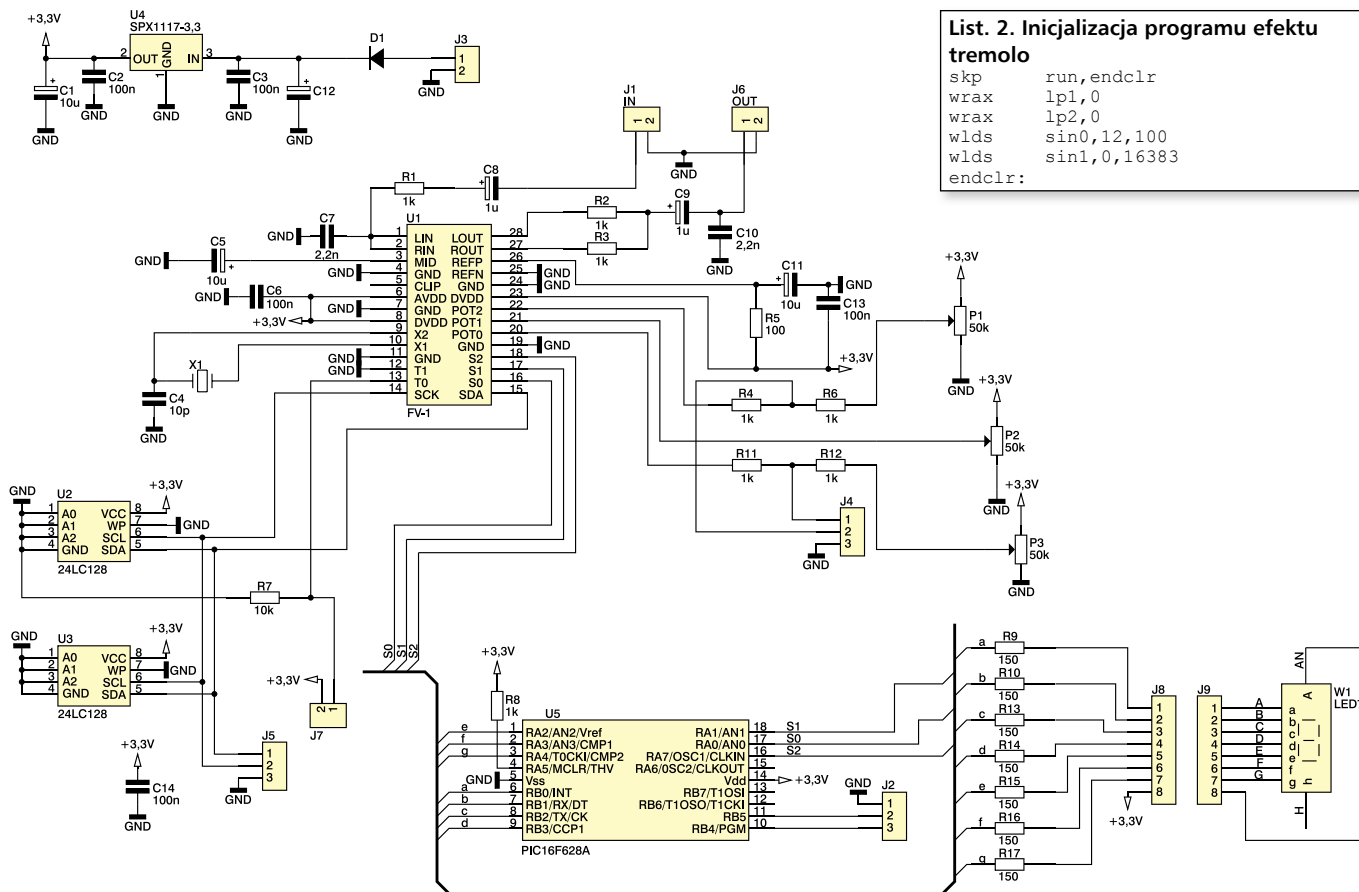
stuszných decyzji,

marzeń wielkich i odważnych

i odwagi by je spełniać.

Życzą Zarząd i Pracownicy

T M E
Electronic Components



List. 2. Inicjalizacja programu efektu tremolo

```

skp      run,endclr
wrxax   lp1,0
wrxax   lp2,0
wlds    sin0,12,100
wlds    sin1,0,16383
endclr:
    
```

Rys. 7. Schemat ideowy efektu

te instrukcje są pomijane. W omawianym wyżej programie tak inicjowane są generatory LFO SIN1, LFO_SIN0 (dla efektu chorus) i obszary pamięci (list. 2).

Omawianie listy rozkazów i dokładne wyjaśnianie zasad pisania programu znacznie wykracza poza ramy tego artykułu. Przytoczony tutaj przykład miał na celu pokazanie idei tworzenia aplikacji efektu. Dokładne dane, wraz z programami w wersji źródłowej, można znaleźć na stronach producenta, można znaleźć na stronach producenta. Jest tam też zamieszczony dokładny opis architektury i praktyczne wskazówki dotyczące pisania programów. Wróćmy zatem do praktycznej realizacji układu generującego efekty gitarowe. Schemat praktycznie wykonanego urządzenia pokazano na rys. 7. Jest to w zasadzie kopia opracowania dostępnego na stronie producenta. Układ jest tak prosty, że przynajmniej na etapie poznawania właściwości FV-1 modyfikacje wydały mi się nieuzasadnione. Jedyną różnicą jest sposób wybierania aktywnego efektu. Tym zadaniem zajmuje się prosty sterownik z mikrokontrolerem PIC16F628A.

Budowa

Mimo że układ FV-1 jest stereofoniczny, to przy wytwarzaniu efektów gitarowych oczywiste jest wykorzystanie go w wersji monofonicznej. Sygnał wejściowy z gitary jest podawany przez kondensator C8 usuwający ewentualną szkodliwą stałą.

Rezystor R1 i kondensator C7 są elementami filtra dolnoprzepustowego na wejściu przetwornika A/C. Filtrowanie jest niezbędne dla wyeliminowania aliasingu. Wejścia LIN i RIN są zwarte, a sygnał wejściowy jest podawany na oba przetworniki jednocześnie. Sygnał wyjściowy z przetworników DAC lewego i prawego kanału jest sumowany na rezystorach R2 i R3. Kondensator C9 eliminuje szkodliwą stałą, a kondensator C10 (łącznie z rezystorami R2 i R3) filtruje dolnoprzepustowo sygnał wyjściowy z przetwornika. Jest to tak zwany dolnoprzepustowy filtr rekonstruujący. Sygnały wejściowy i wyjściowy są podłączane przez złącza J1 i J6. Są to stereofoniczne złącza typu jack.

Pamięć EEPROM (U2 lub U3) z zapisanymi programami efektów jest połączona z układem FV-1 liniami SDA i SCL magistrali I²C. Wszystkie linie adresowe pamięci: A0, A1 i A2, mają stan niski. Na schemacie są narysowane 2 pamięci połączone równolegle. Na płytce drukowanej przewidziano miejsce na pamięć w obudowie SMD i obudowie do montażu przewlekane go DIL. W trakcie pracy układu może być na płytce tylko jedna z tych pamięci. Ja wykorzystałem pamięć w obudowie SMD (U3) programowaną w układzie przez złącze J5.

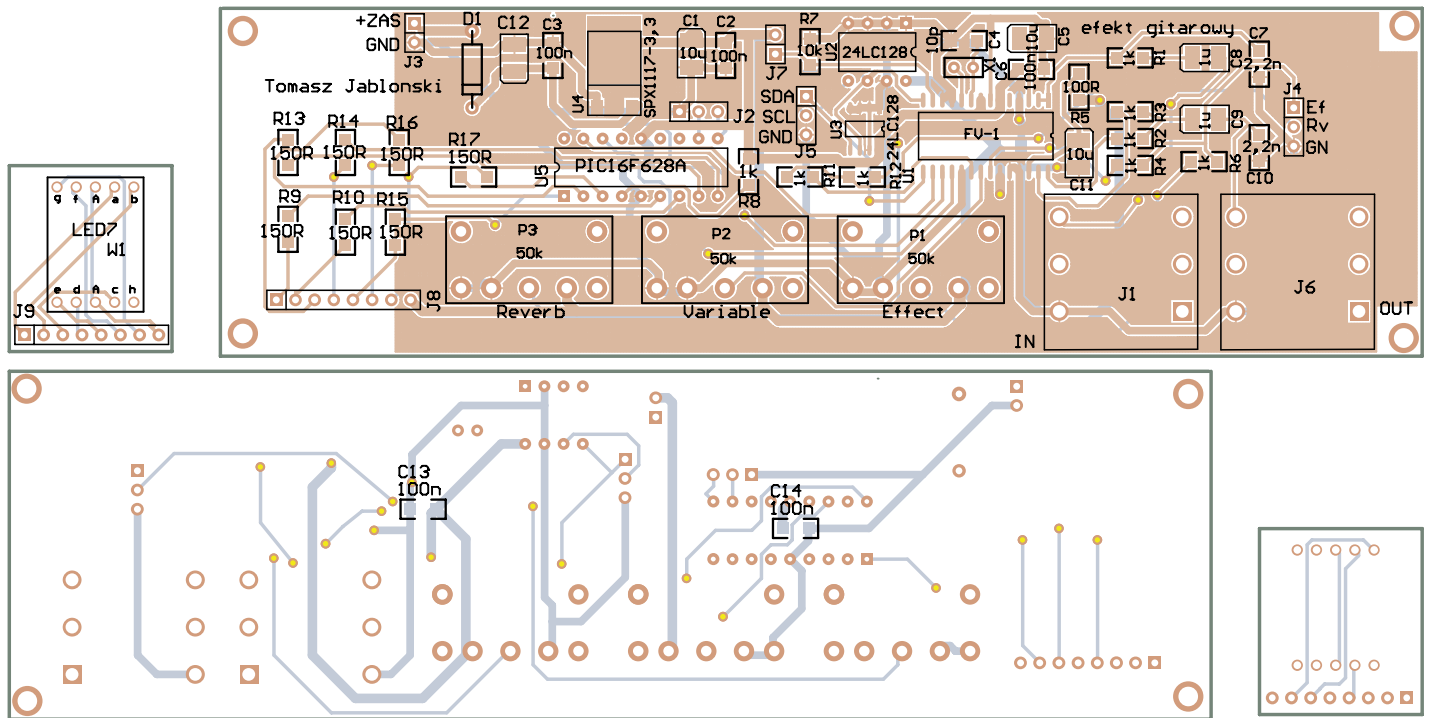
Pomiędzy suwakami potencjometrów P1 i P3 a wyprowadzeniami POT0 i POT2 są włączono po dwa rezystory 1 kΩ połączone szeregowo. Umożliwiają one zwiera-

nie ustawień potencjometrów przez układy pedalał podłączanych przez złącze J4.

Cały układ jest zasilany pojedynczym napięciem +3,3 V ze stabilizatora SPX1117-3.3 (U4). Przy wyprowadzeniach zasilania FV-1 umieszczono ceramiczne kondensatory o pojemności 100 nF (C6 i C13). Wewnętrzne napięcie referencyjne przetworników jest blokowane kondensatorem C11.

W pierwszej wersji efektu do testów poprawności wybierania efektów użyłem mechanicznego enkodera 16 na 4. Pierwszych 8 pozycji enkodera wybierało programy demonstracyjne układu FV-1 zapisane w pamięci ROM. Następnich 8 pozycji wybierało 8 programów zapisanych w pamięci EEPROM. W trakcie testów efektu przeprowadzonych przez gitarzystów okazało się, że zastosowany enkoder jest niewygodny i lepszym rozwiązaniem byłoby sekwencyjne przełączanie efektów za pomocą 2 styków. Umożliwiłoby to przełączanie efektów stykami zwierzanymi nogą bez przerywania gry. Takie sterowanie można realizować na wiele sposobów, ale najwygodniej jest do tego celu wykorzystać mały tani mikrokontroler – na przykład PIC16f628A.

Numer aktywnego efektu sterownik wyświetla na pojedynczym 7-segmentowym wyświetlaczu LED ze wspólną anodą. Rezystory R9, R10, R13, R14, R15, R16 i R17 ograniczają prądy segmentów. Wyświetlacz



Rys. 8. Płytki drukowane: główna i wyświetlacz

LED wyświetla numer efektu w zakresie od 0 do 7. Z tym numerem jest powiązany stan linii RA0, RA1 i RA7 wyboru efektu dołączonych do linii sterujących S0, S1 i S2 układu FV-1. Jeżeli na przykład na wyświetlaczu jest

wartość 6, to sekwencja wybierająca efekt ma wartość binarną 110. W ten sposób można wymusić osiem różnych stanów na liniach S0, S1 i S2 wybierających jeden z efektów. Pozostaje jeszcze wybór pomiędzy efektami

zapisanymi w pamięci stałej FV-1 a efektami zapisanymi w pamięci EEPROM. Do tego celu służy wejście T0 układu FV-1 połączonego z pinem złącze J7. Jeżeli chcemy używać tylko efektów zapisanych w pamięci

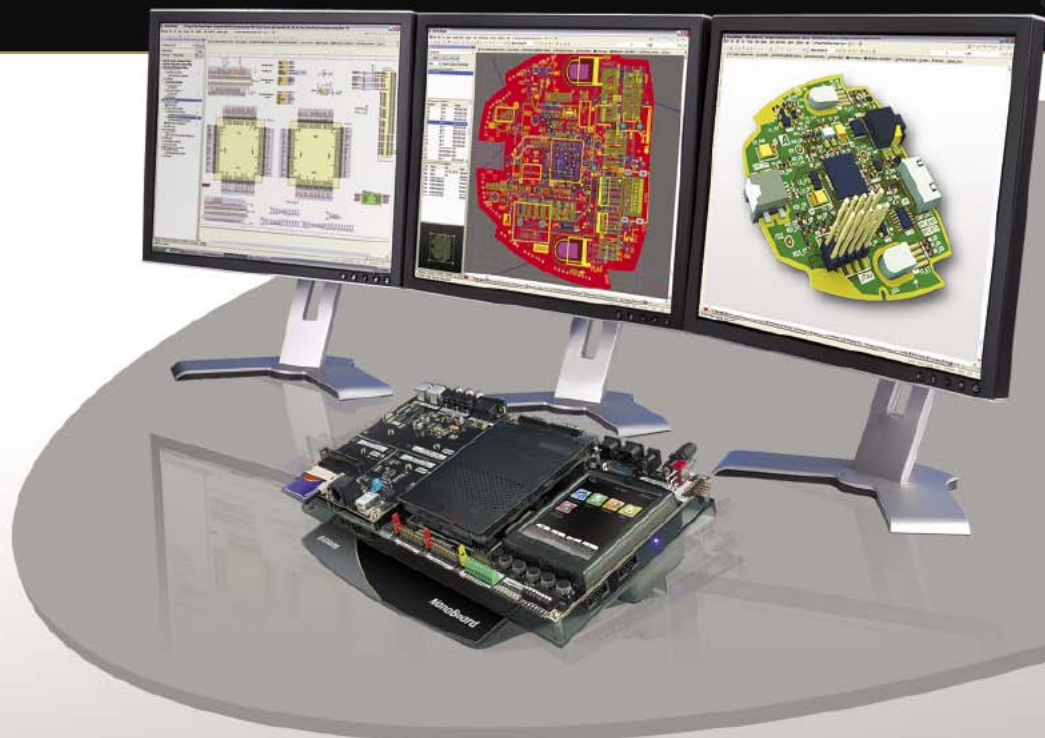
R E K L A M A

Altium Designer

Zostań Pionierem! Wyprzedź Pozostałych

Altium oferuje narzędzia, które ułatwiają realizację złożonych projektów urządzeń elektronicznych. Otrzymujesz najnowsze technologie i cały potencjał, abyś mógł swobodnie realizować swoje pomysły.

Teraz oferujemy większe możliwości za niższą cenę. Sprawdź nasze promocje.



ci EEPROM, to na płytce, zwierając zworke J7, trzeba połączyć T0 z napięciem +3,3. Po rozwarciu zworki stan niski jest wymuszany przez rezystor R7.

Styki przełączające aktywny efekt trzeba dołączyć do złącza J2. Sekwencyjna zmiana efektu następuje po zwarciu linii RB4 (górze) lub RB5 (dół) do masy. Numer wybranego efektu jest zapisywany w pamięci EEPROM mikrokontrolera. Po wyłączeniu i ponownym włączeniu efektu zawsze się ustawi ostatnio zapamiętany numer.

Płytkę drukowaną efektu pokazano na rys. 8. Praktycznie wszystkie elementy oprócz mikrokontrolera, opcjonalnej pamięci EEPROM i potencjometrów oraz gniazd wejściowego i wyjściowego są w obudowach do montażu powierzchniowego. Jednak ich wielkość i rozstaw nóżek nie sprawiają żadnych problemów w trakcie montażu komuś, kto ma odrobinę wprawy w lutowaniu elementów SMD. Oprócz głównej płytki trzeba zmontować małą płytkę, na której jest tylko wyświetlacz LED. Tę płytkę trzeba przez listwę kątowych goldpinów połączyć z główną płytką przez złącze J8. W ten sposób wyświetlacz jest w jednej płaszczyźnie z potencjometrami i gniazdami.

Po zmontowaniu trzeba sprawdzić poprawność napięcia zasilającego i można przystąpić do programowania pamięci. Jeżeli wykorzystuje się pamięć EEPROM w obudowie SMD, to do złącza J5 doprowadzamy sygnały SDA, SCL i GND programatora. Ja do zaprogramowania użyłem programatora T51prog firmy Elnec. W podstawce programatora umieściłem podstawkę DIL8 z przylutowanymi trzema przewodami do nóżek: 4 – MASA, 5 – SDA, 6 – SCL. Te kable potem dołączyłem do złącza J5 (rys. 7). Za pomocą tak spreparowanego kabela programowanie przebiega bez problemu, oczywiście po wcześniejszym zasileniu układu. Można też wykorzystać pamięć w obudowie do montażu przewlekanej, zaprogramować ją w programatorze, a potem umieścić w podstawce na płytce. Oba sposoby są dobre.

Programowanie

Programy można napisać samodzielnie, ale na początek najlepiej jest skorzystać z przykładowych programów przygotowanych przez producenta i udostępnionych na stronie internetowej http://www.spinsemi.com/app_download/GA_DEMO_programs.zip. Po ściągnięciu i rozpakowaniu mamy do dyspozycji projekt z programami demonstracyjnymi ośmiu efektów (tab. 2). Każdy z programów ma wbudowaną funkcję pogłosu *Reverb* regulowanego potencjometrem Pot0. W pakiecie demonstracyjnym jest też umieszczony plik wynikowy. Niestety, zaprogramowanie plikiem, którym ja dysponowałem, nie było możliwe, ponieważ programator zgłaszał błędy CRC dla każdej z linii

Tab. 2. Programy demonstracyjne udostępniane przez firmę Spin Semiconductor

Nr efektu	Nazwa efektu	Regulacje		
		Pot0	Pot1	Pot2
#	program	reverb	variable	Effect level
0	Chorus	Reverb	Rate	Level
1	Flange	Reverb	rate	Level/regen
2	Tremolo	Reverb	Rate	Depth
3	Echo	Reverb	Delay time	Level
4	Echo/rpt	Reverb	Delay time	Level
5	Wah	Reverb	Sensitivity	Level/peaking
6	Vibrato	Reverb	Rate	Level
7	Phase	Reverb	Rate	Width

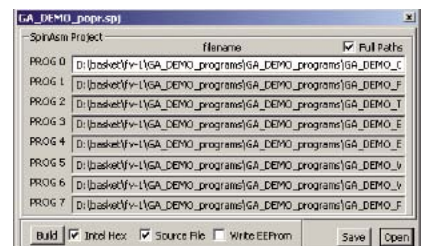
pliku. Pracownik pomocy technicznej firmy Spin stwierdził, że plik był najprawdopodobniej skompilowany starszą wersją kompilatora, którą generował plik wynikowy z błędami. Zaproponował też żeby pliki źródłowe skompilować na nowo najnowszą wersją kompilatora. Żeby to zrobić, trzeba ze strony producenta pobrać plik instalacyjny *SpinSetup_1_1_28.msi* i zainstalować pakiet SpinAsm IDE. Na rys. 9 pokazano okno programu z otwartym oknem menedżera testowej aplikacji ośmiu efektów. W każdym z ośmiu okien jest wyświetlona ścieżka dostępu do pliku źródłowego konkretnego efektu. Okno może nie być aktywne i wtedy w obszarze odpowiadającego mu programu nie zostanie nic zapisane. W IDE można otwierać i edytować pliki źródłowe. Kiedy projekt jest gotowy, wtedy na dole menedżera wybiera się trzy opcje działania: generowanie pliku hex w formacie Intel, generowanie wersji źródłowej lub zaprogramowanie pamięci EEPROM. Plik Intel Hex jest niezbędny do zaprogramowania pamięci przez zewnętrzny programator i ta opcja będzie dla nas najbardziej interesująca.

Po skompilowaniu tym razem nie było problemu z wczytaniem pliku do programatora

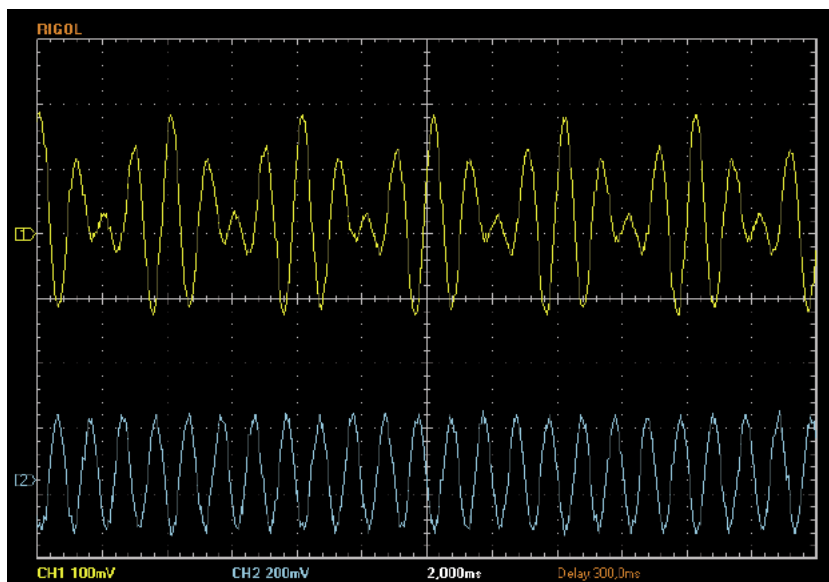
i zaprogramowaniem pamięci. Druga opcja pliku źródłowego jest również bardzo ciekawa, bo w wyniku kompilacji jest tworzonych osiem tablic (po jednej dla każdego z programów), które można wkleić do programu napisanego w języku C. Można to wykorzystać do zbudowania czegoś w rodzaju mikroprocesorowego sterownika, który będzie między innymi ładował program do pamięci EEPROM.

Trzecia opcja jest używana do kompilacji i przesłania wyniku do firmowego modułu ewaluacyjnego firmy Spin.

Środowisko SpinAsm IDE jest niezastąpione w trakcie pracy nad projektem. Oprócz możliwości kompilacji automatycznie są przyznawane obszary pamięci dla poszczególnych programów.



Rys. 9. Okno programu SpinAsm IDE



Rys. 10. Działanie efektu Pitch Shift

Uchem gitarzysty...

Przez redakcję Magazynu Gitarzysty przewijają się dziesiątki efektów, czasem najsłynniejszych marek w branży gitarowej. Multieffekt zbudowany przez ekipę AVT przeszedł moje najśmielsze oczekiwania. Barwy jakie oferuje są nie tylko dobre jakościowo, ale również użyteczne (co w świecie efektów gitarowych nie jest takie oczywiste, jakby się mogło wydawać). Na tych barwach po prostu chce się grać. Żeby to udowodnić nagrałem kilka przykładów (audio znajdziecie na załączonym CD). Poniżej opis kolejnych próbek:

- 1 - krótki, pojedyncze echo użyty do zdwojenia ilości słyszanych dźwięków,
- 2 - chorus podbarwiający rozłożone akordy,
- 3 - tremolo użyte w celu uzyskania efektu retro,
- 4 - delikatne echo dodające przestrzeni solówce,
- 5 - głośne echo spełniające rolę drugiego głosu,
- 6 - automatyczne wah (filtr) idealne do riffów funki,
- 7 - intensywny chorus imitujący Univibe i styl J. Hendriksa,
- 8 - tekstury przypominające syntezator (gałka VOLUME gitary + długie echo).

Krzysztof Inglik
Magazyn Gitarzysty

Uruchomienie

Po zmontowaniu i zaprogramowaniu firmowymi programami demonstracyjnymi pamięci EEPROM można przystąpić do sprawdzania działania.

Napięcie zasilające o wartości minimalnej +6 V trzeba dołączyć do złącza J3. Według specyfikacji producenta sygnał wejściowy powinien mieć amplitudę nie większą niż 0,5 Vpp. Do pierwszych testów użyłem standardowo generatora częstotliwości akustycznych i oscyloskopu. Można w ten sposób sprawdzić, po pierwsze, czy układ w ogóle działa. Ważne jest też sprawdzenie, czy nie ma wzbudzeń lub szumów na wyjściu. Oczywiście takie testy są niezbędne dla nowo projektowanej konstrukcji i w prakty-

ce uruchamiając urządzenie prawidłowo zaprojektowane, można ten etap pominąć. Bardziej dociekliwi mogą zobaczyć, jak wygląda sygnał wyjściowy po modyfikacji przez tor efektu. Na rys. 10 pokazano zrzut ekranu oscyloskopu podczas generowania jednego z efektów. Na wejście efektu podałem sygnał o częstotliwości 1 kHz i amplitudzie ok. 500 mV. Przebieg wejściowy jest widoczny na dole ekranu (kolor niebieski). Drugi przebieg na oscylogramie jest mierzony na wyjściu efektu przy ustawieniu fabrycznie umieszczonego w pamięci ROM efektu numer 3 – *Pitch Shift*. Wybrałem ten efekt, bo jego działanie jest dobrze widoczne na statycznym zrzucie, ale zmiana sygnału wyjściowego wskutek działania innych efektów

również daje pojęcie o tym, jak funkcjonuje dany algorytm.

Uruchomiony układ można spróbować wpiąć w tor wzmacniacza. Producent przestrzega, żeby sygnał z wyjścia ewentualnego efektu distortion był sygnałem wejściowym dla FV-1 a nie odwrotnie.

Na koniec pozostaje jeszcze kwestia dostosowania testowych programów do własnych potrzeb. Tory wzmacniaczy mogą się różnić między sobą na przykład czułością. Być może niezbędna będzie modyfikacja takich parametrów, jak na przykład poziom sygnału wyjściowego. Programy przykładowe producenta mogą być do tego celu znakomitym początkiem.

Tomasz Jabłoński, EP
tomasz.jablonski@ep.com.pl

R E K L A M M A

ELECTRONIC COMPONENTS
TVSAT
ELECTRONIC

Podzespoły elektroniczne w ilościach hurtowych

Układy scalone i elementy bierne

Zawsze aktualna oferta, oraz sklep internetowy:
www.tvsat.com.pl

*

ul. Brukowa 8, 05-092 Łomianki
tel. 22 864 77 85, faks 22 864 77 86

*

e-mail: tvsat@tvsat.com.pl; sakos@medianet.pl

artronic

AV-DISPLAY

128K64 COG TYPE
FFSTN TECHNOLOGY
EXTENDED TEMPERATURE
V/G BACKLIGHT
ST7565R

128K64 COG TYPE
FFSTN TECHNOLOGY
EXTENDED TEMPERATURE
WHITE BACKLIGHT
UC1601

248K128
FFSTN
WHITE BACKLIGHT
EXTENDED TEMPERATURE
DOUBLE FSTN TECHNOLOGY
UC1608

▶ **Technologia Chip On Glass**
 ▶ **Nowoczesne sterowniki**
UC1601, UC1608, ST7565R
 ▶ **Niska cena!**
 ▶ **Zwarta konstrukcja**
 ▶ **Małe gabaryty**
biuro@artronic.pl (058) 668 57 83-84