

# TMS320F2833x

## DSP w zastosowaniach Do It Yourself



Cyfrowe przetwarzanie sygnałów z zastosowaniem procesorów sygnałowych (DSP), to w rozumieniu wielu elektroników rzecz niewykonalna w warunkach domowych. Jak często bywa, decydują o tym w znacznej mierze stereotypy. Współcześnie temat jest w miarę łatwy do opanowania i to bez ponoszenia dużych nakładów finansowych. Pisząc ten artykuł, chciałbym obalić pewne mity związane z programowaniem DSP, a tym samym zachęcić do próby zmierzenia się z procesorami sygnałowymi. Pomoże mi w tym mikrokontroler F2833X firmy Texas Instruments.

Na początek chciałbym przedstawić krótką charakterystykę procesorów sygnałowych, uwzględniając ich cechy charakterystyczne.

DSP to mikroprocesory, których architektura została zaprojektowana pod kątem efektywnego przetwarzania sygnałów cyfrowych, głównie w czasie rzeczywistym. Przetwarzanie sygnałów, to z kolei nic innego jak wykonywanie na nich różnego typu operacji matematycznych. Coraz większe wymagania stawiane choćby jakości przekazu multimedialnego są powodem wzrostu rozdzielczości sygnałów, a tym samym wzrostu ilości danych do przetworzenia. Dodatkowo, współczesne, skomplikowane algorytmy obliczeniowe (np. kompresji) pociągają za sobą dużą liczbę operacji, a gdy czas jest elementem krytycznym, to najważniejsze stają się dwa parametry: szybkość i efektywność wykonywania obliczeń. Są to dwie podstawowe cechy procesorów sygnałowych, które powodują, iż we współczesnym świecie mediów, są one jedną z najprężniej rozwijających się dziedzin elektroniki.

Procesory DSP nie są jednak monopolistami w branży przetwarzania sygnałów. Dużą konkurencją są dla nich układy logiki programowalnej (FPGA). Nie można jednoznacznie stwierdzić, która z tych dwóch technologii jest lepsza – obie mają swoje plusy jak i minusy, jednak w wielu rozwiązaniach DSP częściej współpracuje z FPGA niż z nim konkuruje.

### Cechy procesorów sygnałowych

Większość obliczeń na sygnałach cyfrowych (jak splot sygnałów, FFT itd. – rys. 1) opiera się na operacjach typu MAC tzn. *Multiply And Accumulate*. Polegają one na iteracyjnym mnożeniu liczb i zapisywaniu wyników kolejnych mnożeń. W klasycznych procesorach, mnożenie odbywało się przy pomocy instrukcji dodawania i przesuwania (analogicznie do ręcznego mnożenia w słupku), co zajmowało wiele cykli maszynowych. W procesorach DSP do tego celu przywidziano jednostkę mnożącą (potocznie nazywaną MAC), która wykonuje działania sprzętowo, a wynik operacji otrzymujemy po 1 cyklu

maszynowym. Z kolei operacje dodawania, odejmowania, logiczne (OR, AND, XOR, NOT), konwersji stało/zmiennie-przecinkowej itd. wykonuje jednostka arytmetyczna logiczna (ALU).

Przy operacjach na sygnałach, często wymagana jest precyzja obliczeń. Stąd CPU niektórych rodzin procesorów sygnałowych (zwłaszcza te, które są przeznaczone do zastosowań audio) wyposażone są w jednostkę operującą na liczbach zmiennoprzecinkowych (typu rzeczywistego). Mają one wówczas osobny zbiór instrukcji dla tej jednostki. W przypadku operacji zmiennoprzecinkowych ważna jest możliwość szybkiego przesuwania bitów w lewo/prawo (skalowanie danych). Odpowiada za to tzw. *Barrel-shifter*. Inny charakterystyczny element DSP, to rozbudowana lista rozkazów. Klasyczny zbiór instrukcji został uzupełniony o rozkazy, odpowiadające najczęściej wykorzystywanym operacjom matematycznym na przetwarzanych sygnałach.

Jak wcześniej wspomniano, wielokanałowość i duża rozdzielczość sygnałów powoduje, iż mamy do czynienia ze sporą ilością danych, które muszą być przechowywane. Dlatego też, pomimo rosnących zasobów pamięci wbudowanych w struktury mikrokontrolerów, często zachodzi konieczność stosowania pamięci zewnętrznych.

$$\text{Re } X(k) = \sum_{i=0}^{N-1} x(i) * \cos(2\pi ki / N)$$

$$\text{Im } X(k) = -\sum_{j=0}^{N-1} x(i) * \sin(2\pi ki / N)$$

$$y(i) = \sum_{j=0}^{M-1} h(j) * x(i - j)$$

Rys. 1. Operacje typu MAC wykorzystuje się np. w analizie widmowej (FFT) i w filtracji (opartej na operacji splotu)

Nr rozkazu	Fetch		Decode		Read		E	Write									
n-3	F1	F2	D1	D2	R1	R2	E	W									
n-2		F1	F2	D1	D2	R1	R2	E	W								
n-1			F1	F2	D1	D2	R1	R1	E	W							
n				F1	F2	D1	D2	R1	R2	E	W						
n+1					F1	F2	D1	D2	R1	R2	E	W					
n+2						F1	F2	D1	D2	R1	R2	E	W				
n+3							F1	F2	D1	D2	R1	R2	E	W			
n+4								F1	F2	D1	D2	R1	R2	E	W		

F1 – wystawienie adresu rozkazu, F2 – pobranie rozkazu, D1 – dekodowanie rozkazu, D2 – ustalanie adresu operandu, R1 – wystawienie adresu operandu, R2 – pobranie operandu, E – wykonanie operacji przez CPU, W – zapisanie wyniku

Rys. 2. Kolejka – podział każdej instrukcji na n faz (n=8 dla F2833X) umożliwia niezależne wykonywanie różnych faz dla kolejnych rozkazów

**Tab. 1. Charakterystyka sprzętowa i peryferia procesorów TMS320F2833x**

	F28335	F28334	F28332
Max. f taktowania [MHz]	150	150	100
Wbudowana pamięć flash	256 k×16	128 k×16	64 k×16
Wbudowana pamięć SARAM	34 k×16	34 k×16	26 k×16
Wbudowana pamięć ROM (OTP)	1 k×16	1 k×16	1 k×16
Pamięć inicjalizacji (Boot ROM)	8 k×16	8 k×16	8 k×16
16/32-bit interfejs zewnętrzny (XINTF)	Tak	Tak	Tak
6-kanalowe DMA	Tak	Tak	Tak
Wyjścia PWM	18	16	16
Wyjścia HRPWM	6	6	4
32-bit wejścia Capture lub wyjścia AUX PWM	6	4	4
32-bit QEP – liczba kanałów	2	2	2
12-bit ADC	liczba kanałów	16	16
	MSPS	12,5	12,5
	czas konwersji [ns]	80	80
Wielokanałowy, buforowany port szeregowy (McBSP)	2	2	1
SPI	1	1	1
SCI (UART)	2 (3)	2 (3)	1
CAN	2	2	2
I <sup>2</sup> C	1	1	1
We/Wy ogólnego przeznaczenia (GPIO)	88	88	88
Przerwania zewnętrzne	8	8	8
32-bit licznik	3	3	3
Watchdog	Tak	Tak	Tak
Obudowa	BGA/TQFP	BGA/TQFP	BGA/TQFP

Bardzo ważna jest szybkość dostępu do pamięci dla operacji obliczeń oraz zwielokrotnienie magistral, tak aby w jednym cyklu maszynowym procesor mógł pobierać instrukcje oraz dane, przetwarzać je i zapisywać. W konsekwencji, dzieląc rozkaz na *n* faz, procesor ma możliwość wykonywania w danym cyklu maszynowym różnych faz dla kolejnych *n* rozkazów, uzyskując tym samym wykonanie jednego rozkazu w każdym cyklu maszynowym (rys. 2). Nazywamy to przetwarzaniem potokowym lub potocznie kolejką (pipelining).

Inne ważne cechy procesorów sygnałowych to: szybka reakcja na przerwania, co z kolei wymusza niezwłoczny zapis stanu procesora i ochronę kolejki, wyspecjalizowane porty i peryferia przeznaczone do odbierania, przetwarzania i wysyłania sygnałów, wbudowane DMA, dodatkowe rejestry przeznaczone do rozbudowanych typów adresacji pośredniej itd.

Częstotliwości taktowania procesorów sygnałowych wahają się od kilkudziesięciu MHz do ponad 1 GHz. Najbardziej wydajne, wielordzeniowe DSP mają moc obliczeniową rzędu tysięcy MIPS! Dla jednostek zmiennoprzecinkowych moc obliczeniową wyrażamy we FLOPS'ach. Należy zauważyć, iż pomimo pewnych ograniczeń fizycznych (związanych z maksymalnymi częstotliwościami taktowania), nadal można zauważyć postęp miniaturyzacji. Polega ona obecnie nie tyle na zmniejszaniu wymiarów układu

scalonego co na zwiększaniu jego mocy obliczeniowej bez potrzeby zwiększania obudowy. Obecnie producenci DSP są w stanie zaferować procesor w obudowie BGA z 240 wyprowadzeniami, o powierzchni 1,5 cm<sup>2</sup>.

### Mity i rzeczywiste problemy

Ze względu na skomplikowane algorytmy i stosunkowo rozbudowaną listę rozkazów, zwykle DSP programuje się z użyciem języka C/C++. Producenci procesorów stale pracują nad efektywnością kodu wynikowego generowanego przez kompilatory. Dostarczają też biblioteki gotowych algorytmów obliczeniowych najczęściej stosowanych funkcji. Dostępne są również przykładowe programy konfigurujące procesor, obsługujące poszczególne porty itd. Wszystko to w znacznym stopniu ułatwia początkującym programistom rozpoczęcie pracy z DSP, również tym, którzy wcześniej nie programowali procesorów w C/C++.

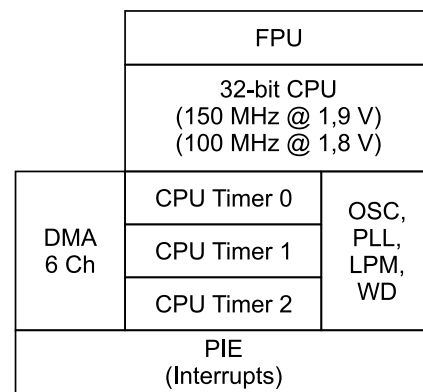
Rzeczywiste problemy są jednak innego typu – gdzie znaleźć procesor i jak go zaprogramować? Procesory sygnałowe nie są popularnymi układami i zazwyczaj trudno o ich zakup w zwykłym sklepie elektronicznym. Można je natomiast kupić u dystrybutorów danej firmy (choć często nie sprzedają układów na pojedyncze sztuki) lub w dużych sklepach internetowych. Niestety, pociąga to za sobą spore koszty (cena nieraz dwukrotnie przekracza hurtową podawaną przez producenta), w tym koszty przesyłki.

Najdroższe DSP to koszt powyżej dwóch tysięcy złotych. Mniej zaawansowane i wydajne to rząd paruset zł.

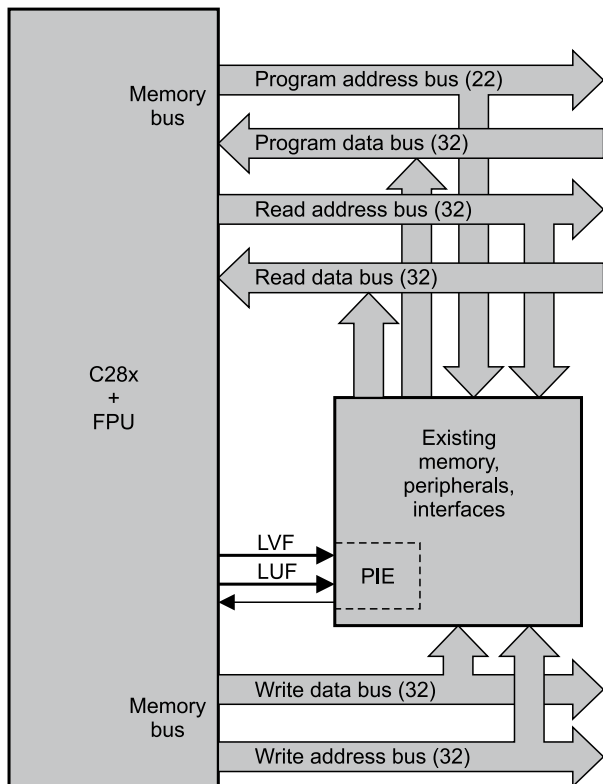
Dostępne są gotowe układy tzw. *Starter Kit* i *Evaluation Module*. Są to zestawy, które zawierają zaprojektowaną i wykonaną płytkę z danym procesorem, zaopatrzoną dodatkowo w różnego rodzaju elementy peryferyjne. Jednym z elementów zestawu jest oprogramowanie umożliwiające łatwą obsługę modułu. W zależności od modelu, płytkę może zawierać wbudowany emulator JTAG lub interfejs JTAG, który umożliwia debuggowanie i emulację w czasie rzeczywistym. Kupując taki układ właściwie mamy gotowe i przyjazne narzędzie do nauki programowania DSP.

Jedyną wadą tego typu rozwiązania jest cena. Powstaje zatem pytanie czy jesteśmy w stanie sami wykonać taką płytkę? Owszem, musimy pokonać pewne trudności, ale na pewno tak. Taki projekt był już publikowany w EP2-3/2009. Komponenty do jego budowy pozyskane były ze starego telefonu komórkowego. Owszem, do tego dochodzi spory koszt programatora i oprogramowania, jednak często wystarczające są wersje ewaluacyjne. Można też postawić na dobrą współpracę z dystrybutorami, którzy to zwykle chętnie wypożyczają sprzęt potrzebny do opracowania danej konstrukcji.

Spora część procesorów DSP jest dostępna w obudowach BGA lub TQFP. O ile można je przylutować również w warunkach warsztatowych co ma ogromne znaczenie przy budowie prototypu, to sprawdzenie czy nie powstały zwarcia w trakcie montażu jest już rzeczą niemożliwą bez JTAG'a. Poza tym obudowy BGA, przy liczbie wyprowadzeń rzędu 100, zazwyczaj wymagają stosowania płytek wielowarstwowych. Dodatkowo dochodzi problem stosowania pamięci zewnętrznych. Przy większej liczbie peryferii, możemy mieć problem ze zmieszczeniem się na dwóch warstwach płytki. Czy to wszystko oznacza, że procesory DSP są niedostępne dla osób, których nie stać na zakup gotowej płytki? Jak się okazuje – nie. Większość z tych problemów można obejść nie ponosząc żadnych kosztów.



Rys. 3. Jednostka centralna procesora F2833X



Rys. 4. Zwiłokrotniona liczba magistral: programowych i danych procesora F2833X (architektura Harvardzka)

**Procesor sygnałowy TMS320F2833x**

TMS320F2833X jest procesorem sygnałowym, który dzięki swojej architekturze i rozbudowanym peryferiom jest bardzo uniwersalnym i wydajnym komponentem, mającym liczne zastosowania, począwszy

od techniki audio, a skończywszy na przemyśle. Z kolei dzięki wbudowanym pamięciom oraz możliwości programowania przez UART, nie wymaga stosowania wielu komponentów zewnętrznych. Dzięki temu powinien zainteresować konstruktorów stosujących mikrokontrolery i zachęcić ich do próby zmierzania się z cyfrowym przetwarzaniem sygnałów.

Mikrokontrolery F2833X należą do rodziny procesorów DSP C2000 firmy Texas Instruments i są dostępne w obudowach BGA (176 lub 179 wyprowadzeń) i LQFP (176 wyprowadzeń). Skrócony wykaz ich charakterystycznych parametrów zawiera **tab. 1**.

F2833X to trzy modele: F28332, F28334 i F28335, różniące się częstotliwością taktowania, pojemnością wbudowanych pamięci

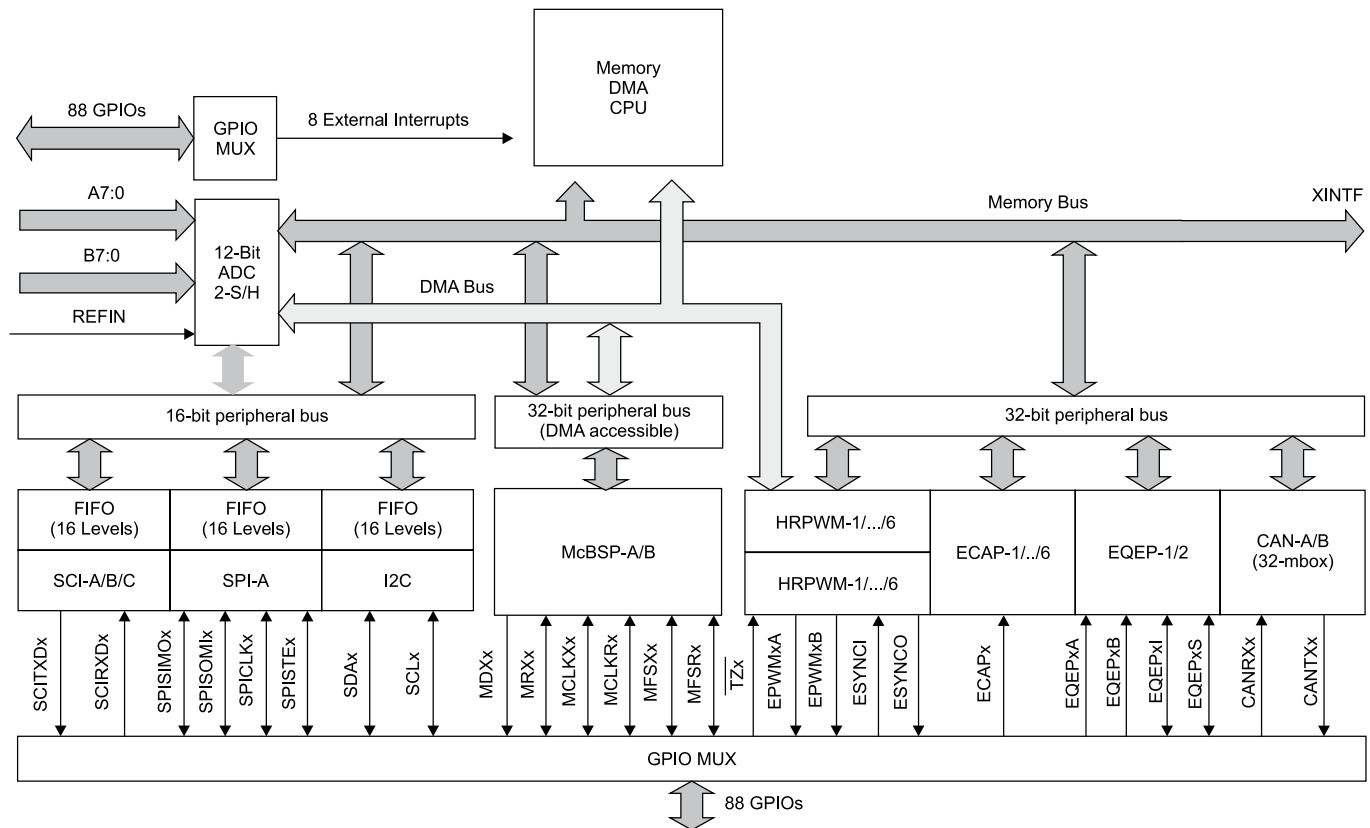
oraz liczbą portów I/O. F2833X oparty jest na 32-bitowym rdzeniu C28X zawierającym 32-bitową jednostkę arytmetyczno-logiczną (ALU), 32-bitową (lub 2×16-bitów) jednostkę mnożącą oraz 16-bitowy Barrel-shifter (**rys. 3**). Ta rodzina mikrokontrolerów wyposażona jest również w 32-bitową jednostkę

zmiennoprzecinkową. Pozwala ona na efektywną implementację w języku C/C++ algorytmów matematycznych, dając możliwość szybkiego wykonywania operacji MAC nie tylko na liczbach całkowitych, ale również rzeczywistych o wysokiej rozdzielczości. Dzięki temu, mikrokontroler ten może być skutecznie wykorzystywany w różnego rodzaju aplikacjach audio wysokiej jakości np. przy realizacji cyfrowego equalizera, układu generującego efekty dźwiękowych itd. CPU wspomagany jest także przez sprzętowy mechanizm umożliwiający debugowanie i emulację programu.

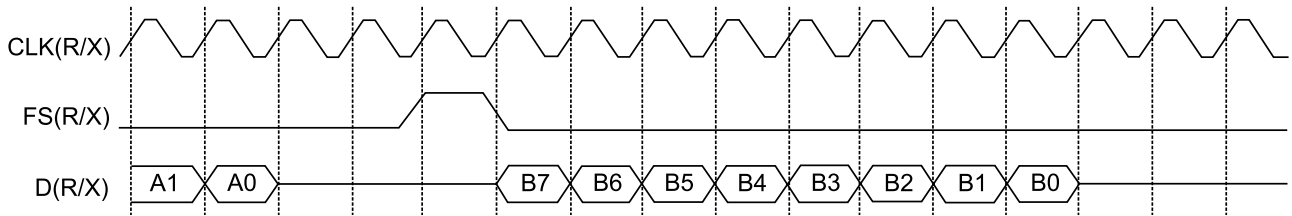
Pomimo iż F28335 posiada stosunkowo dużą liczbę rozkazów, między innymi dedykowanych przetwarzaniu sygnałów, to jest on opisany jako procesor typu RISC (*Reduced Instruction Set Computers*). Oprócz standardowych instrukcji, rodzina F2833X ma osobną listę rozkazów przeznaczonych dla jednostki zmiennoprzecinkowej.

Procesor ma 6 niezależnych magistral (architektura Harvard skorygowany) (**rys. 4**). Zwiłokrotnienie magistral pozwala na potokowe przetwarzanie rozkazów (8-fazowa kolejka), w wyniku czego większość z nich jest wykonywana w trakcie jednego cyklu procesora. 32-bitowe magistrale danych pozwalają na obsługę pamięci o pojemności 4 M×16-bit. Pamięć programu to z kolei maksimum 4 M×16-bit.

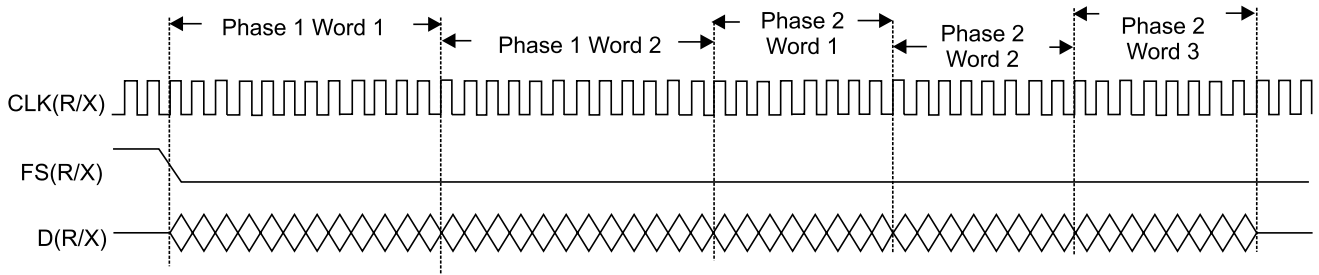
Do komunikacji pomiędzy jednostką centralną a peryferiami służą magistrale peryferyjne (16- lub 32-bitowe), które są również połączone z 6-kanalowym DMA. Dzięki niemu peryferia niewymagające interwencji



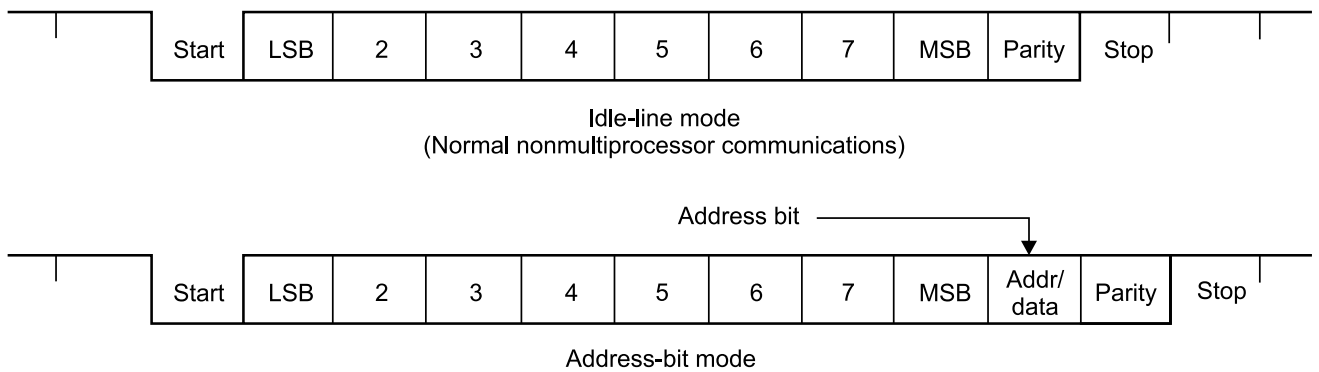
Rys. 5. Peryferia procesora F2833X



Rys. 6. Port McBSP operuje na trzech wyprowadzeniach: danych, zegara i ramki



Rys. 7. McBSP pozwala dzielić ramki na dwie fazy: długości słów i ich liczbie



Rys. 8. Ramka portu SCI

jednostki centralnej mogą bezpośrednio komunikować się z pamięcią.

Pamięć w procesorach F2833x jest pamięcią 16-bitową (większość instrukcji procesora jest 16-bitowa). Wbudowana pamięć Flash umożliwia przechowywanie zarówno instrukcji programu jak i danych. Należy zatem świadomie kontrolować ten podział w celu uniknięcia konfliktów oraz błędów polegających na wykonywaniu 'szybkich' algorytmów z użyciem pamięci Flash. Mikrokontroler posiada również wbudowaną pamięć SARAM przeznaczoną do przechowywania danych wykorzystywanych w bieżących instrukcjach.

**Peryferia**

Trudno jest dokładnie opisać wszystkie peryferia procesora F2833X w jednym artykule. Postaram się zatem przedstawić wybrane i najciekawsze w ogólnym zarysie, a wszystkich zainteresowanych szczegółami Czytelników odsyłam do obszernych not aplikacyjnych dostępnych na stronie producenta. Schemat blokowy układów peryferyjnych pokazano na rys. 5.

**McBSP (Multichannel Buffered Serial Port)**

Do najciekawszych, a zarazem najbardziej przydatnych peryferii F2833X, należy synchroniczny, wielokanałowy (wieloram-

kowy) port szeregowy McBSP (komunikacja full duplex). Port podzielony jest na odbornik (receiver) i nadajnik (transmitter) mające po 3 wyprowadzenia (rys. 6).

Dane (D), poukładane są w słowa o programowalnej długości (8, 12, 16, 20, 24 lub 32 bity). Słowa formowane są w ramki wyzwalane przez sygnał ramki. Szybkość odbierania/nadawania nie powinna być większa niż 25 MHz i musi być co najmniej dwa razy mniejsza od częstotliwości taktowania jednostki centralnej.

Sygnał ramki (FS) to impuls, którego zbocze opadające rozpoczyna odbiór/transmisję danej ramki. Impuls ten może być doprowadzony do pinu procesora lub pochodzić z wewnętrznego PLL. Długość trwania impulsu oraz jego częstotliwość można regulować w szerokim zakresie. McBSP pozwala skonfigurować dwie fazy dla każdej ramki. Przykładowo pierwsza faza ramki może zawierać 2 słowa 16-bitowe, a druga 10 słów 8-bitowych (rys. 7).

Sygnał zegarowy (CLK) może być niezależny lub wspólny dla nadajnika i odbornika, ale można również taktować nadajnik sygnałem zegarowym odebrany i odtwarzany przez odbornik i odwrotnie. Źródło sygnału zegarowego może być zewnętrzne lub pochodzić z wewnętrznego PLL. Procesor ma wbudowany *Sample Rate Generator*, który

umożliwia dobór częstotliwości pracy w dość dużym zakresie. Przykładowo port może odbierać sygnał o danej częstotliwości  $f_s$ , a po dokonaniu operacji nadpróbkowania, wysłać zmieniony sygnał z częstotliwością  $4 \times f_s$ .

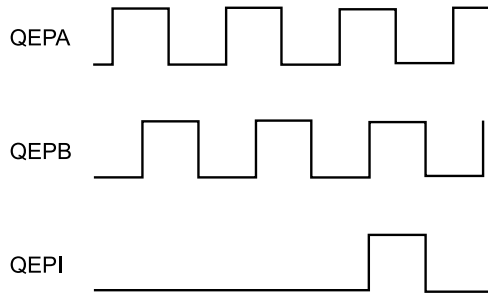
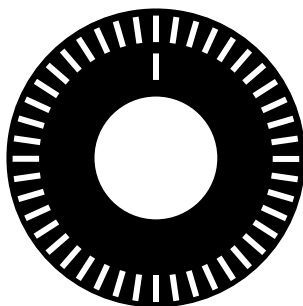
**SCI (Serial Communication Interface)**

Kolejnym przydatnym elementem opisywanego DSP jest szeregowy, asynchroniczny interfejs, znany jako UART. Modele F28334 i F28335 posiadają po 3 porty SCI (A,B i C), przy czym port SCI-C nie może być wykorzystany przez projektanta. Każdy z portów SCI posiada po 2 linie przesyłowe: nadawczą i odbiorczą, które mogą operować w konfiguracji full lub half-duplex. Dane kodowane są

R E K L A M A







Rys. 9. Optyczny enkoder Kwadraturowy

w standardzie non-return-to-zero i są chronione przez 16-bitowe bufor FIFO. Istnieje możliwość zaprogramowania 64 tys. różnych szybkości transmisji, uzależnionych od częstotliwości taktowania zegara systemowego. Istnieje również tryb automatycznej, sprzętowej detekcji szybkości transmisji. Ramka rozpoczyna się bitem startu, a kończy bitem stopu, poprzedzonym bitem parzystości oraz bitem adresu (opcjonalnie). Długość słowa jest programowalna od 1 do 8 bitów. Wygląd kompletnej ramki pokazano na rys. 8.

Port SCI-A może być skonfigurowany jako port rozruchowy, co pozwala na zaprogramowanie procesora bezpośrednio z portu RS232 komputera.

**Moduł QEP (enkoder kwadraturowy)**

Sygnal kwadraturowy określa położenie, kierunek oraz szybkość obrotu danego elementu obrotowego. Moduł QEP w procesorach F28335 (rys. 9) jest dekodery sygnału kwadraturowego pochodzącego z enkodera. Enkodery kwadraturowe mają bardzo szerokie zastosowanie, począwszy od urządzeń domowego użytku (np. myszki komputerowe czy potencjometry), a skończywszy na układach sterowania i pozycjonowania w robotyce.

**ADC (przetwornik analogowo-cyfrowy)**

Procesor F28335 ma dwa, 8-kanałowe, 12-bitowe przetworniki A/C współpracujące z modułem PWM. Funkcja auto-sekwencji pozwala na nadpróbkowanie uzyskanego sygnału wejściowego umożliwiając tym samym zwiększenie rozdzielczości przetwornika. Analogowy sygnał wejściowy powinien zawierać się w zakresie 0...3 V, co daje sygnał cyfrowy o wartości:

$$Wy[bin] = 4096 \frac{We[V] - ADCLO[V]}{3}$$

gdzie ADCLO jest wejściem odniesienia (podłączonym do masy analogowej). Przetwornik taktowany jest sygnałem o częstotliwości 12,5 MHz i pozwala na konwersję z prędkością 6,25 MSPS.

**External interface**

EI umożliwia podłączenie do procesora zewnętrznego interfejsu. Przeznaczono do

tego celu 16/32-bit magistralę danych oraz zarezerwowano 3 obszary pamięci w przestrzeni adresowej. Każdy z obszarów posiada własne wyprowadzenie chip select. Istnieje także możliwość konfiguracji czasu oczekiwania, ustalania i podtrzymania niezależnie dla odczytu i zapisu danych.

Oprócz wymienionych wyżej, w budowie mikrokontrolera można wyróżnić znane i często stosowane moduły: SPI, PWM i HRPWM (PWM o dużej rozdzielczości), CAN, CAP (do pomiaru czasu trwania i częstotliwości), liczniki 32-bitowe. Te układy periferyjne pozwalają na realizację wielu aplikacji, bez niepotrzebnego obciążania CPU.

**Programowanie procesora TMS320F2833X**

Istnieją 3 możliwości zaprogramowania procesora TMS320F2833X:

- a) Za pomocą *Code Composer Studio*, poprzez programator USB – JTAG.
- b) Za pomocą oprogramowania SD Flash, poprzez programator USB – JTAG.
- c) poprzez port RS232 komputera (ewentualnie przejściówkę USB na RS232) i asynchroniczny port procesora SCI-A, również przy użyciu oprogramowania SD Flash.

*Code Composer Studio* jest oprogramowaniem narzędziowym firmy Texas Instruments. Nawet skrócony opis jego funkcji to temat na osobny artykuł. Sporo informacji na ten temat można znaleźć w cyklu artykułów nt. mikrokontrolera Piccolo, publikowanych od EP8/2009. Tu warto jedynie wspomnieć, iż istnieje możliwość pobrania darmowej, 120 dniowej wersji CCS ze strony producenta. Umieściliśmy ją również na płycie CD\_EP8/2009A.

CCS wykorzystywane jest między innymi do pisania programów w językach C/C++ i assemblerze, emulacji, debugowania, programowania, wizualizacji itd. Zawiera również wiele użytecznych bibliotek funkcji gotowych do zastosowania. Alternatywą jest darmowe oprogramowanie firmy *Spectrum Digital*, które jest bez porównania dużo

uboższym narzędziem, jednak w zupełności zaspokoi potrzeby nie tylko początkujących programistów. Jego największą zaletą jest możliwość wspomnianego programowania procesora poprzez zwykły port szeregowy. Wadą takiego rozwiązania jest brak możliwości emulacji procesora. Zaletą – niemal zerowe koszty. Potrzebny programator wykonujemy z dwóch przewodów zasilających, układu dopasowującego napięcia (z 5 V na 3,3 V, np. MAX3232). Do połączenia wymagane są dwa przewody: nadawczy (TX) i odbiorczy (RX). Sterowniki oraz dokładny opis programowania dostępne są na stronie Spectrum Digital.

Z kolei Texas Instruments oferuje pliki API, pliki nagłówkowe, konfiguracyjne, biblioteki zoptymalizowanych funkcji matematycznych (w tym dla jednostki zmienno-przecinkowej), przykłady wykorzystania poszczególnych portów oraz inne przydatne algorytmy. Dostępne są również schematy zestawów ewaluacyjnych.

**Podsumowanie**

Obecnie procesory F2833X nie są dostępne w postaci darmowych próbek. Koszt pojedynczego układu w sklepach internetowych to około 120...160 złotych. Myśląc o praktycznym ich zastosowaniu musimy również uwzględnić koszty wykonania płytki. Wbudowany Flash oraz zwielokrotnienie wyprowadzeń poszczególnych portów umożliwiają zaprojektowanie całego układu na płycie dwustronnej. Przyłutowanie procesora możemy zlecić profesjonalnej firmie lub zrobić to samodzielnie pod warunkiem, iż mamy dostęp do odpowiedniego wyposażenia.

Niewątpliwie rozwiązanie polegające na zaprojektowaniu własnej płytki pod DSP jest zajęciem bardzo czasochłonnym, szczególnie dla mniej doświadczonych osób. Ma to jednak wiele plusów. Mianowicie, możemy sami zdecydować o tym, pod kątem jakich zastosowań będziemy optymalizować nasz układ. Projektowanie całego schematu z pewnością przybliży nam architekturę procesora, której znajomość jest rzeczą niezbędną przy jego programowaniu.

Procesory DSP to ambitne wyzwanie, wymagające sporego zaangażowania i wiedzy. Jednak to do nich należy najbliższa przyszłość elektroniki i jeżeli nie chcemy pozostać w tyle, to warto poświęcić im czas.

**Roman Bogusz**  
roman.bogusz@gmail.com  
**Piotr Pietrzyk**  
P.Pietrzyk@ieee.org