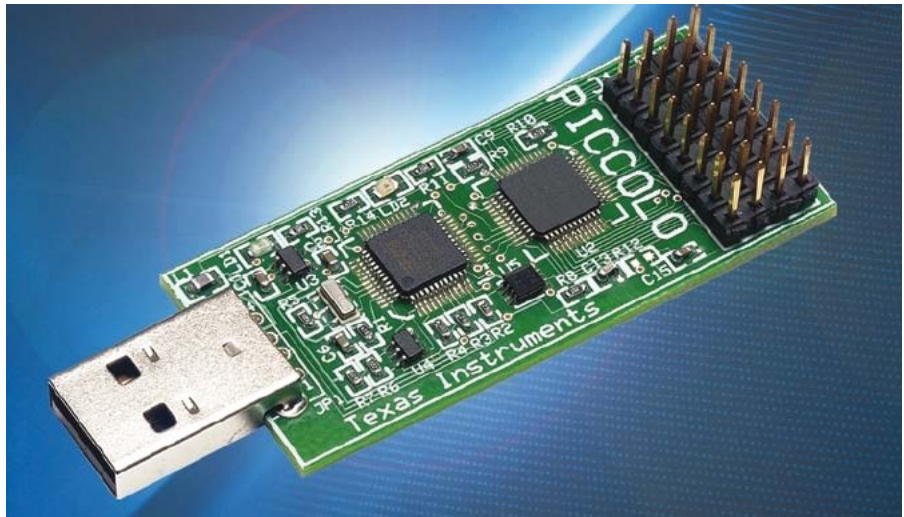


# Mocarny Maluch

## Piccolo – najmniejszy DSP z Texasa



*Domeną „rasowych” procesorów sygnałowych są zazwyczaj zaawansowane aplikacje, które wymagają dużych mocy obliczeniowych. Jednak czasem również aplikacje o mocno ograniczonym budżecie muszą wykonywać dużą liczbę skomplikowanych operacji matematycznych. Z myślą o nich Texas Instruments oferuje mikrokontrolery Piccolo. Artykuł zawiera informacje, jak rozpocząć pracę z mikrokontrolerami Piccolo w oparciu o tani zestaw startowy controlSTICK.*



Pierwszą reakcją firm na zapotrzebowanie rynku z pogranicza procesorów sygnałowych i mikrokontrolerów było rozpoczęcie sprzedaży mikrokontrolerów sygnałowych (DSC – Digital Signal Controller), czyli hybryd procesora sygnałowego i mikrokontrolera. Cechą charakterystyczną DSC jest to, że mają one więcej atrybutów znanych ze

świata mikrokontrolerów niż procesorów sygnałowych.

Firma Texas Instruments poszła o jeden krok dalej i opracowała rodzinę bardzo tanich, a więc mocno uproszczonych, procesorów sygnałowych TMS320C2000. W nomenklaturze TI są one zwane *mikrokontrolerami czasu rzeczywistego* i tego określenia będziemy używać. Mikrokontrolery C2000 dzielą się na trzy rodziny, wszystkie przedstawiono w **tab. 1**.

### Zestaw startowy controlSTICK

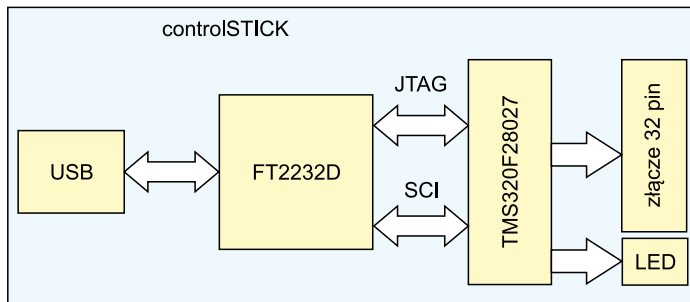
Zestaw startowy controlSTICK ma rozmiary niewiele większe od klasycznego pendrive'a, jednak pomimo tego jest całkiem poważnym narzędziem, które umożliwia dogłębne zapoznanie się z rodziną mikrokontrolerów TMS320C2000. Na płytce controlSTICK został zamontowany mikrokontroler czasu rzeczywistego TMS320F28027 należący do rodziny Piccolo. Cena zestawu jest dość atrakcyjna i wynosi 39 \$ (ok. 130 złotych).

Urządzenie podłączone jest do komputera przez port USB i nie wymaga dodatkowego zasilania. Do sprawdzania poprawności działania i testowania aplikacji na płytce umieszczono 32-doprowadzeniowe złącze szpilkowe z wyprowadzonymi najważniejszymi sygnałami (typu ADC, GPIO, itd.). Mikrokontroler może być programowany i debugowany bezpośrednio przez port USB z poziomu środowiska Code Composer Studio.

Schemat blokowy zestawu controlSTICK przedstawiono na **rys. 1**. Port USB obsługiwany jest przez układ FT2232D, czyli podwójny konwerter USB <=> UART/FIFO. Zadaniem układu FT2232D jest wymiana informacji z mikrokontrolerem za pomocą interfejsu JTAG i opcjonalnie UART (SCI). Pierwszy służy do programowania i debugowania, natomiast drugi może być wykorzystany w tworzonej przez twórcę aplikacji. Do prostej sygnalizacji stanu mikrokontrolera może służyć czerwona dioda LED.

**Tab. 1. Mikrokontrolery z rodziny C2000**

Delfino	TMS320C2834x, zmiennoprzecinkowy
	Efektywna jednostka mogąca pracować z zegarem do 300 MHz (600 MFLOPS) Pamięć Flash do 512 kB, 6-kanalowy DMA, PWM o rozdzielczości 65 ps
	TMS320F2833x, zmiennoprzecinkowy
Piccolo	Maksymalna częstotliwość pracy do 150 MHz, do 512 kB Flash, do 68 kB RAM, 6-kanalowy DMA
	TMS320F2802x, stałoprzecinkowy
	Dostępne w małych, 38-wyprowadzeniowych obudowach. Maksymalna częstotliwość pracy 40...60 MHz, pamięć Flash do 64 kB, ADC 4,64 MSPS, komparator analogowy, ePWM o rozdzielczości 150 ps, I <sup>2</sup> C, SPI, SCI
F28x	TMS320F2803x, stałoprzecinkowy
	Częstotliwość zegara do 60 MHz, 128 kB pamięci Flash, obudowy 64- lub 80-wyprowadzeniowe. W stosunku do 2802x dodano CLA, QEP oraz interfejsy CAN i LIN
	TMS320F2823x, stałoprzecinkowy
	Jest to stałoprzecinkowa wersja F2833x kompatybilna z nim pod względem wyprowadzeń; jedyna różnica polega na zastosowaniu jednostki stałoprzecinkowej w miejsce zmiennoprzecinkowej
	TMS320F280x, stałoprzecinkowy
Maksymalna częstotliwość 60...100 MHz. ADC 12,5 MSPS; wielokrotne, niezależne PWM; eQEP (quadrature encoder pins), eCAP (event captures), do 256 kB pamięci Flash	
TMS320F281x, stałoprzecinkowy	
Częstotliwość pracy do 150 MHz. ADC 12,5 MSPS, 56 GPIO, SPI, CAN, UART	

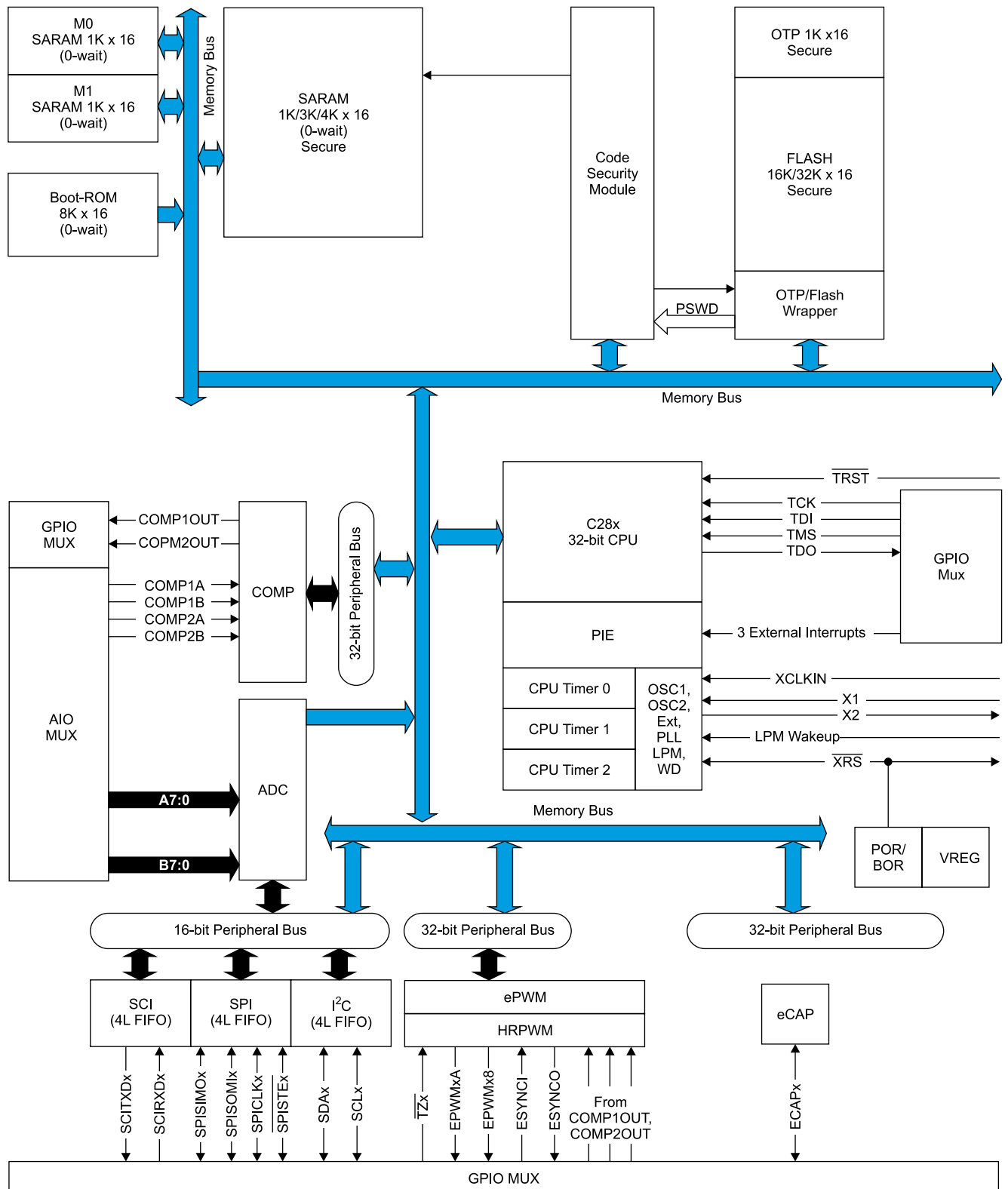


Rys. 1. Schemat blokowy zestawu controlSTICK

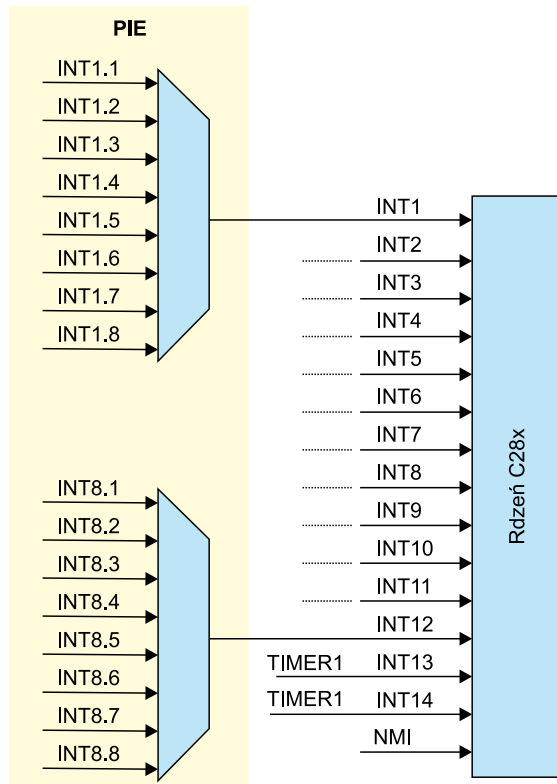
W zestawie otrzymujemy wszystko, co jest niezbędne do rozpoczęcia pracy z nową rodziną mikrokontrolerów. W pudełku, poza płytą controlSTICK, znajdziemy płytę CD z pakietem Code Composer Studio i przykładowymi aplikacjami oraz kabel USB i kilka zwrotek.

### Rdzeń TMS320C28x

Schemat blokowy mikrokontrolera z rodziny Piccolo przedstawiono na rys. 2. Mikrokontrolery z rodziny Piccolo wyposażone



Rys. 2. Schemat blokowy mikrokontrolera z rodziny Piccolo



Rys. 3. System przerwań w mikrokontrolerze TMS320F28027

są w rdzeń C28x – jest to stałoprzecinkowa, 32-bitowa jednostka typu RISC. Architektura ma zbliżoną do konstrukcji rdzeni standardowych mikrokontrolerów, z tym że rdzeń C28x zoptymalizowano i wyposażono w dodatkowe mechanizmy i peryferie usprawniające cyfrowe przetwarzanie sygnałów oraz efektywne wykonywanie kodu napisanego w języku C/C++ i Asemblerze.

Do przeprowadzania szybkich i złożonych operacji mnożenia rdzeń C28x wyposażono w sprzętowy, 32-bitowy układ mnożący. Umożliwia wykonywanie sprzętowego mnożenia liczb w kodzie uzupełnionym do 2, przy czym wynik jest 64-bitowy. Można mnożyć liczby we wszystkich kombinacjach, czyli dwie liczby ze znakiem, jedna ze znakiem, a druga bez oraz, rzecz jasna, dwie liczby bez znaku.

Wbudowane w rdzeń C28x układy debugowania znacznie upraszczają i przyspieszają proces usuwania błędów oprogramowania oraz optymalizację działania aplikacji.

### Mikrokontroler TMS320F28027

Aktualnie do rodziny mikrokontrolerów Piccolo należą trzy układy. Układ TMS320F28027 jest przedstawicielem rodziny Piccolo o średnich możliwościach. Producent zachował równowagę pomiędzy rozmiarem obudowy i ceną z jednej strony a rozmiarem pamięci i wyposażeniem w układy peryferyjne z drugiej.

Schemat blokowy mikrokontrolera F28027 przedstawiono na rys. 2. Jest to

architektura harwardzka, co oznacza, że magistrale: pamięci programu i danych, są rozdzielone. Uzyskuje się dzięki temu znaczne przyspieszenie pracy w stosunku do układów o architekturze von Neumanna.

Maksymalna częstotliwość zegara systemowego, z jaką może pracować układ F28027, wynosi 60 MHz. Napięcie zasilania jest jak na dzisiejsze czasy typowe i równe 3,3 V. Wbudowane układy resetu zapewniają pewny start systemu podczas włączania zasilania, a układy watchdog, jeśli tylko zostaną włączone, pilnują, czy program jest wykonywany przez mikrokontroler poprawnie.

Mikrokontrolery Piccolo wyposażono w podstawowy zestaw komunikacyjnych układów peryferyjnych. Do wykorzystania są sprzętowe kontrolery I<sup>2</sup>C, SPI i SCI. O ile dwa pierwsze akronimy są konstruktorem systemów mikroprocesorowych doskonale znane,

to skrót SCI może wymagać wyjaśnienia. Szeregowy interfejs komunikacyjny, czyli SCI (*Serial Communication Interface*), jest analogiczny w działaniu do układów UART, jakie ma większość mikrokontrolerów dostępnych na rynku. Ramka danych kontrolera SCI jest w pełni konfigurowalna – oprócz ustawiania bitów startu, stopu i parzystości można ustalać długość ramki w zakresie od 1 do 8 bitów.

Do realizacji zadań związanych z odmierzeniem czasu lub zliczaniem impulsów mogą zostać użyte trzy 32-bitowe timery liczące w dół, z 16-bitowymi preskalarami. Preskalary te dzielą częstotliwość sygnału taktującego rdzeń. Za każdym razem, kiedy licznik osiągnie zero, może być generowane przerwanie.

Na szczególną uwagę zasługują układy PWM i przetworników analogowo-cyfrowych. Wbudowany ADC jest 13-kanalowym przetwornikiem o rozdzielczości 12 bitów. Minimalny czas konwersji wynosi tylko 217 ns, czyli szybkość pracy wynosi około 4,6 MSPS. 12 z 13 kanałów podzielonych jest na grupy po dwa kanały, które mogą być przetwarzane jednocześnie. Przetwarzanie może być wyzwalone z wie-

#### Jądro systemu DSP/BIOS

DSP/BIOS jest skalownym jądrem systemu czasu rzeczywistego, który pozwala na znaczne przyspieszenie w uruchamianiu aplikacji DSP z przeznaczeniem na układy TMS320C2000, TMS320C5000, TMS320C6000. Ponadto dostępne funkcje API ułatwiają przenoszenie aplikacji pomiędzy różnymi układami.

#### Automatyzacja pracy z CCStudio

Pracując z programami ładowanymi do pamięci RAM mikrokontrolera, można znacznie przyspieszyć pracę poprzez włączenie opcji ładowania programu automatycznie po zbudowaniu projektu. W tym celu należy przejść do menu *Option/Customize* i na zakładce *Program/Project/CIO* okna, które się pojawi, należy zaznaczyć opcję *Load Program After Build*. Dzięki temu, przy każdym zbudowaniu projektu, jeśli tylko jest wybrana opcja ładowania do pamięci RAM, program zostanie automatycznie przesłany do mikrokontrolera.

lu źródeł m.in: programowo, generatorem PWM, przerwaniami zewnętrznymi lub też za pomocą timerów. Jak wynika z powyższych informacji przetwornik analogowo-cyfrowy, w jaki są wyposażone mikrokontrolery Piccolo, może okazać się w zupełności wystarczający dla mało i średnio wymagających aplikacji.

Bardzo często do sterowania urządzeniami zewnętrznymi wykorzystuje się sygnały PWM. Z myślą o takich aplikacjach w mikrokontrolerach F28027 jest dostępny sprzętowy generator o nazwie ePWM (*enhanced PWM*), czyli o zwiększonych możliwościach w stosunku do standardowych rozwiązań. Mikrokontroler dysponuje czterema modułami ePWM, co w efekcie daje 8 wyjść sygnału. Dodatkowo, konstruktor ma do dyspozycji układ HRPWM (*High Resolution PWM*), który pozwala na uzyskanie sygnału o zwiększonej rozdzielczości w stosunku do wymienionych wcześniej modułów ePWM.

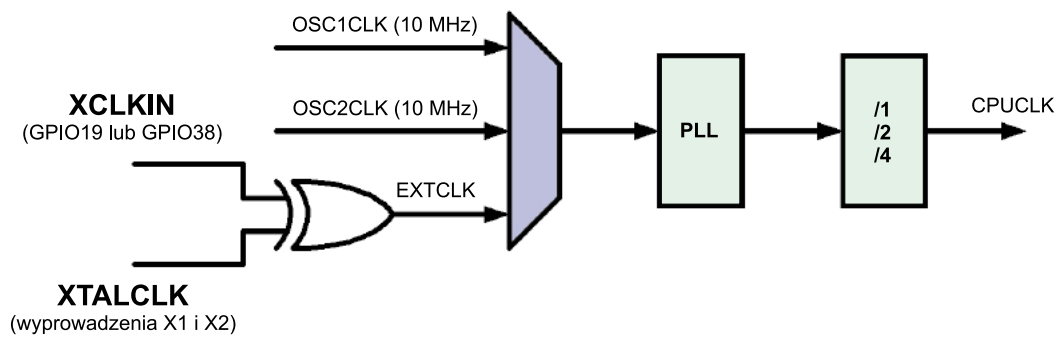
### Przerwania

W sumie, wraz z przerwaniami niemaszkalnym NMI, rdzeń C28x obsługuje 17 źródeł przerwań. Każdy współczesny mikrokontroler może współpracować z większą liczbą źródeł wyjątków. Pracę z dużą licznější grupą wyjątków umożliwia kontroler PIE (*Peripheral Interrupt Expansion*), którego zadaniem jest multipleksowanie przerwań.

Wszystkie przerwania, jakie są dostępne w mikrokontrolerze TMS320F28027, zostały podzielone na 12 grup, po 8 przerwań każda. Rozwiązanie to przedstawiono na rys. 3. Z rysunku wynika, że teoretycznie mikrokontrolery Piccolo mogą obsługiwać 96 przerwań, jednak w rzeczywistości F28027 może obsłużyć 22 przerwania, w tym 3 zewnętrzne. Każde z przerwań zewnętrznych jest konfigurowalne, co oznacza, że może być włączone lub wyłączone oraz że może być ustalone zboczne, na które ma reagować mikrokontroler.

### Zasilanie i tryby obniżonego poboru mocy

Mikrokontroler TMS320F28027 ma trzy rodzaje wyprowadzeń służących do zasilania układu. Są to: wyprowadzenia dla



Rys. 4. Schemat połączenia sygnałów zegarowych dla CPU

napięcia zasilania części analogowej  $V_{DDA}$ , części cyfrowej  $V_{DDIO}$  oraz rdzenia  $V_{DD}$ .

Napięcia  $V_{DDA}$  i  $V_{DD}$  powinny mieć wartość 3,3 V, natomiast  $V_{DD}$  jest zależne od konfiguracji. Do poprawnej pracy rdzeń TMS **TMS320C28x** wymaga napięcia zasilania o wartości 1,8 V. Można je do rdzenia dostarczyć na dwa sposoby. Pierwszy polega na dołączeniu do wyprowadzeń  $V_{DD}$  napięcia o takiej właśnie wartości. W celu wymuszenia tego trybu pracy należy wyprowadzenie VREGENZ podłączyć do dodatkowego bieguna zasilania. Drugi sposób to wykorzystanie wewnętrznego stabilizatora VREG, który sam troszczy się o właściwe zasilanie rdzenia mikrokontrolera. W tym ostatnim przypadku, do wyprowadzeń  $V_{DD}$  nie podłącza się już żadnego napięcia, jednak pomimo tego każda nóżka  $V_{DD}$  musi być wyposażona w kondensator o pojemności rzędu 1,2  $\mu\text{F}$  umieszczony tak blisko wyprowadzeń mikrokontrolera, jak to tylko możliwe.

W aplikacjach energooszczędnych, w których pobór energii jest sprawą kluczową, można wykorzystać tryby pracy o obniżonym poborze mocy. Mikrokontrolery Piccolo mogą pracować w jednym z trzech trybów energooszczędnych:

- IDLE – w trybie tym rdzeń wprowadzany jest w reżym *low-power*, natomiast sygnał zegarowy urządzeń peryferyjnych może być włączany selektywnie. Dzięki tej ostatniej możliwości, w systemie mogą pozostać w stanie aktywnej pracy tylko te peryferie, których praca jest niezbędna. Wybudzenie z trybu IDLE następuje pod wpływem przerwania od urządzenia peryferyjnego lub watchdoga.
- STANDBY – aktywowanie tego trybu powoduje wyłączenie rdzenia i układów peryferyjnych, natomiast nadal pracują pętla synchronizacji fazowej oraz oscylatory. Powrót do normalnego wykonywania programu może nastąpić po wystąpieniu przerwania zewnętrznego.
- HALT – mikrokontroler w tym trybie pobiera najmniejszą ilość energii ze

źródła zasilania z wszystkich trzech trybów obniżonego poboru mocy. Jest to okupione wyłączeniem wszystkich elementów systemu. Możliwe jest jednak takie skonfigurowanie układu, aby po wprowadzeniu trybu HALT nadal pracował wewnętrzny oscylator. Może on być wykorzystany do taktowania watchdoga, a ten z kolei może okresowo wybudzać mikrokontroler. Ponadto, powrót do normalnej pracy zapewni m.in. sygnał zerowania – Reset.

### System zegarowy i sygnały taktowania

Sygnał zegarowy w mikrokontrolerach **TMS320F28027** może pochodzić z trzech źródeł: z wewnętrznych oscylatorów, zewnętrznego rezonatora oraz z zewnętrznego

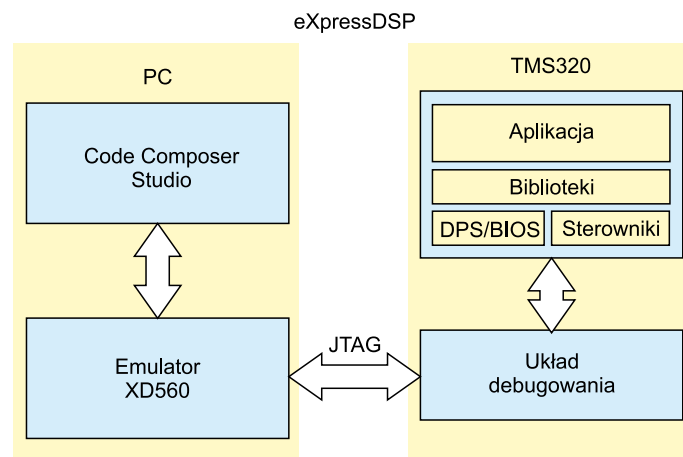
generatora. Schemat połączenia sygnałów zegarowych dla CPU przedstawiono na **rys. 4**. Stosując pierwszą opcję, konstruktor ma do wyboru dwa wewnętrzne oscylatory – INTOSC1 oraz INTOSC2, obydwa o częstotliwości 10 MHz. Mikrokontroler można tak skonfigurować, aby rdzeń był taktowany z jednego oscylatora, natomiast watchdog z drugiego. Znakomicie wpływa to na niezawodność systemu i jego bezpieczeństwo pracy.

Systemowy sygnał zegarowy może być powielony za pomocą układu PLL. Jeśli system ma pracować z zewnętrznym rezonatorem, to należy go podłączyć do wyprowadzeń X1 oraz X2 (nóżki 45 i 46).

Układy Piccolo wyposażono w system detekcji nieprawidłowego sygnału zegarowego lub też całkowitego jego zaniku. Jeśli mikrokontroler wykorzystuje do pracy układ PLL, a źródło sygnału zegarowego przestanie generować sygnał taktujący, to może wtedy nastąpić restart mikrokontrolera.

### Środowisko Code Composer Studio

Każda realizacja projektu wbudowanego z udziałem mikrokontrolera czy procesora sygnałowego składa się zazwyczaj z czterech elementarnych etapów: planowanie, pisanie



Rys. 5. Uproszczony schemat blokowy eXpressDSP

Tab. 2. Biblioteki dostarczane z pakietem Code Composer Studio

C28x Communications Driver Library	Zawiera moduły obsługi szeregowych pamięci EEPROM za pomocą I <sup>2</sup> C i SPI oraz programowy interfejs I <sup>2</sup> C
C28x Fast Fourier Transforms Library	Zawiera funkcje wyznaczające szybką transformatę Fouriera sygnału wejściowego (FFT)
C28x Filter Library	Zawiera funkcje służące do realizacji filtrów FIR i IIR
C28x Fixed-Point Math Library	Zawiera funkcje realizujące stałoprzecinkowe operacje matematyczne
C28x Signal Generator Library	Funkcje służące do generowania sygnałów o różnych kształtach
C28x Software Test Bench (STB) Library	Ułatwia optymalizowanie algorytmów DSP
C28x IQMath Library	Zawiera funkcje umożliwiające efektywne wykonywanie obliczeń zmienneprzecinkowych w stałoprzecinkowych rdzeniach C28x

kodu, usuwanie błędów, testy i optymalizacja. Firma Texas Instruments postanowiła maksymalnie uprościć i przyspieszyć każdy z powyższych etapów, a efektem tych działań jest platforma **eXpressDSP**. Jej uproszczony schemat blokowy umieszczono na **rys. 5**.

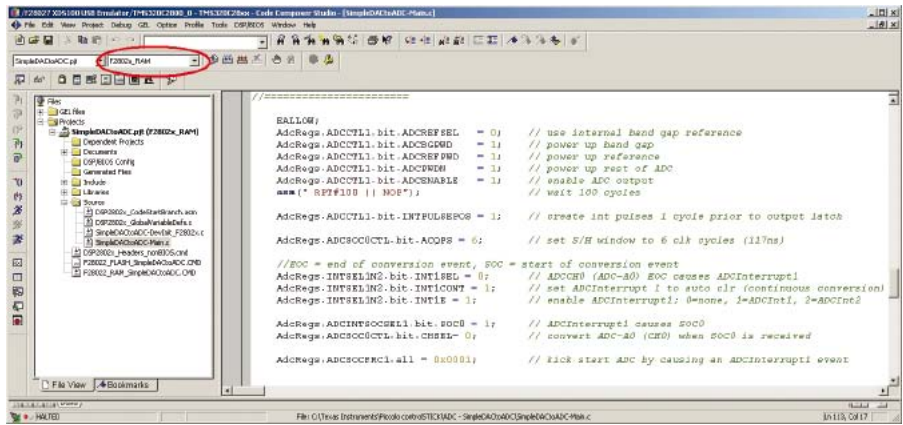
Celem stworzenia eXpressDSP było dostarczenie konstruktorom narzędzia pozwalającego na możliwie maksymalnie sprawne budowanie aplikacji DSP. System eXpressDSP obejmuje po stronie komputera (hosta) CCStudio w roli kompilatora i debugera oraz emulator XD560. Od strony DSP są to sprzętowe układy programowania i debugowania oraz biblioteki dostarczane przez TI. Dostępne biblioteki, ułatwiające realizację zadań cyfrowego przetwarzania sygnałów dla rodziny mikrokontrolerów czasu rzeczywistego Piccolo, wymieniono w **tab. 2**.

Pakiet Code Composer Studio (CCStudio) jest najważniejszym elementem platformy **eXpressDSP**, a służy do tworzenia i uruchamiania aplikacji z użyciem procesorów sygnałowych z rodzin C2000, C5000, C6000. W wersji ewaluacyjnej Code Composer Studio ma ograniczenie w wielkości kodu wynikowego do 32 KB. Ograniczenie to raczej nie jest odczuwalne w trakcie poznawania rodziny Piccolo. Zestaw controlSTICK jest w pełni zintegrowany z CCStudio, dzięki czemu, po podłączeniu zestawu startowego do komputera z zainstalowanym CCStudio, uzyskuje się w pełni sprawny system uruchomieniowy.

**Przykładowa aplikacja**

Wraz z zestawem controlSTICK otrzymujemy kilka przykładowych projektów, których działanie może być szybko przetestowane w praktyce. Wszystkie przykładowe aplikacje można również pobrać ze strony firmy TI ([www.ti.com/f28xgetstarted](http://www.ti.com/f28xgetstarted)), a ich krótki opis zamieszczono w **tab. 3**. Przedstawiona aplikacja demonstracyjna będzie do pracy wykorzystywać przetwornik analogowo-cyfrowy oraz, pośrednio, przetwarzanie cyfrowo-analogowe.

Pierwszym etapem pracy z płytką controlSTICK jest zainstalowanie i skonfigurowanie na komputerze PC środowiska Code Composer Studio. Szczegółowe informacje,



**Rys. 6. Ekran programu CCStudio**

podając krok po kroku niezbędne czynności instalacyjne, zostały przystępnie opisane w broszurce dołączonej do zestawu ewaluacyjnego oraz na stronie producenta.

Gdy proces instalacji przebiegnie prawidłowo, należy uruchomić pakiet CCStudio. Kolejnym krokiem jest otwarcie przykładowego projektu. Plik projektu znajduje się w katalogu *ADC - SimpleDACtoADC* w folderze instalacyjnym zestawu controlSTICK (domyślnie *c:\Texas Instruments*). Po otwarciu pliku *SimpleDACtoADC.pjt* (polecenie *Project/Open*) wygląd środowiska CCStudio będzie podobny do tego z **rys. 6**. Zadaniem omawianej aplikacji jest wytworzenie przebiegu PWM, który po przefiltrowaniu jest podawany na wejście przetwornika analogowo-cyfrowego.

Po lewej stronie okna CCStudio znajduje się drzewiasta struktura projektu. Z punktu widzenia pierwszej styczności z pakietem i przykładową aplikacją, najważniejsze pliki znajdują się w katalogu *Source*.

**Struktura plików**

Struktura plików jest szablonowa, przez co projekt jest przejrzysty i łatwy w nawigowaniu. Kod odpowiedzialny za podstawową konfigurację mikrokontrolera (sygnałów zegarowych, PLL, wyprowadzeń) do pracy umieszczony jest zawsze w pliku o nazwie zakończonej *DevInit\_F2802x.c*, zatem w przypadku omawianej aplikacji będzie to plik *SimpleDACtoADC-DevInit\_F2802x.c*. Właściwe zachowanie aplikacji

opisano w pliku *SimpleDACtoADC-Main.c* i tam też mieści się kod odpowiedzialny za konfigurację układów peryferyjnych wykorzystywanych w danym projekcie.

Warto zwrócić uwagę na to, że wszystkie przykładowe aplikacje dla zestawu controlSTICK przygotowano w dwóch wersjach: do umieszczenia w pamięci Flash lub w pamięci RAM. Wyboru dokonuje się za pomocą listy rozwijanej opcji konfiguracji projektu (**rys. 7**). Jeśli zostanie wybrana opcja ładowania programu do pamięci RAM, to w konsekwencji nie będzie używana pamięć Flash, która ma rzecz jasna ograniczoną liczbę cykli zapisu. Zmieniając opcję ładowania programu, zauważyć można, że niebieska strzałka przy plikach konfiguracyjnych linkera pojawia się przy pliku, który odpowiada za ładowanie programu dla danego ustawienia. W procesorach DSP program do pamięci RAM ładowany jest w aplikacjach, które wymagają bardzo dużej szybkości pracy programu (czas dostępu pamięci RAM jest znacznie krótszy niż pamięci FLASH).

**Uruchomienie aplikacji**

W celu połączenia się z płytką controlSTICK, podłączoną do komputera, w menu *Debug* wybieramy polecenie *Connect* (lub kombinacją klawiszy Alt+C). Gdy połączenie zostanie nawiązane, to w lewym dolnym rogu pojawi się zielona ikona. Proces debugowania rozpoczynamy od zbudowania projektu i załadowania programu do mikrokontrolera. Jeśli mamy w ustawieniach zaznaczoną opcję automatycznego ładowania programu oraz gdy wykorzystywana jest pamięć RAM, to po wybraniu *Project/Build* (lub naciśnięciu klawisza F7) źródło programu zostanie skompilowane, a zbiór wynikowy załadowany do pamięci mikrokontrolera.



**Rys. 7. Rozwijana lista opcji konfiguracji projektu**

Tab. 3. Wykaz przykładów programów	
Timer-BlinkingLED	Miga dioda LED
ADC-SimpleDACtoADC	Przetwarzanie A/C w trybie ciągłym filtrowanego wyjścia DAC
ADC-TriggeredADC	Wyzwalane przetwarzanie A/C
ADC-ContinuousADC	Ciągłe przetwarzanie 11 kanałów A/C dostępnych na płytce controlSTICK
PWM-AsymmetricPWM	Generowanie PWM z regulacją okresu i wypełnienia
PWM-FilteredHRPWM	Generowanie sygnału PWM o dużej rozdzielczości (generator HRPWM)
PWM-SymmetricPWM	Generowanie symetrycznego sygnału PWM
PWM-TripPWM	Wymuszanie określonego stanu na wyjściu PWM
PWM-DeadBandPWM	Wykorzystanie „czasu martwego” podczas generowania PWM
Comparator-TripComp	Zastosowanie komparatora do wymuszenia odpowiedniego stanu na wyjściu PWM
Comparator-CompareValues	Użycie komparatora do generowania sygnału przerwania dla CPU

Po załadowaniu programu można przystąpić do jego uruchomienia. Aby mieć pewność, że MCU uruchomi się w poprawny sposób, należy wykonać sekwencję czynności polegającą na wyborze odpowiednich opcji z menu *Debug*. W pierwszej kolejności, z wymienionego wyżej menu, wybieramy *Reset CPU*, następnie *Restart*, a na końcu *Go Main*. Ostatecznie, rozpoczęcie wykonywania programu odbywa się przez polecenie *Run*, co odpowiada naciśnięciu klawisza F5.

Omawiana przykładowa aplikacja wymaga jeszcze podłączenia ze sobą pinów 3 i 31, które są odpowiednio: wejściem przetwornika A/C i filtrowanym wyjściem generatora PWM. W roli filtra został tutaj użyty prosty układ RC, zatem nie można się po tym rozwiązaniu spodziewać bardzo dobrych efektów działania.

## Debugowanie

Aplikację *SimpleDACtoADC* napisano pod kątem sprawdzenia poprawności działania w trybie debugowania. Podgląd dwóch najważniejszych parametrów, a więc wypełnienia generatora PWM oraz wyniku przetwarzania analogowo-cyfrowego, jest możliwy przy użyciu okna *Watch*. Jeśli wspomniane okno jest niewidoczne, to można je włączyć z poziomu menu *View*.

Wynik przetwarzania jest zawarty w zmiennej *adc\_result*, natomiast aktualne wypełnienie generatora PWM w zmiennej *duty\_cycle*. Aby dodać obie zmienne do podglądu, wystarczy kliknąć na każdej i wybrać *Add to Watch Window*. Zmiany wypełnienia można wprowadzać przez dwukrotne kliknięcie na zmienną *duty\_cycle* i wpisanie jej nowej wartości, pamiętając, że maksymalna wartość nie może być większa niż 30, ponieważ tyle wynosi okres generowanego sygnału PWM.

Krzysztof Paprocki  
paprocki.krzysztof@gmail.com

R E K L A M A



**TWT**  
**AUTOMATYKA**

- Indukcyjne czujniki zbliżeniowe
- Czujniki optyczne
  - odbiciowe
  - refleksyjne
  - bariery
- Indukcyjne czujniki ruchu
- Sygnalizatory poślizgu

TWT s.c.  
ul. Wąflowa 1  
02-971 Warszawa  
tel./fax (22) 648 20 89  
Tel. Kom. (0) 501 777 938  
E-mail: twt@twt.com.pl  
www.twt.com.pl

**zainteresowanym wysyłamy bezpłatnie katalogi!**

# SILNIKI KROKOWE



**SANYO DENKI**

[www.sanyodenki.eu](http://www.sanyodenki.eu)

# STEROWNIKI SILNIKÓW KROKOWYCH




**SEMICON**<sup>®</sup>

[www.semicon.com.pl](http://www.semicon.com.pl)

ul. Zwoleńska 43/43a, 04 - 761 Warszawa  
tel. 022 615 73 71, 022 615 64 31  
info@semicon.com.pl [www.semicon.com.pl](http://www.semicon.com.pl)