

Migracja z Cortex-M3 do Cortex-M4

ARM Cortex-M4, to najnowszy rdzeń firmy ARM, przeznaczony do zastosowań w systemach wbudowanych. Jest szczególnie użyteczny tam, gdzie zachodzi konieczność szybkiego, cyfrowego przetwarzania sygnałów. W artykule przedstawiono cechy nowego rdzenia, porównano go z układami Cortex-M3 oraz omówiono problematykę przenoszenia kodu ze starszych układów na nowsze.

Rdzenie Cortex-M3 i Cortex-M4 mają zbliżoną budowę. W praktyce, nowszy z nich (M4) stanowi rozbudowaną wersję starszego.

Układy DSP

Jedną z podstawowych różnic pomiędzy układami z rdzeniem Cortex-M3 a Cortex-M4, jest podsystem DSP (cyfrowego przetwarzania sygnałów). Blok ten stanowi część samego rdzenia Cortex-M4 i znacząco zwiększa wydajność wykonywania takich operacji, jak np. filtrowanie sygnałów cyfrowych filtrami FIR, IIR, obliczanie szybkiej transformaty fourierowskiej (FFT), przetwarzanie strumieni multimedialnych, czy też nawet obliczenia algorytmów PID. Skrócenie czasu wykonywania tych operacji jest wyraźnie widoczne na rysunkach 3 i 4. Przykładowo, nałożenie filtru o nieskończonej odpowiedzi impulsowej zajmuje w przypadku obliczeń 16-bitowych ponad 3 razy mniej czasu układowi z rdzeniem Cortex-M4, niż układowi z rdzeniem Cortex-M3, jeśli oba są taktowane sygnałem o tej samej częstotliwości. Nowe układy pozwalają znacząco przyspieszyć przetwarzanie sygnałów cyfrowych, niezależnie od tego czy są wykony-

wane operacje na liczbach 16-, czy 32-bitowych. Co więcej, w układach z rdzeniem Cortex-M4 wszystkie instrukcje DSP są realizowane w trakcie jednego cyklu. Rdzeń Cortex-M3 nie ma podsystemu DSP, więc wymaga więcej niż jednego cyklu maszynowego, aby wykonać operacje skutkujące takim samym rezultatem. Na przykład dekodowanie strumienia MP3 w czasie rzeczywistym zrealizowane za pomocą rdzenia Cortex-M3 wymaga sygnału taktowania o częstotliwości ok. 20...25 MHz, a nowszy rdzeń Cortex-M4 wymaga zaledwie 10...12 MHz.

32-bitowy blok MAC

W rdzeniu Cortex-M4 zastosowano nowy, 32-bitowy blok MAC (*Multiply And Accumulate*), który został zoptymalizowany oraz wzbogacony o dodatkowe instrukcje. Dzięki temu jest on w stanie wykonać operację przemnożenia dwóch liczb 32 bitowych i zsumowania ich z liczbą 64 bitową (rezultat jest 64-bitowy) w trakcie jednego cyklu. Alternatywnie, możliwe jest jednoczesne wykonanie w jednym cyklu dwóch niezależnych operacji mnożenia liczb 16-bitowych.

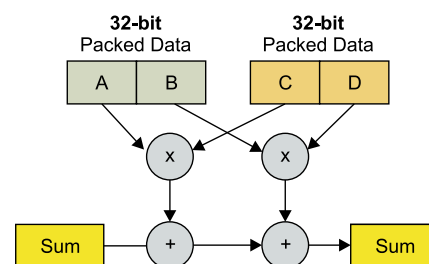
Wszystkie instrukcje realizowane przez nowy blok MAC zajmują tylko jeden cykl. Pozwalają na przeprowadzanie operacji mnożenia liczb 16- i 32-bitowych oraz na sumowania liczb 32- i 64-bitowych.

Instrukcje SIMD

Rdzeń Cortex-M4 obsługuje instrukcje typu SIMD (*Single Instruction, Multiple Data*), które nie były dostępne w poprzednich układach z rodziny Cortex-M. Do zestawu poleceń SIMD należy m.in.: dodawanie, odejmowanie, mnożenie oraz mnożenie i sumowanie, które są użyteczne przy przetwarzaniu sygnałów. Przykładowo, układy z oma-

Dodatkowe informacje:

Artykuł został udostępniony przez Farnell we współpracy z Freescale. Więcej informacji o nowych produktach jest dostępne na stronie internetowej Farnell www.farnell.com/pl oraz na portalu społecznościowym dla projektantów elektroniki www.element14.com.



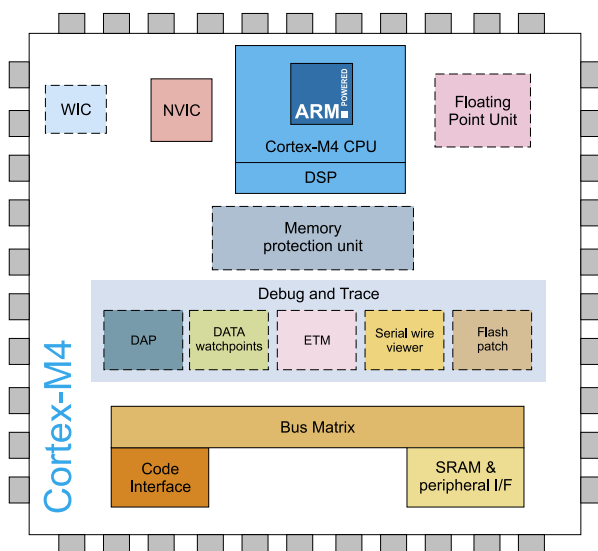
Rys. 2. Suma dwóch iloczynów niezależnych zmiennych 32-bitowych wykonuje się w jednym cyklu

wiane rodziny mogą wykonywać równocześnie w jednym cyklu cztery 8-bitowe lub dwa 16-bitowe sumowania/odejmowania. Oznacza to, że przykładowo możliwe jest obliczenie w jednym cyklu sumy jednej zmiennej oraz dwóch iloczynów, każdy dwóch niezależnych zmiennych.

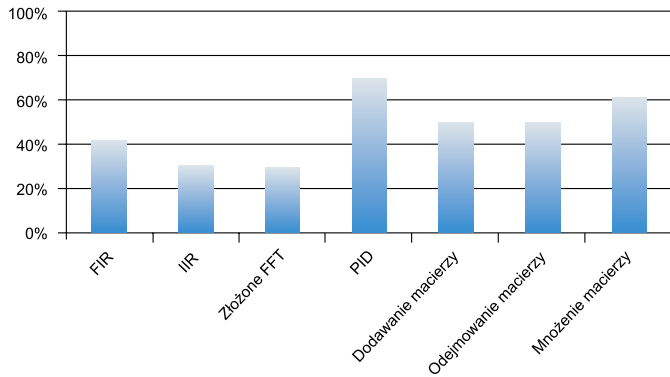
Jednostka zmiennoprzecinkowa

W zależności od potrzeb, producent mikrokontrolera może zdecydować o implementacji opcjonalnej jednostki zmiennoprzecinkowej (FPU). Pozwala ona na wykonywanie operacji: dodawania, odejmowania, mnożenia, dzielenia, mnożenia i sumowania oraz obliczania pierwiastka kwadratowego na zmiennych o pojedynczej precyzji. Umożliwia też dokonywanie konwersji zmiennych stałoprzecinkowych na zmiennoprzecinkowe (32- lub 16-bitowe) i odwrotnie. Obsługuje również polecenia z wykorzystaniem stałych zmiennoprzecinkowych. Rejestry FPU mogą być dwójako obsługiwane: jako 32 rejestry (S0...S31) o pojedynczej precyzji lub jako 16 rejestrów (D0...D15) o podwójnej precyzji.

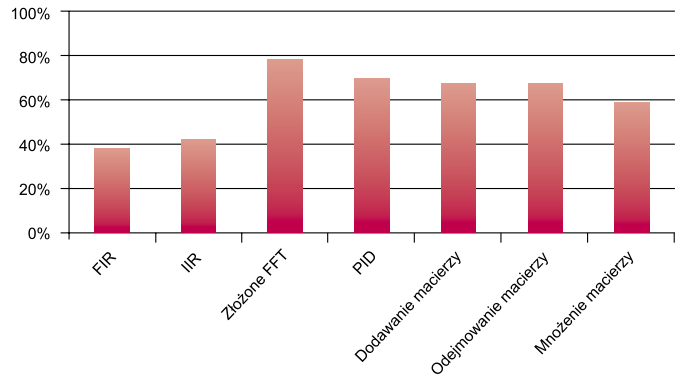
W praktyce, zastosowane w Cortex-M4 FPU odpowiada ona jednostce FPv4-SP z rdzenia ARMv7-M. W podstawowym trybie pracuje zgodnie ze standardem IEEE754. Może jednak zostać przełączony w dwa inne tryby: *Flush-to-Zero* lub *Default-NaN*. W pierwszym z nich, do którego przechodzi się po ustawieniu bitu FZ w rejestrze FPSCR (stanu i kontroli FPU), jednostka traktuje jako zera wszystkie nietypowe zmienne wejściowe, na których mają być wykonane operacje koprocesora (CDP). Wszelkie wyjątki powstające w wyniku tych działań są



Rys. 1. Schemat blokowy mikrokontrolera z rdzeniem Cortex-M4



Rys. 3. Czas wykonywania typowych zadań DSP na liczbach 16-bitowych z użyciem układu z rdzeniem Cortex-M4 w porównaniu do Cortex-M3



Rys. 4. Czas wykonywania typowych zadań DSP na liczbach 32-bitowych z użyciem układu z rdzeniem Cortex-M4 w porównaniu do Cortex-M3

odpowiednio sygnalizowane. Operacje: VABS (wartość bezwzględna), VNEG (zmiana znaku) i VMOV (kopiowanie zmiennej) nie są uznawane za operacje i w związku z tym nietypowe zmienne wejściowe nie są traktowane jako zera. Liczby o wartościach na tyle małych, że wg standardu IEEE754 wymagają zaokrąglenia, bo precyzja rejestru, w którym mają być zapisane jest zbyt mała, są w tym trybie zamieniane na zera. Zmianę zmiennej wejściowej na zero sygnali-

zuje flaga IDC. Zmianę wyniku operacji na zero sygnalizuje flaga UFC.

W trybie *Default-NaN*, do którego przechodzi się po ustawieniu bitu DN w rejestrze FPSCR, wszelkie w których zmiennymi wejściowymi są liczby niezgodne ze standardem, dają w wyniku wartość NaN (*Not-a-Number*). Jedynie w przypadku operacji VABS, VNEG i VMOV, wynik zawiera mantysę i wykładnik identyczny jak w zmiennej wejściowej.

Sposób na migrację

Różnice pomiędzy rdzeniami Cortex-M3 i Cortex-M4 zamieszczono w tabeli 1. Ponieważ rdzeń Cortex-M4 jest w praktyce rdzeniem Cortex-M3 wzbogaconym o dodatkowe moduły, przenoszenie kodu na nowsze układy jest łatwe. Obsługa stosu, pamięci, lokalizacja kodu i danych w pamięci oraz lista przerwania i instrukcji są identyczne dla obu rdzeni. Jeśli kod został napisany w języku C dla układu z rdzeniem Cortex-M3, będzie on równie dobrze działał z rdzeniem Cortex-M4. Kompilator samoczynnie tak zbuduje kod wynikowy, by wykorzystywał blok MAC i instrukcje SIMD w zadaniach DSP.

Podsumowanie

Układy z rdzeniem Cortex-M4 są zdecydowanie szybsze w licznych, bardzo popularnych zadaniach, w których jest konieczne przetwarzanie sygnałów cyfrowych. Ponadto są kompatybilne pod względem listy instrukcji i sposobu programowania, z układami z rdzeniem Cortex-M3, dzięki czemu przenoszenie kodu ze starszych układów na nowsze jest praktycznie bezproblemowe. Co więcej, w przypadku kodu napisanego w C, jego ponowna kompilacja pod kątem rdzenia Cortex-M4 automatycznie spowoduje użycie instrukcji wspomagających DSP. Dzięki czemu dotychczasowe projekty będą mogły być błyskawicznie przebudowane, tak by zmniejszyć taktowanie zegara mikrokontrolera i tym samym ograniczyć jego zużycie energii i nagrzewanie się. Pozwoli to wydłużyć czas pracy urządzeń przenośnych lub zmniejszyć ich masę, redukując rozmiary baterii i radiatorów.

Jednymi z pierwszych producentów, którzy rozpoczęli prace nad wprowadzeniem do sprzedaży układów z rdzeniami Cortex-M4 są Freescale, NXP i STMicroelectronics. W przypadku firmy Freescale są to układy z serii K10, K20, K30, K40 K50, K60 i K70. Aby ułatwić rozpoczęcie pracy z tymi układami, producent oferuje zestawy ewaluacyjne: TWR-K40X256-KIT i TWR-K60N512-KIT.

Marcin Karbowiczek, EP

Tabela 1. Porównanie cech rdzeni Cortex-M3 i Cortex-M4		
	Cortex-M3	Cortex-M4
Architektura	ARMv7-M (Harvard)	
Lista instrukcji	Thumb/Thumb-2	
Rozszerzenia DSP	Niedostępne	16- i 32-bitowe instrukcje MAC, podwójne 16-bitowe instrukcje MAC, 8- i 16-bitowe instrukcje SIMD – wszystkie wykonywane w jednym cyklu procesora. Ponadto sprzętowe dzielenie (od 2 do 12 cykli)
Opcjonalna jednostka FPU	Niedostępne	Jednostka FPU pojedynczej precyzji, zgodna z IEEE754
Potok przetwarzania	3 potoki oraz przetwarzanie spekulacyjne	
Dhrystone	1,25 DMIPS/MHz	
Ochrona pamięci	Opcjonalne MPU obsługujące 8 regionów, z subregionami i regionem tła	
Przerwania	Przerwanie niemaskowalne (NMI) oraz od 1 do 240 przerwania sprzętowych	
Opóźnienie obsługi przerwania	12 cykli	
Opóźnienie pomiędzy przerwaniem	6 cykli	
Liczba priorytetów przerwania	Od 8 do 256	
Przerwania budzące	Do 240	
Debugowanie	Opcjonalnie JTAG i porty debugowania szeregowego. Do 8 punktów wstrzymania i do 4 obserwowanych zmiennych.	
Śledzenie	Opcjonalnie śledzenie wykonywania instrukcji (ETM), śledzenie danych (DWT) oraz śledzenie zmiennych i sygnalizacja zdarzeń/przejść (ITM)	

Tabela 2. Porównanie wydajności rdzeni Cortex-M3 i Cortex-M4				
	Cortex-M3		Cortex-M4	
Proces technologiczny	TSMC 90 nm G		65 nm low power	
Rodzaj optymalizacji	Pod kątem szybkości	Pod kątem powierzchni	Pod kątem szybkości	Pod kątem powierzchni
Biblioteka elementów standardowych	ARM SC9	ARM SC9	ARM SC12	ARM SC9
Wydajność przetwarzania liczb całkowitych [DMIPS]	344	63	375	188
Częstotliwość [MHz]	275	50	300	150