

Wstęp

Zrealizowanie projektu z użyciem układu procesorowego rodziny TMS320C2000 firmy Texas Instruments nie musi być trudne. W tej książce znajdziesz wszystko co jest potrzebne do zrealizowania dobrego projektu z tym procesorem.

Książka jest przeznaczona dla użytkowników układów procesorowych wszystkich serii rodziny TMS320C2000: F2802x/3x/6x Piccolo, F2833x/34x Delfino oraz F28M35x Concerto. Również może być przydatna dla użytkowników starszych serii tej rodziny: F280xx oraz F281x. Wszystkie procesory rodziny TMS320C2000 zawierają ten sam rdzeń C28x oraz te same układy peryferyjne (z niewielkimi modyfikacjami). Również takie samo jest środowisko programowe oraz sposoby programowania układów procesorowych. Zagadnienia dotyczące tworzenia aplikacji dla układów procesorowych serii F2802x/3x/6x Piccolo omówiono szczególnie dokładnie.

Narzędzia do tworzenia programów dla układów procesorowych rodziny TMS320C2000 są dostarczane razem ze środowiskiem programowym *Code Composer Studio – CCSv4*. Wspomaganie programowania układów procesorowych rodziny TMS320C2000 jest zorganizowane jako pojedynczy pakiet programowy *controlSUITE*. Zawiera on również szczegółową dokumentację techniczną modułów sprzętowych. W pakiecie *controlSUITE* dostępne jest wsparcie dla układów procesorowych serii F2802x/3x/6x Piccolo, F2833x/34x Delfino oraz F28M35x Concerto. Zalecane (a właściwie konieczne) jest użycie pakietu *controlSUITE* podczas tworzenia własnego projektu dla układów procesorowych rodziny TMS320C2000. Do pracy z projektem potrzebna jest znajomość środowiska programowego *CCSv4* oraz technik programowania systemów wbudowanych czasu rzeczywistego w języku C. Konieczna jest również znajomość pakietów wspomagania programowania *Firmware Development Package* (w ramach *controlSUITE*) oraz środowiska czasu wykonania a także znajomość organizacji i działania sprzętowych zestawów uruchomieniowych. Bardzo przydatne jest również zrozumienie działania sprzętowych emulatorów JTAG oraz debuggera środowiska *CCSv4*. Najlepszym sposobem rozpoczęcia pracy jest użycie gotowego (działającego!) przykładowego projektu i zmodyfikowanie go do swoich potrzeb. Takie aplikacje są doskonałą platformą do prób i startu z własnym programem.

W książce najpierw dokładnie omówiono środowisko programowe *Code Composer Studio v4.2*. Dalej jest opis sposobów wspomagania programowania układów procesorowych z użyciem pakietu *controlSUITE*, metod programowania wewnętrznej pamięci Flash oraz sposobów wykonywania kodu z pamięci Flash a także odpowiedniego wyłączenia modułu CPU Watchdog. Następnie jest opis emulatorów i modułów sprzętowych.

Ćwiczenia praktyczne są zamieszczone w dalszej części książki. Jest ich dużo i umożliwiają zapoznanie się ze środowiskiem programowym *CCSv4*, sposobami inicjalizacji procesora, sposobami bootowania procesora oraz używania jego modułów pe-

ryferyjnych. Ćwiczenia można również wykonywać z użyciem środowiska CCSv5. Występują tylko nieduże zmiany zorganizowania perspektyw, okien, poleceń i sposobu ich używania.

Ćwiczenia pomyślano tak aby mogły być wykonywane z użyciem różnych modułów sprzętowych. Podstawowy opis dotyczy wykonywania ćwiczeń z zestawem sprzętowym:

- *PiccoloF28035 Experimenter Kit* firmy Texas Instruments – płytką bazową USB z modułem *F28035 Piccolo contrlCARD*.

Do wykonywania ćwiczeń można również zastosować zestawy z układem procesorowym innych serii *Piccolo*:

- Zestaw *Piccolo F28027 Experimenter Kit*. Płytką bazową USB z modułem *F28027 Piccolo contrlCARD*.
- Zestaw *Piccolo F28069 Experimenter Kit*. Płytką bazową USB z modułem *F28069 Piccolo contrlCARD*.

Do wykonywania ćwiczeń można również zastosować wtyczki USB:

- Wtyczka USB (*dongle*) *Piccolo F28027 controlSTICK*,
- Wtyczka USB (*dongle*) *Piccolo F28069 controlSTICK*.

Ćwiczenia można także wykonywać z użyciem innych układów procesorowych serii *F2802x/3x/6x Piccolo*.

Te same ćwiczenia, z niewielkimi zmianami, można wykonywać z użyciem układów procesorowych serii *F2833x/34x Delfino*. W pakietach programowych *Firmware Development Package* (w ramach *controlSUITE*) dla tych serii są udostępnione te same przykładowe projekty, które są używane w ćwiczeniach opisanych w książce. Podobnie jest w przypadku dwurdzeniowych układów procesorowych serii *F28M35x*. Przykłady dla rdzenia *C28x* są takie same.

Rozdział pierwszy zawiera przegląd układów procesorowych serii *F2802x/3x/6x Piccolo* firmy Texas Instruments.

Rozdział drugi zawiera opis środowiska Code Composer Studio. Dokładnie omówiono ścieżkę postępowania zaczynając od instalacji CCSv4 oraz definiowania konfiguracji sprzętowej. Następnie pokazano uruchamianie środowiska CCSv4, uruchamianie projektu, debugowanie programu (także w czasie rzeczywistym) oraz wizualizację danych. Są także omówione narzędzia do tworzenia programów, czyli kompilator, assembler, linker i program konwersji. Omówiono także format pliku COFF i środowisko czasu wykonania.

Rozdział trzeci zawiera opis sposobów wspomagania programowania układów procesorowych serii *TMS320F2802x/3x/6x Piccolo*. Na początku jest omówiona implementacja języka C. Następnie są dokładnie omówione pakiety programowe *Firmware Development Package* oraz środowisko czasu wykonania. Dalej przedstawiono metody programowania wewnętrznej pamięci Flash oraz sposoby wykonywania kodu z pamięci Flash a także odpowiedniego wyłączania modułu CPU Watchdog. Na koniec omówione jest oprogramowanie do cyfrowego przetwarzania sygnałów (i sterowania) udostępniane przez firmę Texas Instruments.

Rozdział czwarty zawiera opis emulatorów i modułów sprzętowych. Najpierw zamieszczony jest opis sprzętowych emulatorów JTAG oraz zastosowanego w nim konwertera USB-RS232 oraz oprogramowanie do jego programowania i obsługi. Dalej przedstawiono sprzętowe zestawy uruchomieniowe.

Rozdział piąty zawiera opis przykładowych ćwiczeń opracowanych tak, aby w kolejnych krokach poznać praktycznie działanie układu procesorowego serii F2803x Piccolo. Jest to wykonywane na poziomie języka C, z mały wycieczkami w kierunku assemblera. Ćwiczenie 1 umożliwia praktyczne zapoznanie się ze środowiskiem CCSv4 oraz z tworzeniem projektu, edytowaniem kodu, jego uruchamianiem i debugowaniem. Ćwiczenie 2 umożliwia praktyczne i dokładne zapoznanie się z inicjalizowaniem układu procesorowego serii F2803x Piccolo. Następne ćwiczenia 3–11 umożliwiają zapoznanie się z pracą modułów peryferyjnych układu procesorowego. Ćwiczenie 12 pozwala na praktyczne wykonywanie programowania pamięci Flash. Ćwiczenie 13 umożliwia poznanie trybów bootowania układów. Zadania zamieszczone w rozdziale 5 zawierają zadania umożliwiające zweryfikowanie nabytej wiedzy lub jej rozszerzenie. Dla ułatwienia sprawdzenia poprawności ich wykonania w książce zamieszczono dodatek C: Rozwiązania zadań z rozdziału 5. Opisy niektórych rozwiązań są bardzo obszerne, co umożliwia doskonałe rozszerzenie praktycznej wiedzy.

Dodatek A zawiera krótki opis środowiska programowego CCSv5.

Dodatek B zawiera listing kodu źródłowego dla ćwiczenia 1.

Dodatek C zawiera rozwiązania zadań zamieszczonych w ćwiczeniach z rozdziału 5.

Książka zawiera ponad 170 rysunków oraz ponad 50 tabel, które tak dobrano, aby maksymalnie ułatwić zrozumienie omawianych zagadnień.

Książka nie jest poradnikiem na temat programowania w języku C oraz w języku assemblerowym. Czytelnik powinien potrafić pisać programy w języku C. Powinien mieć również podstawową wiedzę o programowaniu w języku assemblerowym.

Książka nie również jest poradnikiem na temat projektowania systemów mikroprocesorowych. Czytelnik powinien posiadać podstawową wiedzę w tym zakresie.

Książka nie jest też poradnikiem na temat cyfrowego przetwarzania sygnałów. Jest wiele dobrych pozycji na ten temat dostępnych w języku polskim. Przykłady i ćwiczenia są tak dobrane, aby nie wymagało to szerokiej wiedzy z tej dziedziny.

W książce są zastosowane nazwy i oznaczenia zgodne z używanymi w dokumentacji producenta procesorów, firmy Texas Instruments. Ułatwi to korzystanie z oryginalnej dokumentacji firmowej. Występują jednak czasami niezgodności nazewnictwa i oznaczeń pomiędzy dokumentami. Czasami też oryginalne nazwy mogą być niedokładne i wręcz mylące. Takie sytuacje wymagały przyjęcia jednoznacznego rozwiązania.

Każdy układ procesorowy rodziny TMS320C200 ma bardzo dużo rejestrów modułów peryferyjnych i ich poszczególnych pól bitowych. Dla ułatwienia dokładnego określenia umiejscowienia bitu w rejestrze w książce podawana jest nazwa bitu razem z nazwą rejestru i numerem zajmowanej w nim pozycji bitowej, np. RXRDY(SCIRXST[6]). Podobną notację zastosowano dla pól bitowych rejestrów, np. pole bitowe PM[3:0](ST0[9:7]). Jest to notacja bardzo podobna do stosowa-

nej ostatnio w oryginalnej dokumentacji Texas Instruments. Ta zmiana wydaje się jednak konieczna, aby uniknąć wielu błędów i niedokładności, które występują w tej dokumentacji. Tak rozszerzona notacja powinna zdecydowanie pomóc podczas programowania, zwłaszcza z zastosowaniem modułów peryferyjnych. Szczególnie, że jednocześnie w opisie zamieszczonym w książce usunięto liczne błędy i pomyłki źródłowej dokumentacji.

Dla uważnych (a może cierpliwych czytelników) opowiem w dalszej części książki o pouczającej porażce budowy systemu czasu rzeczywistego, jaka mi się przydarzyła. Spróbujemy razem wyciągnąć z niej wnioski i znaleźć prawidłowe rozwiązanie. Zagadnienia dotyczące budowy układów procesorowych rodziny TMS320C200 są omówione w książce Henryka A. Kowalskiego *Procesory DSP dla praktyków*. Najpierw omówiony jest tam wspólny rdzeń C28x procesorów rodziny TMS320C2000 oraz budowa układów procesorowych serii F2802x Piccolo i F2803x Piccolo. Następnie są dokładnie omówione moduły peryferyjne obu serii tych układów.

Została uruchomiona witryna internetowa **PiccoloDSP** (<http://www.picolodsp.pl/>). Jest ona pomyślana jako podstawowe źródło aktualnych informacji dotyczących układów procesorowych serii TMS320F2802x/3x/6x Piccolo w języku polskim. Witryna zawiera aktualne informacje i materiały uzupełniające dla Czytelników książki. Jest też pomyślana jako aktualizowane rozszerzenie informacji zamieszczonych w książce o nowe tematy i zagadnienia. Planowane jest omawianie tam na bieżąco zmian w przeprowadzaniu ćwiczeń, które będą wymuszone poprzez przyszłe zmiany wersji środowiska narzędziowego.

Mam nadzieję, że zapoznanie się z tą książką pozwoli czytelnikowi z powodzeniem realizować własne projekty sprzętowe i programowe.

Henryk A. Kowalski