

---

<b>Wstęp</b> .....	<b>11</b>
<b>1. Procesory serii TMS320F2802x/3x/6x Piccolo</b> .....	<b>15</b>
1.1. Organizacja układów procesorowych serii F2802x Piccolo.....	23
1.2. Organizacja układów procesorowych serii F2803x Piccolo.....	29
1.3. Organizacja układów procesorowych serii F2806x Piccolo.....	34
<b>2. Code Composer Studio v4 – zintegrowane środowisko projektowe</b> .....	<b>41</b>
2.1. Instalowanie i uruchamianie środowiska CCSv4 .....	44
2.1.1. Pobieranie pliku instalacyjnego środowiska CCSv4.....	45
2.1.2. Instalowanie środowiska CCSv4.....	47
2.1.3. Uruchamianie środowiska CCSv4 .....	49
2.1.4. Aktualizowanie środowiska CCSv4 .....	54
2.1.5. Tworzenie/importowanie projektu .....	56
2.1.6. Perspektywa <i>C/C++</i> .....	62
2.1.7. Okno <i>C/C++ Projects</i> .....	64
2.1.8. Definiowanie konfiguracji sprzętowego systemu docelowego .....	66
2.2. Praca z projektem w środowisku CCSv4.....	73
2.2.1. Dodawanie plików z kodem źródłowym do projektu.....	74
2.2.2. Edytowanie kodu .....	75
2.2.3. Perspektywa <i>Debug</i> i uruchamianie debuggera.....	75
2.2.4. Dołączanie/odłączanie sprzętu do CCSv4.....	78
2.2.5. Okno <i>Debug</i> .....	79
2.2.6. Okno <i>Disassembly</i> .....	80
2.2.7. Budowanie projektu .....	82
2.2.8. Ładowanie kodu programu.....	83
2.2.9. Uruchamianie programu, praca krokowa i polecenie <i>Reset</i> .....	86
2.2.10. Pułapki.....	89
2.2.11. Debugowanie w czasie rzeczywistym.....	96
2.2.12. Okna podglądu, pamięci, rejestrów, stosu i symboli .....	97
2.2.13. Wizualizacja danych .....	103
2.3. Narzędzia generacji kodu dla układów procesorowych F2802x/3x/6x Piccolo .....	104
2.3.1. Pakiet <i>Code Generation Tools</i> (CGT) .....	104
2.3.2. Konfigurowanie narzędzi generacji kodu w środowisku CCSv4.....	107

<b>3. Programowanie układów procesorowych rodziny TMS320C2000 .....</b>	<b>111</b>
3.1. Implementacja języka C/C++ dla układów procesorowych rodziny TMS320C2000.....	112
3.1.1. Typy danych języka C/C++.....	112
3.1.2. Słowa kluczowe i wstawki assemblerowe.....	113
3.2. Metody programowania układów procesorowych rodziny TMS320C2000 w języku C/C++ .....	115
3.2.1. Mechanizm pól bitowych.....	116
3.2.2. Rozmiar kodu i wydajność .....	118
3.2.3. Optymalizacja rozmiaru kodu i czasu dostępu .....	120
3.2.4. Wpływ operacji RMW podczas stosowania dostępu pól bitowych rejestrów .....	122
3.2.5. Rejestry, których bity mogą być zmienione podczas operacji RMW.....	122
3.2.6. Rejestry z bitami typu zapisz jedynekę dla wyzerowania .....	123
3.2.7. Rejestry wymagające specjalnej wartości .....	125
3.2.8. Rejestry 32-bitowe (eCAN).....	126
3.3. Pakiety programowe <i>Firmware Development Package</i> dla układów procesorowych F2802x/3x/6x Piccolo.....	127
3.3.1. Organizacja plików pakietów programowych <i>Firmware</i> .....	128
3.3.2. Folder <i>headers</i> pakietów <i>Firmware Development Package</i> .....	130
3.3.3. Folder <i>common</i> pakietów <i>Firmware Development Package</i> .....	140
3.3.4. Projekty przykładowe pakietów programowych <i>Firmware</i> .....	154
3.4. Środowisko czasu wykonania .....	159
3.4.1. Model pamięci .....	159
3.4.2. Inicjacja środowiska języka C/C++ .....	161
3.4.3. Struktura funkcji, konwencje wywołań i obsługa przerwania .....	166
3.5. Wykonywanie kodu z pamięci Flash po operacji <i>Reset</i> .....	170
3.6. Wyłączenie modułu CPU Watchdog podczas uruchamiania programu w języku C .....	171
3.7. Programowanie wewnętrznej pamięci Flash .....	176
3.7.1. Biblioteki Flash API dla układów procesorowych F2802x/3x/6x Piccolo .....	179
3.7.2. Narzędzie programowe <i>On-Chip Flash</i> środowiska CCSv4 .....	181
3.8. Program C2Prog.....	183
3.8.1. Instalowanie programu C2Prog.....	185

---

3.8.2.	Przygotowanie pliku *.hex .....	185
3.8.3.	Programowanie pamięci Flash poprzez łącze UART/SCI.....	185
3.9.	Pakiet programowy <i>controlSUITE</i> .....	189
3.9.1.	Pakiety programowe <i>Firmware Development Package</i> .....	189
3.9.2.	Materiały dla zestawów sprzętowych.....	190
3.9.3.	Biblioteki.....	193
3.10.	Inne materiały z przykładami programowania .....	194
3.11.	Oprogramowanie do cyfrowego przetwarzania sygnałów .....	195
<b>4.</b>	<b>Emulatory i moduły sprzętowe.....</b>	<b>197</b>
4.1.	Sprzętowe emulatory dla łącza JTAG .....	198
4.2.	Emulator sprzętowy klasy XDS100.....	200
4.2.1.	Aplikacja Mprog – reprogramowanie konwertera FT2232C/D/H.....	202
4.2.2.	Program <i>xds100serial.exe</i> odczytu numeru seryjnego.....	204
4.2.3.	Wirtualny port COM na komputerze PC .....	205
4.3.	Sprzętowe moduły uruchomieniowe .....	206
4.3.1.	Wtyczka USB F28027 Piccolo <i>controlSTICK</i> .....	207
4.3.2.	Wtyczka USB F28069 Piccolo <i>controlSTICK</i> .....	208
4.3.3.	Moduł F28027 Piccolo <i>controlCARD</i> .....	209
4.3.4.	Moduł F28035 Piccolo <i>controlCARD</i> .....	211
4.3.5.	Moduł F28069 Piccolo <i>controlCARD</i> .....	213
4.3.6.	Płytki bazowa <i>USB docking station</i> .....	214
4.3.7.	Zestawy uruchomieniowe F28027, F28035 oraz F28069 „Experimenter Kit” .....	215
4.3.8.	Zestawy aplikacyjne z układami procesorowymi F2802x/3x Piccolo .....	216
4.4.	Konfigurowanie zestawów uruchomieniowych do pracy z emulatorem sprzętowym .....	218
<b>5.</b>	<b>Przykładowe ćwiczenia .....</b>	<b>221</b>
5.0.	Konfiguracja programowa i sprzętowa do ćwiczeń.....	224
5.0.1.	Konfiguracja programowa do ćwiczeń .....	224
5.0.2.	Konfiguracja sprzętowa do ćwiczeń .....	226
5.0.3.	Dobre (po)rady.....	228
5.1.	Ćwiczenie 1 – Pierwszy program w języku C w środowisku CCSv4.....	229
5.1.1.	Konfiguracja sprzętowa i programowa .....	230

5.1.2.	Tworzenie nowej lokalizacji folderu roboczego projektu ( <i>workspace</i> ) .....	231
5.1.3.	Tworzenie nowego projektu w środowisku CCSv4.....	232
5.1.4.	Tworzenie i edytowanie plików kodu źródłowego .....	235
5.1.5.	Budowanie projektu.....	237
5.1.6.	Definiowanie lokalnej docelowej konfiguracji sprzętowej .....	238
5.1.7.	Uruchamianie debuggera i ładowanie projektu .....	241
5.1.8.	Uruchamianie i debugowanie projektu .....	242
5.1.9.	Zastosowanie innej platformy sprzętowej .....	244
5.2.	Ćwiczenie 2 – Inicjalizacja systemu, układu przerwań i CPU Watchdog .....	244
5.2.1.	Konfiguracja sprzętowa i programowa .....	246
5.2.2.	Tworzenie folderu roboczego projektu ( <i>workspace</i> ) środowiska CCSv4 .....	247
5.2.3.	Otwieranie istniejącego projektu w środowisku CCSv4 .....	247
5.2.4.	Otwieranie pliku dokumentacji pakietu programowego <i>Firmware Development Package</i> .....	249
5.2.5.	Budowanie projektu .....	249
5.2.6.	Wybór typu układu procesorowego i konfigurowanie pamięci .....	252
5.2.7.	Definiowanie wspólnej docelowej konfiguracji sprzętowej .....	253
5.2.8.	Dołączanie wspólnej docelowej konfiguracji sprzętowej do projektu.....	254
5.2.9.	Budowanie projektu, uruchamianie debuggera oraz ładowanie kodu.....	254
5.2.10.	Obsługa modułu CPU Watchdog po operacji <i>Reset</i> .....	256
5.2.11.	Inicjowanie ustawień systemu .....	256
5.2.12.	Konfigurowanie działania modułu CPU Watchdog.....	258
5.2.13.	Wybór wejściowego sygnału zegarowego.....	258
5.2.14.	Ustawianie parametrów modułu PLL i systemowego sygnału zegarowego SYSCLKOUT.....	258
5.2.15.	Konfigurowanie wyjścia XCLKOUT oraz sygnału zegarowego LOSPCLK.....	261
5.2.16.	Inicjalizowanie sygnałów zegarowych dla modułów peryferyjnych.....	261
5.2.17.	Inicjalizowanie układu przerwań CPU oraz modułu PIE .....	262
5.2.18.	Kod użytkownika projektu <i>Example_2803xWatchdog</i> .....	264
5.2.19.	Obsługa przerwań.....	266
5.2.20.	Praca z programem <i>Example_2803xWatchdog</i> .....	266
5.2.21.	Zastosowanie innej platformy sprzętowej .....	268
5.3.	Ćwiczenie 3 – Obsługa modułu GPIO oraz liczników CPU Timer.....	268
5.3.1.	Konfiguracja sprzętowa i programowa .....	269
5.3.2.	Uruchamianie projektu <i>Example_2803xLedBlink</i> .....	269
5.3.3.	Konfigurowanie wyprowadzeń cyfrowych I/O (GPIO).....	271

---

5.3.4.	Inicjalizowanie modułów CPU Timer0/1/2.....	273
5.3.5.	Kod użytkownika projektu <i>Example_2803xLedBlink</i> .....	275
5.3.6.	Obsługa przerwania TINT0.....	276
5.3.7.	Rozmieszczenie zmiennych 32-bitowych w pamięci.....	276
5.3.8.	Ustawianie i zmiana stanu wyprowadzeń I/O (GPIO).....	278
5.3.9.	Odczyt stanu wyprowadzeń GPIO .....	278
5.3.10.	Zastosowanie innej platformy sprzętowej oraz inne przykłady pracy modułu GIO oraz CPU TIMER.....	279
5.4.	Ćwiczenie 4 – Generowanie sygnałów PWM z użyciem modułu ePWM.....	281
5.4.1.	Konfiguracja sprzętowa i programowa .....	282
5.4.2.	Uruchamianie projektu <i>Example_2803xEPwmUpAQ</i> .....	282
5.4.3.	Konfigurowanie wyprowadzeń cyfrowych I/O (GPIO) dla modułów ePWM1/2/3.....	284
5.4.4.	Konfigurowanie modułu ePWM .....	285
5.4.5.	Procedury obsługi przerwania modułu ePWM .....	290
5.4.6.	Praca programu <i>Example_2803xEPwmUpAQ</i> .....	292
5.4.7.	Debugowanie programu w trybie <i>Real-time</i> .....	293
5.4.8.	Praca swobodna modułu ePWM po zatrzymaniu debugowym.....	295
5.4.9.	Obsługa krytycznych przerwania po zatrzymaniu debugowym.....	295
5.4.10.	Zastosowanie innej platformy sprzętowej oraz inne przykłady pracy modułu ePWM.....	296
5.5.	Ćwiczenie 5 – Generowanie sygnałów PWM z użyciem modułu HR PWM .....	298
5.5.1.	Konfiguracja sprzętowa i programowa .....	298
5.5.2.	Uruchamianie projektu <i>Example_2803xHRPWM</i> .....	299
5.5.3.	Konfigurowanie modułu HRPWM.....	301
5.5.4.	Praca projektu <i>Example_2803xHRPWM</i> .....	304
5.5.5.	Generowanie sygnału analogowego .....	305
5.5.6.	Zastosowanie innej platformy sprzętowej oraz inne przykłady pracy modułu HRPWM .....	307
5.6.	Ćwiczenie 6 – Używanie modułu przetwornika ADC.....	308
5.6.1.	Konfiguracja sprzętowa i programowa .....	309
5.6.2.	Uruchamianie projektu <i>Example_2803xAdcSoc</i> .....	310
5.6.3.	Inicjalizowanie modułu ADC .....	312
5.6.4.	Konfigurowanie modułu ADC.....	313
5.6.5.	Procedura obsługi przerwania modułu ADC.....	313
5.6.6.	Konfigurowanie modułu ePWM1 .....	314

5.6.7.	Badanie wyjścia XCLKOUT sygnału zegarowego .....	314
5.6.8.	Praca programu <i>Example_2803xAdcSoc</i> .....	315
5.6.9.	Wizualizacja graficzna zmiennych tablicowych .....	317
5.6.10.	Niezynchronizowany podgląd stanu zmiennych .....	319
5.6.11.	Zastosowanie innej platformy sprzętowej oraz inne przykłady pracy modułu ADC .....	320
5.7.	Ćwiczenie 7 – Pomiar sygnału z czujnika temperatury układu procesorowego .....	321
5.7.1.	Konfiguracja sprzętowa i programowa .....	322
5.7.2.	Uruchamianie projektu <i>Example_2803xAdc_TempSensorConv</i> .....	323
5.7.3.	Funkcje odczytu parametrów czujnika temperatury .....	325
5.7.4.	Obsługa odczytu temperatury .....	326
5.7.5.	Obsługa przerwania ILLTRAP nielegalna instrukcja .....	328
5.7.6.	Odczyt typu i wersji układu procesorowego .....	332
5.7.7.	Konwersja odczytu wartości z czujnika temperatury .....	332
5.7.8.	Zastosowanie innej platformy sprzętowej oraz inne przykłady pracy modułu ADC z czujnikiem temperatury .....	334
5.8.	Ćwiczenie 8 – Pomiar okresu i częstotliwości sygnału cyfrowego z użyciem modułu eCAP .....	335
5.8.1.	Konfiguracja sprzętowa i programowa .....	335
5.8.2.	Uruchamianie projektu <i>Example_2803xECap_Capture_Pwm</i> .....	336
5.8.3.	Konfigurowanie wyprowadzeń GPIO dla modułu ePWM3 oraz eCAP1 .....	338
5.8.4.	Inicjalizowanie modułu ePWM3 oraz eCAP1 .....	339
5.8.5.	Procedura obsługi przerwania ECAP1_INT modułu eCAP1 .....	342
5.8.6.	Praca programu <i>Example_2803xECap_Capture_Pwm</i> .....	344
5.8.7.	Zastosowanie innej platformy sprzętowej dla pracy modułu eCAP .....	345
5.9.	Ćwiczenie 9 – Generowanie sygnałów PWM z użyciem modułu eCAP w trybie APWM .....	346
5.9.1.	Konfiguracja sprzętowa i programowa .....	347
5.9.2.	Uruchamianie projektu <i>Example_2803xECap_Capture_Pwm</i> .....	347
5.9.3.	Konfigurowanie wyprowadzeń GPIO dla modułu eCAP1 .....	349
5.9.4.	Inicjalizowanie modułu eCAP1 .....	350
5.9.5.	Praca programu <i>Example_2803xECap_apwm</i> .....	351
5.9.6.	Zastosowanie innej platformy sprzętowej dla pracy modułu eCAP w trybie APWM .....	352
5.10.	Ćwiczenie 10 – Transmisja danych poprzez łącze SCI/UART .....	352
5.10.1.	Konfiguracja sprzętowa i programowa .....	353

---

5.10.2.	Uruchamianie projektu <i>Example_2803xECap_Capture_Pwm</i> .....	355
5.10.3.	Konfigurowanie wyprowadzeń GPIO dla modułu SCI .....	356
5.10.4.	Inicjalizowanie modułu SCI.....	357
5.10.5.	Praca programu <i>Example_2803xSci_Echoback</i> .....	358
5.10.6.	Zastosowanie innej platformy sprzętowej oraz inne przykłady pracy modułu SCI.....	360
5.10.7.	Stosowanie sterownika standardu RS-232 .....	361
5.11.	Ćwiczenie 11 – Używanie modułu I2C .....	362
5.11.1.	Konfiguracja sprzętowa i programowa .....	363
5.11.2.	Uruchamianie projektu <i>Example_2803xI2C_eeprom</i> .....	365
5.11.3.	Struktura programu <i>Example_2803xI2C_eeprom</i> .....	367
5.11.4.	Konfigurowanie wyprowadzeń GPIO dla modułu I2C.....	368
5.11.5.	Inicjalizowanie modułu I2C .....	369
5.11.6.	Procedura obsługi przerwania I2CINT1A modułu I2C .....	371
5.11.7.	Praca programu <i>Example_2803xI2C_eeprom</i> .....	373
5.11.8.	Zastosowanie innej platformy sprzętowej do pracy modułu I2C.....	382
5.12.	Ćwiczenie 12 – Uruchamianie programu z wewnętrznej pamięci Flash układu procesorowego .....	383
5.12.1.	Konfiguracja sprzętowa i programowa .....	384
5.12.2.	Uruchamianie projektu <i>Example_2803xFlash</i> .....	385
5.12.3.	Konfigurowanie pamięci do pracy z pamięcią Flash.....	387
5.12.4.	Przepisywanie kodu programu z pamięci Flash do pamięci RAM.....	389
5.12.5.	Inicjalizowanie ustawień pamięci Flash .....	391
5.12.6.	Praca programu <i>Example_28035_Flash</i> .....	391
5.12.7.	Ładowanie symboli.....	393
5.12.8.	Znikająca pułapka programowa.....	394
5.12.9.	Zastosowanie innej platformy sprzętowej przy pracy z pamięcią Flash.....	397
5.12.10.	Nota aplikacyjna <i>Running an Application from Internal Flash Memory on the TMS320F28xx DSP</i> .....	398
5.13.	Ćwiczenie 13. Tryby bootowania układów procesorowych serii F2802x/3x/6x Piccolo.....	399
5.13.1.	Konfiguracja sprzętowa i programowa .....	400
5.13.2.	Sprawdzanie działania układu procesorowego podczas bootowania .....	401
5.13.3.	Ustawianie trybów bootowania układu procesorowego serii F2802x/3x/6x Piccolo .....	402
5.13.4.	Uruchamianie kodu programu po zakończeniu bootowania układu procesorowego serii F2802x/3x/6x Piccolo .....	406

<b>Dodatki</b> .....	<b>413</b>
Dodatek A. Code Composer Studio v5 .....	414
Dodatek B. Plik <i>Lab1.txt</i> .....	417
Dodatek C. Rozwiązania zadań z rozdziału 5 .....	418
<b>Literatura</b> .....	<b>430</b>