

# MSP430 w przykładach (7)

## Timer w trybie „wejścia-wyjścia”



**W poprzednim odcinku kursu pokazaliśmy jak korzystając z układu Timer odmierzać czas. Prezentowane były przykłady zegarka elektronicznego oraz pozytywki. Teraz zademonstrujemy sposób pomiaru czasu trwania i częstotliwości sygnału wejściowego, oraz metody generowania impulsów wyjściowych o określonej częstotliwości i wypełnieniu.**

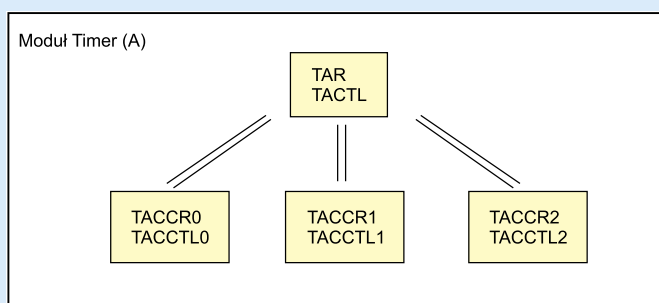
Zainstalowany w module Komputerek mikrokontroler MSP430f1232 ma 16-bitowy licznik TAR, który współpracuje z rejestrami TACCR0, TACCR1, TACCR2. Rejestry TACCRx mogą pracować w trybie porównywania – Compare, albo w trybie przechwytywania – Capture. Licznik TAR i rejestry w trybie porównującym mogą być użyte do odmierzenia czasu. Można ich także użyć do generowania sygnałów wyjściowych np. impulsów PWM.

Licznik TAR i rejestry w trybie przechwytywania mogą być użyte do pomiaru czasu trwania impulsu, pomiaru czasu pomiędzy impulsami, pomiaru częstotliwości sygnału. Budowa i działania układu Timer w MSP430f1232 prezentowane były w EP02/2013. Strukturę rejestrów modułu Timer ilustruje **rysunek 1**.

### Licznik i rejestry w trybie porównywania

W trybie porównywania, Compare, wartość licznika TAR jest porównywana z wartością wpisaną do rejestru TACCRx. W momencie, gdy obie wartości są równe, to w rejestrze konfiguracyjnym TACCTLx jest ustawiana flaga przerwania CCIFG. Dodatkowo, jest ustawiany wewnętrzny sygnał sterujący EQUx.

Rejestry TACCRx wyposażono w moduł wyjścia Output Unit. Wewnątrz modułu jest wytwarzany sygnał OUTx, który można wyprowadzić na linię wejścia-wyjścia mikrokontrolera, albo przekierować do modułu przetwornika analogowo-cyfrowego (wyzwalanie sekwencji pomiarowej). Kierunek sygnału OUTx konfiguruje bit CCISx. Moduł wyjścia może pracować w jednym z 8 trybów pracy konfigurowanych za pomocą bitu OUTMODx. Tryb pracy modułu określa sposób sterowania sygnałem OUTx. W trybie pracy 0 stan sygnału OUTx jest ustawiany za pomocą bitu konfiguracyjnego OUT. W pozostałych trybach pracy, sterowanie sygnałem OUTx jest uzależ-



**Rysunek 1.** Licznik TAR i rejestry TACCRx. Rejestr TACTL – konfiguracja pracy licznika TAR. Rejestry TACCTLx konfiguracja pracy rejestrów TACCRx

#### Dodatkowe materiały na CD/FTP:

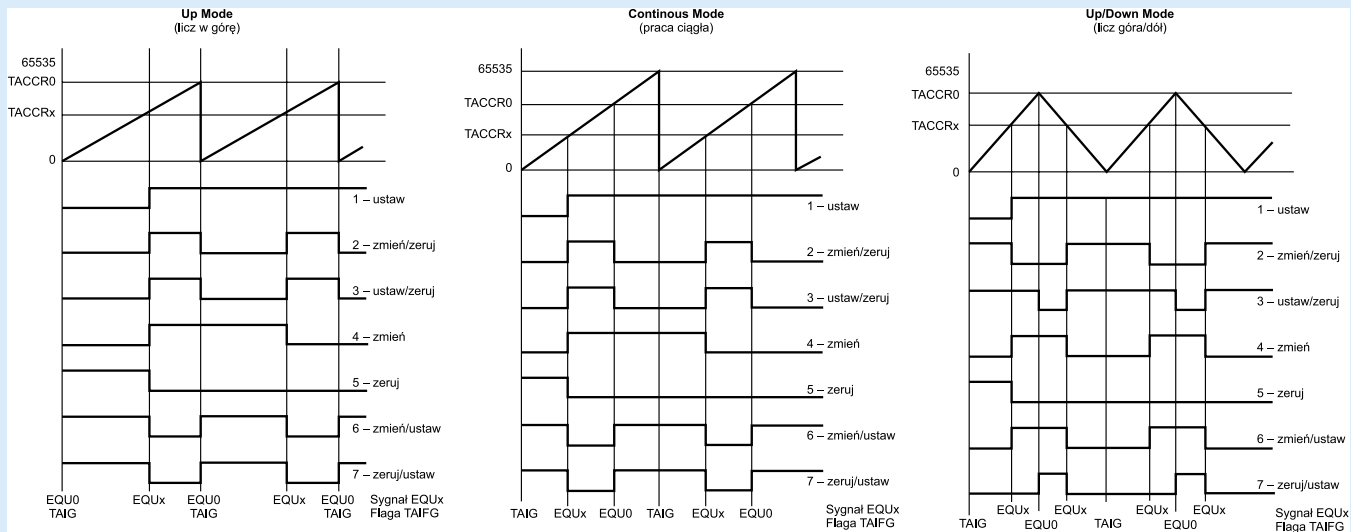
W materiałach dodatkowych do drugiego odcinka kursu (EP10/2012) publikowany był program „Pomiar częstotliwości sygnału DCOCLK”. Program demonstruje w jaki sposób korzystając z licznika TAR, oraz jednego z rejestrów TACCR0 w trybie przechwytywania wykonać pomiar częstotliwości sygnału. Na płycie CD oraz na serwerze FTP prezentujemy filmy ilustrujące działanie przykładów zamieszczonych w artykule.

<ftp://ep.com.pl>, user: 63048, pass: 632vmey5

nione od zmian wewnętrznych sygnałów EQUx (sygnały ustawiane wraz z flagą CCIFG, gdy wartość licznika TAR jest równa wartości wpisanej do rejestru TACCRx). Tryby pracy 1 oraz 5 stosowane są do wytwarzania pojedynczych zdarzeń czasowych. Sygnał OUTx może być ustawiony, albo wyzerowany. Tryb 4 może być użyty do generowania sygnału OUTx o częstotliwości o połowę mniejszej niż częstotliwości sygnału taktującego licznik TAR. Tryby pracy 2, 3, 6 oraz 7 wykorzystane są do generowania sygnałów OUTx o ustalonej częstotliwości i regulowanym wypełnieniu (impulsy PWM). Działanie trybów pracy modułu wyjścia prezentuje **tabela 1**. Przebiegi czasowe sygnałów OUTx w zależności od trybu pracy modułu wyjścia i trybu pracy licznika TAR ilustruje **rysunek 2**.

**Tabela 1.** Tryby pracy modułu wyjścia

Tryb	Nazwa	Sterowanie sygnałem OUTx
0	Wyjście. (Output)	Wartość wyjścia OUTx ustawia bit konfiguracyjny OUT
1	Ustaw. (Set)	Ustaw sygnał OUTx, gdy TAR = TACCRx (EQUx). Sygnał OUTx pozostaje ustawiony do momentu wyzerowania licznika TAR (bit TACLR).
2	Zmień/Zeruj (Toggle/Reset)	Zmień stan sygnału OUTx, gdy TAR = TACCRx (EQUx). Zeruj sygnał OUTx, gdy TAR = TACCR0 (EQU0).
3	Ustaw/Zeruj (Set/Reset)	Ustaw sygnał OUTx, gdy TAR = TACCRx (EQUx). Zeruj sygnał OUTx, gdy TAR = TACCR0 (EQU0).
4	Zmień. (Toggle)	Zmień stan sygnału OUTx, gdy TAR = TACCRx (EQUx).
5	Zeruj. (Reset)	Zeruj sygnał OUTx, gdy TAR = TACCRx (EQUx).
6	Zmień/Ustaw. (Toggle/Set)	Zmień stan sygnału OUTx, gdy TAR = TACCRx (EQUx). Ustaw sygnał OUTx, gdy TAR = TACCR0 (EQU0).
7	Zeruj/Ustaw. (Reset/Set)	Zeruj sygnał OUTx, gdy TAR = TACCRx (EQUx). Ustaw sygnał OUTx, gdy TAR = TACCR0 (EQU0).



**Rysunek 2. Charakterystyka sygnału OUTx. (moduł wyjścia w trybach pracy 1-7, licznik TAR w trybach pracy 1-3)**

W trybach pracy PWM (2,3,6,7) do sterowania sygnałem OUTx używany jest sygnał EQU0 (TACCRO), oraz jeden z sygnałów EQU1, EQU2 (TACCR1, TACCR2). Tryby pracy PWM (2,3,6,7) mogą być używane wyłącznie z rejestrami TACCR1, TACCR2. Pozostałe tryby pracy można stosować dla wszystkich rejestrów TACCRx. Wartość wpisana do rejestru TACCRO definiuje częstotliwość generowanego sygnału PWM. Wartość wpisana do rejestru TACCR1, TACCR2 określa wypełnienie sygnału PWM.

### Licznik i rejestry w trybie przechwytywania

W trybie przechwytywania, Capture, w momencie wystąpienia sygnału wyzwalającego, wartość licznika TAR jest zapisywana do rejestru TACCRx. Innymi słowy, wartość licznika TAR jest przechwytywana przez rejestr TACCRx (zatrzaśkiwana w rejestrze). Dodatkowo, w momencie wystąpienia zdarzenia w rejestrze konfiguracyjnym TACTLx jest ustawiana flaga przerwania CCIFG. Sygnałem wyzwalającym może być wewnętrzny sygnał zegarowy ACLK lub sygnał zewnętrzny doprowadzony do wejścia mikrokontrolera. Źródło sygnału wyzwalającego konfiguruje bit CCISx. Zbocze sygnału wyzwalającego, po wystąpieniu którego następuje przechwycenie wartości TAR i zapisanie w rejestrze TACCRx, konfiguruje bit CMx. Możliwe nastą-

Kiedy licznik TAR pracuje w trybie „licz do”, a rejestry pracują w trybie porównaj, oraz moduł wyjścia rejestru TACCR1/TACCR2 ma włączony siódmy tryb pracy, to częstotliwość sygnału PWM na wyjściu OUT1/OUT2 określa wzór 7.1. Wartość którą należy wpisać do rejestru TACCRO obliczymy ze wzoru 7.2.

$$f_{\text{Sygnału}} = f_{\text{Zegara}} / (w_{\text{Rejestr0}} + 1) \quad (7.1)$$

$$w_{\text{Rejestr0}} = (f_{\text{Zegara}} / f_{\text{Sygnału}}) - 1 \quad (7.2)$$

gdzie:

- $f_{\text{Sygnału}}$  - częstotliwość sygnału na wyjściu OUT1/2 (Hz)
- $f_{\text{Zegara}}$  - częstotliwość sygnału taktującego licznik TAR (Hz)
- $w_{\text{Rejestr0}}$  - wartość rejestru TACCRO

Procent wypełnienia sygnału określa wzór 7.3. Wartość którą należy wpisać do rejestru TACCR1 / TACCR2 obliczamy ze wzoru 7.4.

$$p_{\text{Wypelnienia}} = (w_{\text{Rejestr0}} / w_{\text{Rejestr12}}) * 100 \quad (7.3)$$

$$w_{\text{Rejestr12}} = (w_{\text{Rejestr0}} + 1) * p_{\text{Wypelnienia}} \quad (7.4)$$

gdzie:

- $p_{\text{Wypelnienia}}$  - procent wypełnienia sygnału na wyjściu OUT1/2 (%)
- $w_{\text{Rejestr0}}$  - wartość rejestru TACCRO
- $w_{\text{Rejestr12}}$  - wartość rejestru TACCR1/2

wy to: zbocze narastające, opadające oraz narastające i opadające.

Jeśli pomiędzy przechwyceniami nie odczytamy wartości z rejestru TACCRx (nadpisanie poprzedniej wartości nową), to ustawiony zostanie bit COV. Odczytując wartość bitu można wykryć przepełnienie trybu przechwytywania. Poziom sygnału wyzwalającego można odczytać sprawdzając wartość bitu CCI. Zazwyczaj sygnał wyzwalający jest asynchroniczny względem sygnału taktującego licznik TAR. Może zająć sytuacja, w której w momencie przepisywania wartości licznika TAR do rejestru TACCRx wartość licznika

Kiedy licznik TAR pracuje w trybie pracy ciągłej, a przechwytywanie wartości z TAR do TACCRx aktywują oba zbocza sygnału wyzwalającego (opadające i rosnące) to aby obliczyć czas trwania impulsu wyzwalającego należy obliczyć liczbę taktów licznika TAR pomiędzy zatrzaśnięciami. Jeśli druga wartość zatrzaśnięta w rejestrze TACCRx jest większa bądź równa pierwszej to korzystamy ze wzoru 7.5 (rysunek 3a). W przeciwnym wypadku korzystamy ze wzoru 7.6 (rysunek 3b). W obu przypadkach uwzględniamy możliwość przepełnienia licznika TAR.

$$It_{\text{TAR}} = d_{\text{WartoscTAR}} - p_{\text{WartoscTAR}} + 65536 * IP \quad (7.5)$$

$$It_{\text{TAR}} = [1 + d_{\text{WartoscTAR}} + 65535 - p_{\text{WartoscTAR}} + 65536 * (IP-1)] \quad (7.6)$$

gdzie:

- $It_{\text{TAR}}$  - liczba taktów licznika TAR pomiędzy zatrzaśnięciami
- $d_{\text{WartoscTAR}}$  - druga wartość TAR zatrzaśnięta w TACCRx
- $p_{\text{WartoscTAR}}$  - pierwsza wartość TAR zatrzaśnięta w TACCRx
- $IP$  - liczba przepełnień licznika TAR pomiędzy zatrzaśnięciami

Znając liczbę taktów licznika TAR pomiędzy zatrzaśnięciami, czas trwania impulsu wyzwalającego przechwytywanie obliczamy ze wzoru 7.7.

$$\text{czasImpulsu} = It_{\text{TAR}} / f_{\text{Zegara}} \quad (7.7)$$

gdzie:

- $\text{czasImpulsu}$  - czas trwania impulsu wyzwalającego przechwytywanie (s)
- $It_{\text{TAR}}$  - liczba taktów licznika TAR pomiędzy zatrzaśnięciami
- $f_{\text{Zegara}}$  - częstotliwość sygnału taktującego licznik TAR (Hz)

**Tabela 2. Konfiguracja wyjścia sygnału OUTx w MSP430f1232**

rejestr TACCRx	rejestr TACCTLx	bit CCISx	wyjście sygnału OUTx (linia we-wy, przetwornik A/C)
TACCR0	TACCTL0	CCIS_0	linia P1.1
		CCIS_1	linia P1.5
		CCIS_2	linia P2.2
		CCIS_3	wejście przetwornika ADC10
TACCR1	TACCTL1	CCIS_0	linia P1.2
		CCIS_1	linia P1.6
		CCIS_2	linia P2.3
		CCIS_3	wejście przetwornika ADC10
TACCR2	TACCTL2	CCIS_0	linia P1.3
		CCIS_1	linia P1.7
		CCIS_2	linia P2.4
		CCIS_3	wejście przetwornika ADC10

TAR zostanie zwiększona. Wówczas wartość zapamiętana w rejestrze TACCRx będzie nieprawidłowa. Żeby ustrzec się od błędów tego typu, należy włączyć synchronizację operacji przechwytywania z następnym zboczem sygnału taktującego licznik TAR. Opcję tę konfiguruje bit SCS.

W praktyce, korzystając z trybu przechwytywania, możemy wykonać pomiar czasu trwania impulsu, czasu pomiędzy impulsami, a po zastosowaniu przekształceń matematycznych – częstotliwości sygnału. Możemy zmierzyć parametry sygnału taktującego licznik TAR lub sygnału wyzwalającego. Warunkiem koniecznym do wykonania pomiarów jest znajomość parametrów jednego z sygnałów (taktującego licznik TAR lub wyzwalającego). Wówczas na podstawie wartości zatrzaśniętej w rejestrze TACCRx i parametrów jednego z sygnałów możemy obliczyć parametry drugiego.

### Przykłady

W przykładzie „Generator przebiegu prostokątnego” będziemy wytwarzać dwa przebiegi prostokątne o określonej częstotliwości i wypełnieniu. *Użyjemy licznika*

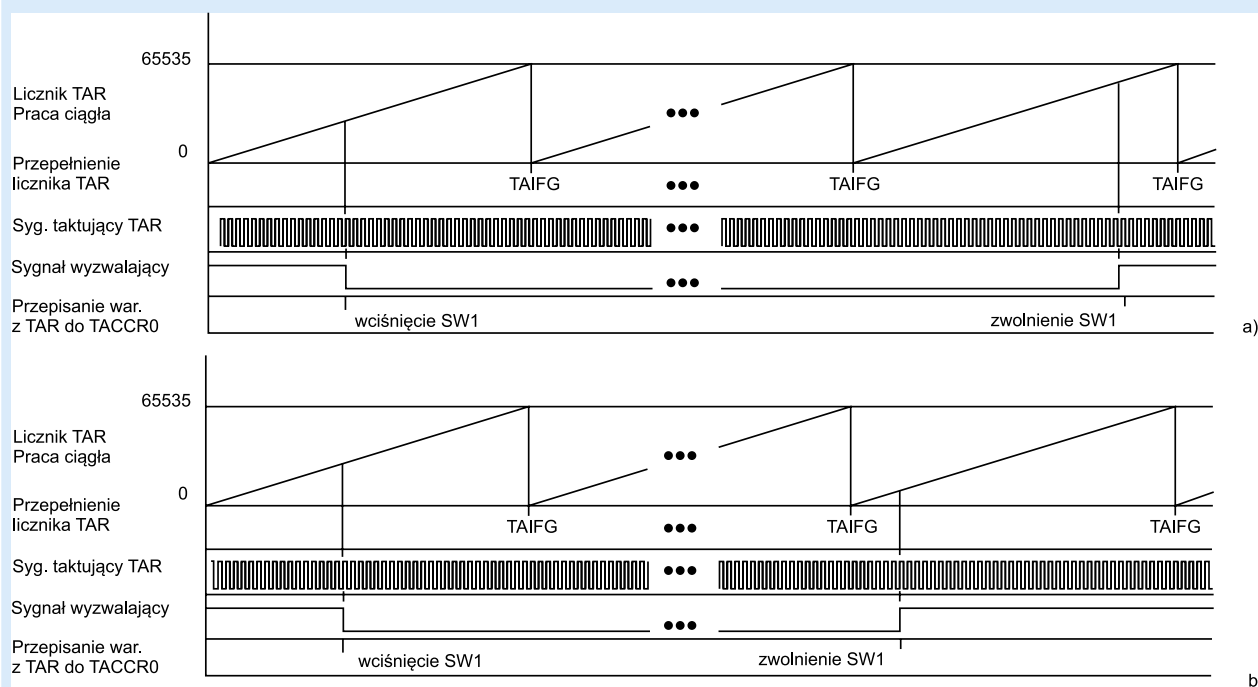
TAR pracującego w trybie „licz do” oraz rejestrów TACCR0, TACCR1, TACCR2 pracujących w trybie porównaj. Moduł wyjścia rejestrów TACCR1, TACCR2 ustawimy w tryb pracy PWM, a sygnały OUT1, OUT2 wyprowadzimy na linie P1.2, P1.3 mikrokontrolera. Generowane będą dwa sygnały o częstotliwości 100 Hz. Pierwszy o wypełnieniu 25% drugi 50%.

W przykładzie „Pomiar czasu trwania impulsu” zmierzmy czasu wciśnięcia przycisku SW1. Wykorzystamy licznik TAR pracujący w trybie pracy ciągłej, oraz rejestr TACCR0 w trybie przechwytywania. Licznik TAR będziemy taktować sygnałem ACLK. Sygnał wyzwalający przechwytywanie doprowadzimy do linii P1.1, do której w module „Komputerek” jest dołączony przycisk SW1. Wciśnięcie przycisku spowoduje zwarcie do masy i wyzerowanie linii. W sposób graficzny pracę licznika TAR, oraz przebieg sygnału wyzwalającego ilustruje rysunek 3.

### Generator przebiegu prostokątnego

Program „Generator przebiegu prostokątnego” uruchamiamy korzystając z modułu „Komputerek”. Zworki JP7, JP8 dołączające rezonator kwarcowy do źródła zegarowego LFXT1 należy ustawić w pozycji LF. Pozostałe zworki układu należy ustawić w pozycji IO/Off.

Pliki źródłowe programu zostały zamieszczone na płycie CD i serwerze FTP. W programie głównym zatrzymywana jest praca układu Watchdog, konfigurowane są linie I/O mikrokontrolera oraz jest 4-krotnie zmniejszana częstotliwość sygnału zegarowego ACLK taktującego licznik TAR ( $BCSCTL1 \mid = DIVA_2;$ ). Ponieważ sygnał ACLK jest podawany ze źródła LFXT1 (kwarc zegarkowy o częstotliwości 32768 Hz), to po przeskalowaniu częstotliwość sygnału ACLK wynosi 8192 Hz. Linie P1.2 i P1.3, na które wyprowadzone będą sygnały OUT1 OUT2 ustawiane są w tryb pracy funkcyjny ( $P1SEL \mid = (BIT3 + BIT2);$ ). Następnie jest konfigurowana praca rejestrów TACCRx. Rejestr TACCR0 jest ustawiany w tryb pracy z porównaniem (wyzerowanie bitów konfigura-



**Rysunek 3. Pomiar czasu trwania impulsu – przebiegi sygnałów. Druga przechwycona wartość TAR (zwolnienie przycisku) jest a) większa b) mniejsza od pierwszej przechwyconej wartości TAR (wciśnięcie przycisku)**

**Tabela 3. Konfiguracja wejścia sygnału wyzwalającego w MSP430f1232**

rejestr TACCRx	rejestr TACCTLx	bit CCISx	wejście sygnału wyzwalającego
TACCR0	TACCTL0	CCIS_0	linia P1.1
		CCIS_1	linia P2.2
		CCIS_2	---
		CCIS_3	---
TACCR1	TACCTL1	CCIS_0	linia P1.2
		CCIS_1	linia P2.3
		CCIS_2	---
		CCIS_3	---
TACCR2	TACCTL2	CCIS_0	linia P1.3
		CCIS_1	sygnał zegarowy ACLK
		CCIS_2	---
		CCIS_3	---

cyjnych). Aby ustalić częstotliwość sygnałów OUT1, OUT2 na 100 Hz do rejestru TACCR0 wpisywana jest wartość 81 (patrz wzór 7.2).

$$TACCTL0 = 0;$$

$$TACCR0 = 81;$$

Rejestry TACCR1 jest przełączany do trybu pracy z porównaniem. Układ wyjścia rejestru jest konfigurowany w 7 tryb pracy (tryb PWM – zeruj/ustaw). Sygnał OUT1 jest wyprowadzany na linii P1.2 (tabela 2). Aby ustalić poziom wypełnienia sygnału OUT1 na 25%, do rejestru TACCR1 jest wpisywana wartość 21 (patrz wzór 7.4).

$$TACCTL1 = CCIS_0 + OUTMOD_7;$$

$$TACCR1 = 21;$$

Rejestr TACCR2 jest konfigurowany w analogiczny sposób, jak rejestr TACCR1. Z tą różnicą, że poziom wypełnienia sygnału OUT2 jest ustalany na 50% (wzór 7.4), a sygnał jest wyprowadzany na wyjście P1.3 (tabela 2).

$$TACCTL1 = CCIS_0 + OUTMOD_7;$$

$$TACCR1 = 42;$$

Na zakończenie są konfigurowane parametry pracy licznika TAR. Licznik jest przełączany do pracy w trybie „licz do” (bit MC\_1). Włączane jest taktowanie licznika sygnałem zegarowym ACLK (bit TASSEL\_1). Wartość licznika jest zerowana (bit TACLRL). Mikrokontroler jest wprowadzany w tryb uśpienia LPM3 (w uśpieniu sygnał zegarowy ACLK taktujący licznik TAR jest aktywny). Na wyjściach P1.2 i P1.3 mikrokontrolera generowane są sygnały o częstotliwości 100 Hz i wypełnieniu 25% oraz 50%.

### Pomiar czasu trwania impulsu

Program „Pomiar czasu trwania impulsu” uruchamiamy korzystając z modułu „Komputerek”. Zworki JP7, JP8 dołączające rezonator kwarcowy do źródła zegarowego LFXT1 należy ustawić w pozycji LF. Pozostałe zworki układu należy ustawić w pozycji IO/Off, a w złączu szpilkowym Dis1 zamontować wyświetlacz LCD.

Pliki źródłowe programu zostały zamieszczone na płycie CD i serwerze FTP. W pierwszych liniach programu dołączane są pliki nagłówkowe, deklarowane są procedury oraz zmienne globalne. W programie głównym zatrzymywana jest praca układu Watchdog, konfigurowane są linie I/O mikrokontrolera, wyświetlacz LCD, parametry pracy licznika TAR, rejestru TACCR0

oraz jest włączana obsługa przerw maskowalnych. Linia P1.1, na którą jest podawany sygnał wyzwalający (przycisk SW1) jest konfigurowana jako wejście pracujące w trybie funkcyjnym.

$$P1DIR \&= \sim BIT1;$$

$$P1SEL |= BIT1;$$

Rejestr TACCR0 jest przełączany do pracy w trybie przechwytywania (bit CAP). Sygnał wyzwalający można podać na linii P1.1 albo P1.2 (tabela 3). W przykładzie ustawiana jest linia P1.1 (bit CCIS\_0). Przechwytywanie aktywują oba zbocza (narastające i opadające) sygnału wyzwalającego (bit CM\_3). Operacja przechwytywania jest zsynchronizowana z sygnałem taktującym licznik TAR (bit SCS).

$$TACCTL0 = CM_3 + CCIS_0 + CAP + SCS;$$

Licznik TAR jest taktowany sygnałem zegarowym ACLK (bit TASSEL\_1). Licznik pracuje w trybie pracy ciągłej (bit MC\_2). Przed rozpoczęciem pracy licznika zerowana jest jego wartość (bit TACLRL), oraz włączana jest obsługa przerw od przepełnienia licznika (bit TAIE).

$$TACTL = TASSEL_1 + MC_2 + TACLRL + TAIE;$$

W pętli głównej programu jest zerowana flaga przerwania dla rejestru TACCR0 oraz włączana obsługa przerw.

$$TACCTL0 \&= \sim CCIFG;$$

$$TACCTL0 |= CCIE;$$

Następnie mikrokontroler jest wprowadzany w tryb obniżonego poboru energii LPM3, w którym jest aktywny sygnał ACLK taktujący licznik TAR. Wciśnięcie przycisku SW1 powoduje zmianę poziomu sygnału na linii P1.1. Opadające zbocze sygnału wyzwalającego uruchamia procedurę przechwytywania. Wartość licznika TAR jest zapisywana do rejestru TACCR0 oraz jest ustawiana flaga przerwania CCIFG. Ponieważ włączyliśmy obsługę przerw dla rejestru TACCR0, to jest uruchamiana procedura obsługi przerwania. W procedurze wartość rejestru TACCR0 jest zapamiętywana w zmiennej pomocniczej, jest zerowany licznik przepełnień licznika TAR (przepełnienia licznika są cały czas liczone), a mikrokontroler wraca do trybu uśpienia. Zwolnienie przycisku SW1 powoduje zmianę stanu na linii P1.1. Narastające zbocze sygnału wyzwalającego uruchamia procedurę przechwytywania. Wartość licznika TAR jest zapisywana do rejestru TACCR0, oraz jest ustawiana flaga przerwania CCIFG. Ponownie jest uruchamiana procedura obsługi przerwania. W procedurze wartość rejestru TACCR0 jest zapamiętywana w zmiennej pomocniczej, a mikrokontroler budzony z uśpienia. Po wyjściu z uśpienia w pętli głównej programu jest wyłączana obsługa przerw dla rejestru TACCR0.

$$TACCTL0 \&= \sim CCIE;$$

Następnie, na podstawie zapamiętanych wartości rejestru TACCR0 i liczby przepełnień licznika TAR (patrz wzór 7.7), jest obliczany czas trwania impulsu (czas wciśnięcia przycisku SW1), a wynik pomiaru wyświetlany jest na ekranie LCD. W pętli głównej programu jest ponownie czyszczona flaga przerwania dla rejestru TACCR0, włączana obsługa przerw, a mikrokontroler jest usypiany. Moduł „Komputerek” jest gotowy do kolejnego pomiaru.

**Łukasz Krysiwicz, EP**