

# MSP430, część 5

## Co ma takiego, czego inne nie mają?

*Kontynuujemy omawianie możliwości podstawowych liczników-timerów, w które wyposażono mikrokontrolery z rodziny MSP430. Autor artykułu, korzystając ze swojej wiedzy i ogromnego doświadczenia odnosi się w tekście do innych mniej i bardziej znanych architektur mikrokontrolerowych, co ułatwi ocenę rozwiązań zastosowanych w MSP430.*

Można pójść jeszcze krok dalej i zamiast modulowania tylko współczynnika wypełnienia modulować też tenże współczynnik modulacji. Prowadzi to do modulatorów drugiego (a także wyższych rzędów), o jeszcze lepszych potencjalnie parametrach. Istnieją też nieco inne sposoby modulacji sigma-delta, chociażby przez modulację współczynnika podziału częstotliwości preskalera poprzedzającego licznik PWM, albo nawet okresu powtarzania impulsów.

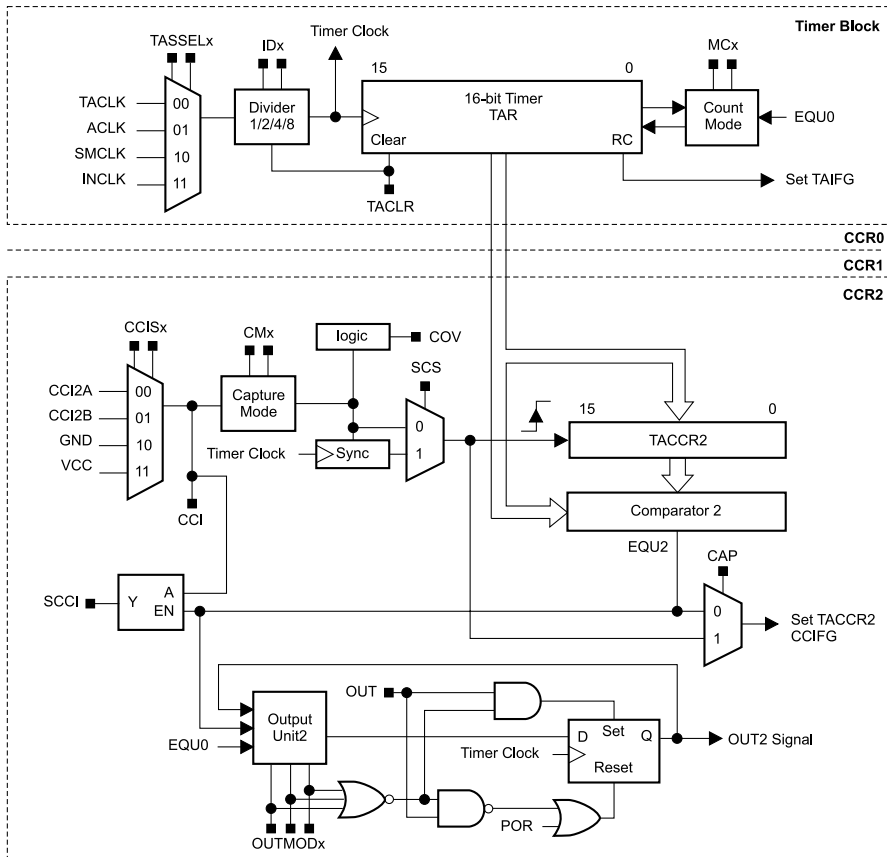
Taki sposób modulacji przebiegu PWM, mimo swych zalet, nie jest zbyt często spotykany, bo jednak jest trudniejszy realizacyjnie od rozwiązań najprostszych. W znanych autorowi mikrokontrolerach jest zaimplementowany w niezbyt popularnym w Polsce CoolRiscu szwajcarskiej firmy Xemics. Jego parametry są doskonałe. Nieco podobne rozwiązanie przyjęto też w mikrokontrolerach serii ADuC8xx firmy Analog Devices. Pozostaje jeszcze problem zachowania generatora PWM przy zmianach wartości współczynnika modulacji. Chodzi o to, by zmiany te były płynne i pomiędzy wartością początkową i docelową nie występowały jakieś niepożądane zakłócenia i szkodliwe impulsy. Niech w rozpatrywanym poprzednio przykładzie

genera-  
tora PWM  
o okresie od-  
powiadającym 1000  
taktów zegara do rejestru  
PWM wpisana jest liczba 387,  
a aktualny stan licznika wynosi  
350. W tym momencie do rejestru  
wpisywana jest nowa wartość, po-  
wiedzmy 200. Ponieważ licznik  
nie osiągnął jeszcze wartości 387,  
w chwili wpisu nowej wartości stan  
na wyjściu też nie ulega zmianie.  
Po modyfikacji zawartości rejestru  
pozostaje on nadal na dotychczas-  
sowym poziomie, bo teraz stan  
licznika jest większy od zawartości  
rejestru, a zmiana stanu wyjścia na-  
stępuje przecież z chwilą zrównania  
obu wartości. Nastąpi to dopiero  
w następnym cyklu. W efekcie po  
impulsach o współczynniku wypeł-  
nienia 0,387 pojawi się pojedynczy  
impuls bardzo długi, odpowiadający  
pełnemu cyklowi + nowa wartość,  
a więc aż 1200 taktów. Po tym  
następne impulsy będą już miały  
poprawny współczynnik wypełnie-  
nia 0,200. Jak zatem widać, nawet  
w realizacji prostych wydawałoby  
się układów PWM kryją się pewne  
pułapki. Stąd rzeczywiście imple-  
mentowane generatory PWM bywa-  
ją całkiem wyrafinowane, i często  
bazują na liczniku o przełączanym

kierunku  
zliczania  
górze/dół.

U k ł a -  
dy licznikowo-  
czasowe są uży-  
wane nie tylko  
do ogólnie pojętego  
odmierzania czasu,  
mogą także reagować na sygnały  
zewnętrzne. W 8051 jest dostępna  
funkcja *gate* (bramkowanie) pozwa-  
lająca licznikowi zliczać tylko wte-  
dy, gdy poziom sygnału zewnętr-  
znego, na skojarzonym z licznikiem  
wyprowadzeniu, jest wysoki. Gdy  
jest niski – stan licznika nie ule-  
ga zmianie. W ten prosty sposób  
można chociażby zamienić czas  
trwania impulsu wejściowego na  
odpowiadający mu stan końcowy  
licznika. W bardziej zaawansowa-  
nych rozwiązaniach zakres ingeren-  
cji sygnału zewnętrznego na pracę  
licznika jest większy. Najbardziej  
uniwersalna jest funkcja *capture*  
– przechwytywanie – w której ak-  
tualny stan licznika jest kopiowa-  
ny w dodatkowym rejestrze *capture*  
*register* dokładnie w momencie po-  
jawienia się zewnętrznego zdarze-  
nia (*event*), a licznik kontuuje  
normalną pracę. Zdarzeniem tym  
może być wystąpienie narastające-  
go zbocza sygnału, zbocza opadają-





Rys. 8. Schemat blokowy licznika A

cego lub dowolnego z tych zboczy. Równocześnie wywoływane jest odpowiednie przerwanie, w obsłudze którego, pomimo nieuniknionych przecież opóźnień, programista ma w rejestrze przechwytywania nadal dostępną informację o faktycznym momencie wystąpienia śledzonego zdarzenia. Możliwe jest nawet wyłączenie w ogóle systemu przerwań i cykliczna kontrola znacznika wystąpienia oczekiwanego zdarzenia. Czasami kopiowanie stanu licznika do rejestru następuje nie z każdym zboczem, a co kilka, np. co 4 lub co 16. Rozszerza to funkcjonalność tej opcji. Bywa też możliwe włączenie filtra przeciwwakłóceniewego, kiedy to muszą być spełnione dodatkowe warunki nałożone na sygnał wejściowy, aby nastąpiło „zamrożenie” stanu licznika. Zwykle po zmianie stanu sygnał musi być stabilny przez wybrany okres czasu, wyrażany w liczbie taktów zegara. Jeżeli pojawi się tylko krótki impuls, to jest on ignorowany. Ponieważ bardzo rzadko pojawia się potrzeba równoczesnego wykorzystywania w tym samym liczniku zarówno funkcji *capture* jak i *compare*, więc często, przede wszystkim dla zmniejszenia liczby reje-

strów obsługiwanych programowo, bywa dostępna tylko jedna z nich, na podstawie wyboru w rejestrze konfiguracyjnym. Ten sam rejestr pełni albo funkcję *capture register*, albo *compare register*. Oczywiście w licznikach prawie zawsze można wybrać tryb zliczania zboczy zewnętrzznego sygnału wejściowego.

Jak wynika z tego krótkiego przeglądu, różnorodność spotykanych w mikrokontrolerach układów licznikowo-czasowych jest bardzo duża, przy czym niektóre bardziej zaawansowane funkcje – *compare*, *capture*, PWM należą obecnie do standardu. Poza typowymi układami licznikowymi spotykane bywają liczniki dedykowane, przeznaczone do konkretnych zastosowań. Wymienić tu można układy nadzorujące (*watchdog timer*, nie ma chyba dobrej polskiej nazwy) oraz przede wszystkim liczniki taktujące układy transmisji szeregowej. Szczególnie te ostatnie wymagają czasem niekonwencjonalnych rozwiązań. Główny problem polega na tym, aby standardowe prędkości transmisji szeregowej dały się uzyskać przy dowolnej częstotliwości zegarowej mikrokontrolera. W starszych rozwiązaniach nie było to moż-

liwe i wymagało stosowania konkretnych częstotliwości generatora kwarcowego (przykładowo chociażby 11,059 MHz w 8051). Obecnie nawet przy taktowaniu mikrokontrolera stosunkowo małymi częstotliwościami zegarowymi daje się często uzyskiwać prędkości transmisji szeregowej nie odbiegające nadmiernie od wartości znamionowych. MSP430 stoi tu znowu na pozycji lidera, gdyż wbudowany w układ transmisji szeregowej licznik taktujący, z modulacją okresu, pozwala na uzyskanie poprawnej transmisji o nawet szybkości 9600 bodów przy częstotliwości pracy mikrokontrolera „zegarkowej”, zaledwie 32768 Hz! Nie istnieje chyba żadne inne rozwiązanie o aż tak dobrych parametrach.

Nawet w najprostszych wersjach mikrokontrolerów MSP430 zawsze występuje jeden układ licznikowo-czasowy; nosi on nazwę *Timer\_A* (rys. 8). W bardziej rozbudowanych egzemplarzach występuje jeszcze dodatkowy *Timer\_B*. Pod względem funkcjonalnym są one bardzo do siebie podobne. Liczniki te mają bardzo rozbudowane możliwości konfiguracyjne i mogą wykonywać większość dyskusowanych wyżej zadań.

*Timer\_A* jest asynchronicznym licznikiem 16-bitowym, „obudowanym” co najmniej dwoma, albo nawet trzema rejestrkami *capture/compare*. Może pracować w 4 różnych trybach, z możliwością wyboru jednego z 4 źródeł sygnału taktującego. Źródłami tymi są:

- *ACLK*, czyli buforowany przebieg wyjściowy generatora kwarcowego małej częstotliwości; zazwyczaj ma on częstotliwość „zegarkową” 32768 Hz, która może być dodatkowo podzielona przez 2, 4 lub 8.
- *SMCLK*, a więc zasadniczo przebieg wyjściowy generatora RC albo drugiego generatora kwarcowego, dużej częstotliwości (jeżeli istnieje, nie w każdej wersji MSP430 ten generator jest implementowany i jest wtedy zastępowany generatorem małej częstotliwości). Źródłowa częstotliwość sygnału dla *SMCLK* też może być opcjonalnie podzielona przez 2, 4 lub 8.
- jeden z 2 sygnałów wejściowych, z wyprowadzeń przyporządkowanych licznikowi A i oznaczonych *TACLK* oraz *TAINCLK*.

Poza uniwersalnymi układami licznikowo-czasowymi w MSP430 zaimplementowano także licznik nadzorujący WDT (*WatchDog Timer*). Jego zasadniczym zadaniem, zgodnie z nazwą, jest nadzorowanie poprawnej pracy oprogramowania, które powinno cyklicznie, w określonych ostępach czasu, zerować go. Jeżeli z jakichkolwiek powodów to nie nastąpi, to przepiętnienie licznika wywołuje zerowanie, wyprowadzające mikrokontroler ze stanu zawieszenia programu. Układ ten jest konfigurowalny w szerokim zakresie, a jeżeli jego funkcja nadzorczy działania programu nie jest potrzebna, to da się wykorzystać do generowania stabilnej podstawy czasu dla innych zadań. Sam 16-bitowy licznik WDTCNT (*WatchDog Timer CouNter*) nie jest bezpośrednio dostępny programowo, czyli nie da się wprost odczytać jego stanu ani niczego do niego zapisać; można go tylko zatrzymać albo wyzerować. Operacje przez niego realizowane są kontrolowane za pośrednictwem rejestru sterującego WDTCTL (*Watch Dog ConTrol*), dostęp do którego jest chroniony rodzajem hasła. Zabezpiecza to przed niepożądaną zmianą działania układu w przypadku zawieszenia programu i wykonywania przezeń jakichś przypadkowych instrukcji. Każde pogwałcenie hasła skutkuje natychmiastowym zerowaniem mikrokontrolera. Źródłem sygnału zegarowego dla licznika jest do wyboru albo ACLK, albo SMCLK. Przepiętnienie licznika, generujące zerowanie systemowy lub przerwanie, jest kontrolowane na jego bicie szóstym, dziewiątym, trzynastym albo piętnastym. Ponieważ normalnie po przepiętnieniu licznik jest zerowany, można powiedzieć, że istnieje możliwość wyboru liczby taktów zegarowych wywołujących przepiętnienie pomiędzy 64, 512, 8192 i 32768. Aby zapobiec przypadkowemu wygenerowaniu sygnału zerującego, zmiana długości licznika powinna być łączona z równoczesnym jego wyzerowaniem, co daje się zrobić w jednej instrukcji. Podobnie przy zmianie źródła sygnału taktującego należy licznik chwilowo zatrzymać. Gdy wybrana jest funkcja periodycznego odliczania czasu, przepiętnienie licznika ustawia odpowiedni znacznik (WDTIIFG, *WatchDog Timer Interrupt FlaG*) i może wygenerować przerwanie, jeżeli tylko jest ono odblokowane bitem WDTIE (*WatchDog Timer Interrupt Enable*) oraz ustawiony jest bit ogólnego włączenia systemu przerwań GIE (*General Interrupt Enable*). Przejście do procedury obsługi przerwania automatycznie zeruje znacznik WDTIIFG. Wektor tego przerwania jest inny, niż w trybie nadzorującym (czyli nie jest to wektor resetu). Znaczniki WDTIIFG oraz WDTIE są umieszczone odpowiednio w rejestrach IFG1 oraz IE1. Po resecie, niezależnie od jego przyczyn, licznik jest automatycznie ustawiany jako *watchdog*, a źródłem zegara staje się generator RC (*DCOCLK*). Jeżeli w ciągu około 32 ms nie nastąpi programowa zmiana funkcji, wyzerowanie lub zatrzymanie licznika, zostanie wygenerowany sygnał zerujący i sytuacja będzie się powtarzać.

Niezależnie od źródła, częstotliwość sygnału zegarowego licznika ulega podzieleniu w związanym z licznikiem indywidualnym prescalerze o współczynniku podziału 1, 2, 4 lub 8. W efekcie minimalna

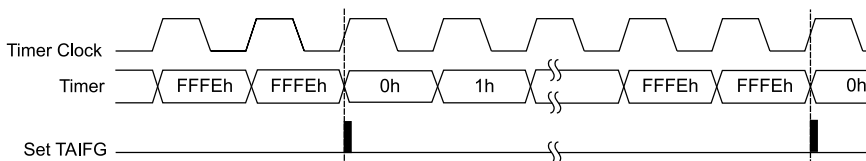
częstotliwość taktowania licznika sygnałem wewnętrznym mikrokontrolera to zaledwie 512 Hz, podczas gdy maksymalna sięga 16 MHz. Tak ogromnego zakresu wyboru nie zapewnia chyba żaden inny

mikrokontroler! Przy wykorzystaniu pełnej pojemności przepiętnienia licznika następuje po czasie zawierającym się pomiędzy 4 ms a 128 sekund. Pokrywa to w praktyce wszystkie potencjalne potrzeby. Możliwość uzyskiwania szerokiego zakresu odliczanych czasów istnieje także w wielu innych mikrokontrolerach, ale wiąże się ze zmianami częstotliwości zegarowej jednostki centralnej. Gdy potrzebne jest odmierzenie długich czasów, częstotliwość pracy rdzenia musi być po prostu mniejsza. W MSP430 są to parametry całkowicie niezależne – jednostka centralna może być taktowana częstotliwością 4 kHz, a licznik 16 MHz, albo odwrotnie: rdzeń 16 MHz, a licznik 512 Hz. Za jedyny mankament można uznać brak możliwości wyboru aktywnego zbocza sygnału zegarowego licznika, zawsze zmienia on stan na zboczu narastającym. Wybór zbocza bywa jednak potrzebny w praktyce niezmiernie rzadko.

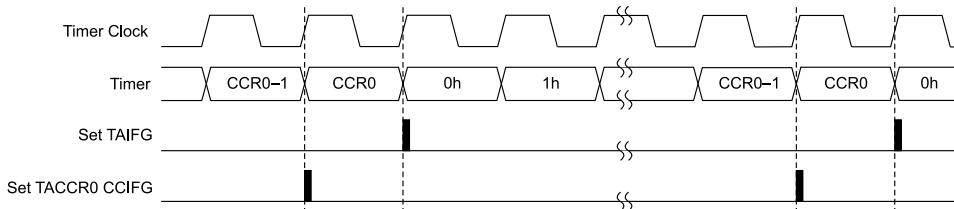
Układ *Licznika\_A* może pracować w 3 trybach:

- licząc w sposób ciągły w górę od stanu 0000 do FFFFh i ponownie od 0000 (praca ciągła – *Continuous Mode*, **rys. 9**).
- licząc w górę od stanu 0000 do wartości zawartej w rejestrze porównania zerowego bloku *capture/compare* TACCR0 (*Timer\_A Capture/Compare Register #0*) i ponownie od zera – *Up Mode* (**rys. 10**).
- licząc w górę od stanu 0000 do zawartości rejestru TACCR0, a następnie z powrotem w dół aż do zera, po czym cykl taki jest ciągle powtarzany (*Up/Down Mode*, **rys. 11**).

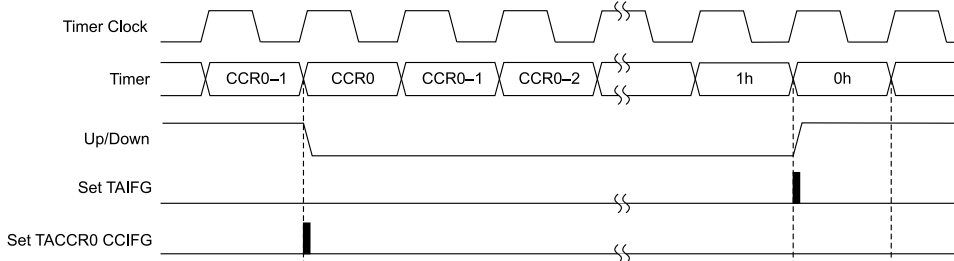
W trybie pracy ciągłej przejście ze stanu FFFFh do zera ustawia odpowiedni znacznik TAIFG i generuje przerwanie, jeżeli nie jest zablokowane. Aby to nastąpiło, musi mieć miejsce faktyczna zmiana stanu licznika poprzez wejście zegarowe; samo wpisanie do licznika wartości zerowej nie wystarcza. Taka sama akcja ma miejsce w trybie *Up* przy przejściu ze stanu maksymalnego (czyli zawartości rejestru TACCR0) do zera. O jeden takt zegara wcześniej, po zrównaniu stanu licznika z zawartością rejestru TACCR0, przyjmuje wartość jeden inny



Rys. 9. Ustawianie znacznika przepiętnienia licznika w trybie pracy ciągłej (*Continuous mode*)



Rys. 10. Ustawianie znaczników w trybie liczenia w górę (*Up mode*)



Rys. 11. Ustawianie znaczników w trybie pracy dwukierunkowym (*Up/Down*)

# OSCYSKOPY RĘCZNE

**HPS10**
**10MHz**

- częstotliwość próbkowania 10 Ms/s
- pasmo analogowe do 2 MHz
- maksymalne napięcie wejściowe 100 V
- pełny auto-setup
- odczyt DVM z opcją x10
- obliczanie mocy audio (rms i peak)
- pomiar dBm, dBV, DC, rms...
- odczyt częstotliwości
- zapis sygnału (2 pamięci)
- wyświetlacz LCD niebieski z podświetleniem: 128x64 pikseli
- wbudowany układ ładowania akumulatorów

Zestaw zawiera:

- oscyloskop HPS10SE
- instrukcję w języku polskim
- izolowaną sondę pomiarową PROBE60S
- walizkę
- zasilacz 9 V/500 mA

**799 zł**
**HPS40**
**40MHz**

- częstotliwość próbkowania 40 Ms/s
- pasmo analogowe do 12 MHz
- maksymalne napięcie wejściowe 100 V
- tryb multimetru z odczytem wartości dBm, dBV, DC, rms
- pomiar mocy audio
- markery dla napięcia i czasu
- wskazanie częstotliwości
- wyświetlacz LCD o wysokiej rozdzielczości 192x112 punktów z podświetleniem
- wbudowany układ ładowania akumulatorów
- izolowane optycznie wyjście do PC – standard RS232

Zestaw zawiera:

- oscyloskop HPS40
- instrukcję w języku polskim
- futerał
- izolowaną sondę pomiarową PROBE60S
- przewód RS232
- walizkę
- zasilacz 9 V/500 mA

**1699 zł**

znacznik TACCR0 CCIFG i także może zostać wywołane przerwanie (ale inne). Znacznik ten jest identycznie ustawiany także w trybie *Up/down*, jednak flaga przepełnienia licznika TAIFG jest wtedy uaktywniana w fazie liczenia w dół, przy przejściu ze stanu 0001 do 0000.

Zatrzymać licznik można wybierając nie ujęty powyżej tryb pracy „stop” albo też wybierając tryb *Up* lub *Up/down* i wpisując do rejestru TACCR0 same zera. Każdy wpis do rejestru licznika TAR ma skutek natychmiastowy, a odczyt jest zawsze poprawny, gdy licznik jest taktowany sygnałem wewnętrznym. Jeżeli źródło sygnału zegarowego jest zewnętrzne, asynchroniczne względem jednostki centralnej, to przy odczycie nie da wykluczyć przypadkowego trafienia dokładnie w moment zmiany stanu licznika i odczyt może być wówczas błędny. W takiej sytuacji lepiej przed odczytem licznik chwilowo zatrzymać.

Gdy licznik liczy w górę w trybie *Up* albo *Up/down*, wpis do rejestru TACCR0 liczby mniejszej od aktualnego stanu skutkuje jego natychmiastowym wyzerowaniem.

Pozostałe bloki *capture/compare* działają w standardowy sposób; w rejestrze konfiguracyjnym można wybrać dla danego rejestru tylko jedną z tych opcji. W trybie ciągłym blok o numerze 0 działa tak, jak pozostałe. W innych trybach decyduje on o długości cyklu zliczania. Przy wybraniu opcji *compare* do wyboru pozostaje jedna z aż siedmiu możliwych akcji podejmowanych na zewnętrznym wyprowadzeniu (*output modes*), a nie tylko jego permanentne wyzerowanie (*reset*), ustawienie (*set*) albo zmiana na stan przeciwny (*toggle*). Zależy to od trybu pracy licznika i generalnie sprowadza się do dodatkowych zmian stanu wyjścia w momencie uaktywnienia uprzywilejowanego bloku zerowego. Przykładowo możliwa jest zmiana stanu wyjścia na przeciwny, gdy stan licznika zrówna się z zawartością rejestru TACCR1 i wyzerowanie tego wyprowadzenia po osiągnięciu przez licznik wartości z rejestru TACCR0. W efekcie blok wyjściowy zapewnia niespotykaną w innych rozwiązaniach możliwość kształtowania przebiegu wyjściowego i ogromną elastyczność zastosowań. W trybie *Up/down* porównanie następuje zarówno przy

liczeniu w górę, jak i w dół. Pozwala to na uzyskanie w ten sposób przebiegu impulsowego o zmiennym wypełnieniu, symetrycznym względem zarówno początku, jak i środka cyklu zliczania. Wykorzystując dwa lub więcej układy *compare* da się także łatwo wygenerować kilka przebiegów impulsowych nigdy nie nachodzących na siebie.

Funkcja *capture* jest aktywowana z boczku na narastającym, opadającym, albo na każdym z nich. Ponieważ zewnętrzne zdarzenie wywołujące zapamiętanie stanu licznika w rejestrze TACCRx może być całkowicie asynchroniczne względem jego sygnału zegarowego, co skutkować może zapisaniem stanu przypadkowego, przewidziano możliwość synchronizacji wpisu najbliższym z boczkiem zegara. Wtedy wpis jest zawsze poprawny. Gdy kolejne aktywności bloku *capture* występują w odstępach porównywalnych z czasami obsługi programowej tych zdarzeń, istnieje prawdopodobieństwo pominięcia jakiegoś odczytu i „nadpisanie” poprzedniej zawartości rejestru nową. Sytuacja taka jest wykrywana przez układ sterowania blokiem *capture* i jest ustawiany odpowiedni znacznik, który może być sprawdzany programowo.

Z układem licznikowo-czasowym związane są przerwania z dwoma odrębnymi wektorami: jeden przyporządkowany jest przerwowaniu z układu *capture/compare\_0* TACCR0 CCIFG, oraz drugi, TAIV (*Timer\_A Interrupt Vector*), przydzielony w kolejności priorytetów obsługi blokom *capture/compare\_1*, *capture/compare\_2* oraz przepełnieniu licznika. Aby zidentyfikować konkretne źródło przerwania, programista powinien sprawdzić zawartość rejestru TAIV, zawierającego odpowiednie znaczniki. Najprościej dodawać jego zawartość – bezpośrednio albo po jakiejś modyfikacji – do licznika rozkazów, co automatycznie pociąga za sobą skok do właściwej procedury. Po raz kolejny obrazuje to efektywność jednostki centralnej w MSP430; sztuczka taka rzadko bywa możliwa w innych mikrokontrolerach. Jakikolwiek zapis lub odczyt rejestru TAIV kasuje automatycznie znacznik przerwania o najwyższym priorytecie. Jeżeli zatem ustawione były dwa znaczniki, z bloku *capture/compare\_1* i *capture/compare\_2*, a w programie obsługi

W bardziej rozbudowanych wersjach mikrokontrolerów z tej rodziny występuje jeszcze drugi, bardzo podobny układ licznika B. Współpracują z nim trzy, lub nawet aż siedem bloków *capture/compare* i różni się w pewnych szczegółach od licznika A. W szczególności może mieć programowaną długość 8, 10, 12 lub 16 bitów, a skojarzone z nim wyjścia mogą przyjmować nie tylko stan niski albo wysoki, ale także być wprowadzane w stan wysokiej impedancji.

przerwania wystąpił odczyt rejestru TAIV (co wydaje się być oczywistością), to po instrukcji powrotu RETI pozostanie ustawiony już tylko ten drugi i natychmiast, bez żadnej zbędnej zwłoki, nastąpi uruchomienie kolejnego przerwania.

Z układem *Licznika\_A* jest związanych 9 rejestrów:

- rejestr licznika TAR (*Timer\_A counter*),
- rejestr sterujący TACTL (*Timer\_A control*),
- rejestr bloku przechwytyjąco/porównującego nr 0 TACCR0 (*Timer\_A capture/compare 0*),
- rejestr konfigurujący ten blok TACCTL0 (*Timer\_A capture/compare control 0*),
- rejestr bloku przechwytyjąco/porównującego nr 1 TACCR1 (*Timer\_A capture/compare 1*),
- rejestr konfigurujący ten blok TACCTL1 (*Timer\_A capture/compare control 1*),
- rejestr bloku przechwytyjąco/porównującego nr 2 TACCR1 (*Timer\_A capture/compare 2*),
- rejestr konfigurujący ten blok TACCTL2 (*Timer\_A capture/compare control 2*),
- rejestr przerwań Timer\_A interrupt vector TAIV (*Timer\_A interrupt vector*).

Wszystkie te rejestry są zerowane po wyzerowaniu mikrokontrolera.

Jak wynika z powyższej prezentacji, układ licznikowo-czasowy A w MSP430 jak żaden inny wyróżnia się wyjątkową elastycznością i różnorodnością możliwych do wyboru opcji. Trzeba się natrudzić, aby znaleźć jakieś mankamenty – przychodzi do głowy jedynie brak możliwości wyboru aktywnego zbozca zegarowego i brak modulacji  $\Sigma/\Delta$  PWM. Ten drugi można jednak w stosunkowo prosty sposób ominąć programowo.

**Maciej Nowiński**