



## Podstawy Techniki Mikroprocesorowej Laboratorium

### Ćwiczenie 3

Liczniki 0, 1, 2

*(Timer Counters T/C0, T/C1, T/C2)*

#### Program ćwiczenia:

- obsługa trybu pracy „normalny” wybranego licznika,
- obsługa trybu pracy „CTC” wybranego licznika,
- obsługa trybu pracy „PWM” wybranego licznika.

#### Zagadnienia do przygotowania:

- liczniki T/C0, T/C1, T/C2 – podstawowe różnice,
- tryby pracy: normalny, CTC, PWM – sposób działania, zastosowania,
- generowanie sygnału o zadanej częstotliwości – konfiguracja licznika,
- generowanie sygnału o zadanej częstotliwości i współczynnika wypełnienia – konfiguracja licznika,
- wejścia oraz wyjścia liczników,
- rejestry sterujące – rodzaje, zastosowanie,
- przerwania liczników.

#### Literatura:

- [1] Wykład.
- [2] Dokumentacja mikrokontrolera ATmega8535, [www.atmel.com](http://www.atmel.com).
- [3] Mikrokontrolery AVR ATmega w praktyce, R. Baranowski, BTC 2005.
- [4] AVR i ARM7. Programowanie mikrokontrolerów dla każdego, P. Borkowski, Helion 2010.

## Zawartość instrukcji

|   |    |
|---|----|
| 1. Wstęp .....  | 2  |
| 2. Sposób pracy liczników .....   | 2  |
| 3. Liczniki – konfiguracja i sterowanie.....                                    | 3  |
| 3.1. Rejestry.....  | 3  |
| 3.2. Wejścia / wyjścia.....   | 4  |
| 3.3. Przerwania .....   | 5  |
| 4. Tryby pracy liczników .....  | 5  |
| 4.1. Tryb normalny .....  | 5  |
| 4.2. Tryb CTC ( <i>Clear Timer on Compare</i> ) .....                           | 6  |
| 4.3. Tryb PWM ( <i>Pulse Width Modulation</i> ).....                            | 6  |
| 5. Opis rejestrów sterujących .....   | 7  |
| 5.1. Rejestry kontrolne TCCR (Timer/Counter Control Register):.....             | 7  |
| 5.1.1. Liczniki T/C0 oraz T/C2.....   | 7  |
| 5.1.2. Licznik T/C1 .....   | 8  |
| 5.2. Rejestry zliczające TCNT (Timer/CouNter Register): .....                   | 8  |
| 5.3. Rejestry porównawcze OCR (Output Compare Register): .....                  | 9  |
| 5.4. Rejestr maski (aktywacji) przerw TIMSK (TimerInterruptMaSK register):..... | 9  |
| 5.5. Pozostałe rejestry .....   | 9  |
| 6. Urządzenia wspomagające zajęcia laboratoryjne .....                          | 9  |
| 6.1. Brzęczyk (buzzer) .....  | 9  |
| 6.2. Serwomechanizm .....   | 9  |
| 7. Programy przykładowe .....   | 10 |
| 7.1. T/C0, tryb pracy normalny.....   | 10 |
| 7.2. T/C2, tryb pracy CTC.....  | 11 |
| 8. Przykładowe zadania.....   | 11 |

UWAGA! Do poprawnej obsługi liczników niezbędne jest korzystanie z dokumentacji mikrokontrolera.

Instrukcja zawiera pewne uproszczenia w stosunku do rzeczywistych możliwości mikrokontrolera.

W tekście znajdują się odniesienia do tabel i rysunków znajdujących się w dokumentacji mikrokontrolera ATmega8535. Przykład przyjętych oznaczeń: Tab. 43/85 odnosi się do tabeli 43 na stronie 85.

Znak „*n*” pojawiający się w instrukcji oznacza cyfrę za zakresu 0-2, np. TCCRN oznacza rejestr TCCR0 lub TCCR1 lub TCCR2, rzadziej literę z zakresu A-B (w zależności od kontekstu).

## 1. Wstęp

Podstawową funkcją liczników (Timer/Counter, T/Cn) jest zliczanie doprowadzonych impulsów. Jednak zazwyczaj liczniki pełnią jeszcze jedną funkcję w mikrokontrolerze (MCU, *MicroController Unit*) – generowanie przebiegów prostokątnych o zadanej częstotliwości lub o zadanym współczynniku wypełnienia.

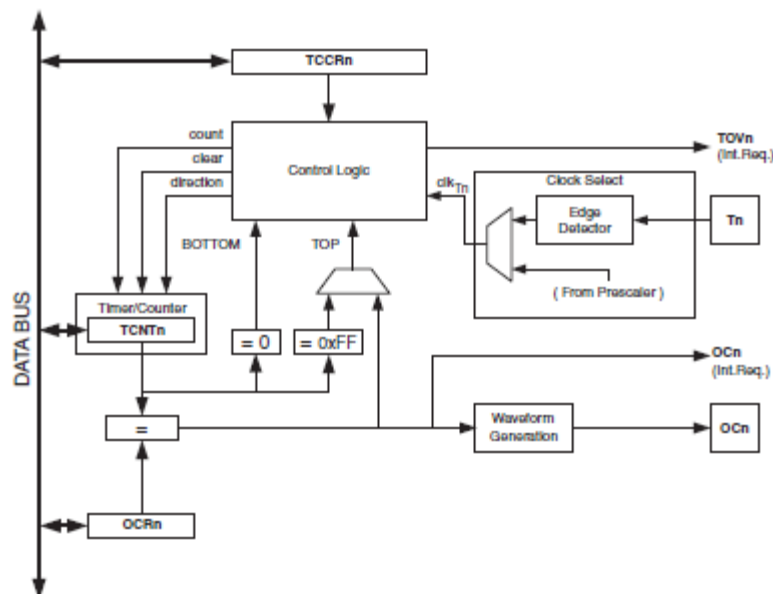
Mikrokontroler ATmega8535 został wyposażony w 3 liczniki: T/C0, T/C1 oraz T/C2. Charakteryzują się one podobnymi funkcjami, podobną zasadą działania i są sterowane w niemal identyczny sposób. Podstawowe różnice między nimi to:

|  | T/C0                     |               | T/C1  |               | T/C2  |               |
|--|--------------------------|---------------|---|---------------|---|---------------|
| <b>Zakres zliczania</b>  | 8-bitowy                 |               | 16-bitowy   |               | 8-bitowy  |               |
| <b>Tryby pracy</b>   | 4 (normalny, CTC, 2xPWM) |               | 15 (normalny, 2xCTC, 12xPWM)                        |               | 4 (normalny, CTC, 2xPWM)                            |               |
| <b>Praca</b>   | Synchroniczna            |               | Synchroniczna                                       |               | synchroniczna i asynchroniczna                      |               |
| <b>Wejścia dla sygnału zewnętrznego (Tn)</b>                         | 1 (T0)                   |               | 2 (T1, ICP1)  |               | 0 (brak możliwości zliczanie impulsów zewnętrznych) |               |
| <b>Wyjścia dla generowanego przebiegu prostokątnego</b>              | 1 (OC0)                  |               | 2 (OC1A, OC1B)                                      |               | 1 (OC2)   |               |
| <b>Dostępne dzielniki częstotliwości taktującej (prescaler)</b>      | 5                        |               | 5   |               | 7   |               |
| <b>Przerwanie od przepelnienia (TOVn)</b>                            | 1                        |               | 1   |               | 1   |               |
| <b>Przerwania od zrównania stanu licznika z rejestrem OCRn (OCn)</b> | 1                        |               | 2   |               | 1   |               |
| <b>Rejestry sterujące</b>  | TCCR0<br>TCNT0<br>OCR0   | TIMSK<br>TIFR | TCCR1A<br>TCCT1B<br>TCNT1<br>OCR1A<br>OCR1B<br>ICR1 | TIMSK<br>TIFR | TCCR2<br>TCNT2<br>OCR2<br>ASSR2                     | TIMSK<br>TIFR |

## 2. Sposób pracy liczników

Liczniki mogą być taktowane dwoma rodzajami sygnałów (asynchroniczny T/C2 – trzema). Są to impulsy taktujące mikrokontroler (ustawienia fabryczne dla ATmega8535 to 1 MHz) lub impulsy doprowadzone z zewnątrz przez wejście Tn (np. T0). Są one doprowadzane do bloku *ClockSelect* (Fig. 27/71), gdzie przy pomocy multiplexera następuje ostateczny wybór rodzaju taktowania. Multiplexer sterowany jest przy pomocy bitów CSxx (zob. opis rejestru TCCR0).

Figure 27. 8-bit Timer/Counter Block Diagram



Sygnal taktujący licznik ( $clk\_Tn$ ) dociera do układu sterującego urządzeniem (*Control Logic*). Każdy impuls zwiększa aktualny stan licznika (stan rejestru  $TCNTn$ ) o 1. Jeżeli rejestr  $TCNTn$  osiągnie wartość maksymalną (np. 255 dla licznika 8-bitowego) i zostanie zwiększony jeszcze o 1 nastąpi jego przepełnienie – zostanie wyzerowany, ustawiona zostanie flaga przepełnienia  $TOVn$ . Jeżeli aktualny stan rejestru  $TCNTn$  zrówna się ze stanem rejestru  $OCRn$  zostanie ustawiona flaga informująca o zrównaniu ( $OCn$ ).

W trybach pracy związanych z generowaniem sygnału prostokątnego zrównanie  $TCNTn$  oraz  $OCRn$  wykorzystuje się do formowania przebiegu na wyjściu  $OCn$  (pojawia się zbocze narastające lub opadające).

### 3. Liczniki – konfiguracja i sterowanie

#### 3.1. Rejestry

Do sterowania licznikami wykorzystuje się rejestry. Rejestry poszczególnych liczników mają zbliżone nazwy (różnią się tylko ostatnimi znakami – oznaczającymi, do którego licznika należą), ponieważ w obrębie danego licznika pełnią identyczne funkcje. I tak:

- **TCCRn** – rejestry konfiguracyjne; tu określamy sposób pracy licznika itp; dokładny opis funkcji poszczególnych bitów znajduje się w dalszej części instrukcji;
- **TCNTn** – rejestry zliczające; czyli aktualny stan licznika; można zmodyfikować stan T/Cn wpisując tutaj żadaną wartość;

- **OCRn** – rejestry porównawcze; pod warunkiem uruchomienia odpowiedniego trybu pracy można w nich określić do jakiej wartości licznik ma zliczać; rejestry te biorą udział w formowaniu przebiegu prostokątnego generowanego na wyjściu OCn;
- **TIMSK** – rejestr maski przerw liczników (odblokowywanie przerw); wspólny dla wszystkich liczników; dokładny opis funkcji poszczególnych bitów znajduje się w dalszej części instrukcji;
- **TIFR** – flagi przerw liczników (informacja o spełnieniu warunku przerwania); wspólny dla wszystkich liczników; dokładny opis funkcji poszczególnych bitów znajduje się w dalszej części instrukcji;
- **ICR1** – tylko T/C1; rejestr zatraskujący (zapamiętujący) aktualny stan licznika; zatraskiwanie stanu wywoływane sygnałem zewnętrznym;
- **ASSR2** – tylko T/C2; rejestr konfiguracyjny dla pracy asynchronicznej; dokładny opis w dokumentacji MCU.

W rejestrach TCCRn, TIMSK oraz TIFR każdy bit służy do uruchomienia określonej funkcjonalności licznika.

W rejestrach TCNTn, OCRn oraz ICR1 wszystkie bity służą do tego samego – przechowują wartość liczbowa.

### 3.2. Wejścia / wyjścia

Wejścia / wyjścia liczników możemy podzielić na 4 grupy. Wejścia (3 grupy) służą do doprowadzania do licznika zliczanych (taktujących) impulsów lub do wyzwolenia funkcji specjalnej (pin ICP1). Wyjścia służą do generowania przebiegu prostokątnego o zadanej częstotliwości lub zadany współczynnik wypełnienia. I tak:

- **T0, T1** – wejścia odpowiednio dla T/C0 oraz T/C1. Jeżeli w rejestrze sterującym TCCRn ustawione zostanie odpowiedni sposób taktowania (bity CSn) to licznik zlicza sygnały (zbocza narastające lub opadające) doprowadzone na to wejście.
- **ICP1** – wejście T/C1. Pojawienie się sygnału na tym wejściu powoduje skopiowanie aktualnego stanu licznika (TCNT1) do rejestru zatraskującego (ICR1).
- **TOSC1, TOSC2** – wejścia licznika T/C2. Między te piny należy podłączyć dodatkowy oscylator zewnętrzny, jeżeli licznik2 ma pracować asynchronicznie (tzn. niezależnie od głównego zegara taktującego MCU).
- **OCn** – wyjścia. Na nich pojawi się formowany przebieg prostokątny. W celu zapewnienia poprawnej pracy koniecznie muszą zostać wysterowane jako wyjście (rejestr DDRn).

### 3.3. Przerwania

Liczniki wykorzystują 3 różne przerwania wewnętrzne informujące o ich aktualnym stanie:

- **TOVn** – *TimerOverflow*, czyli informacja o przepełnieniu licznika. Rejestr TCNTn jest już pełny i zwiększa się jeszcze o jeden. W rezultacie zostaje wyzerowany a informacja o tym zdarzeniu pojawia się na odpowiednim bicie w rejestrze TIFR. Jeżeli został ustawiony odpowiedni bit w rejestrze TIMSK (oraz odblokowano globalnie przerwania) MCU przejdzie do obsługi przerwania.
- **OCn** – *OutputCompare*, czyli informacja o zrównaniu stanu rejestru zliczającego TCNTn z wartością zapisaną w rejestrze porównawczym OCRn. Pojawia się na odpowiednim bicie w rejestrze TIFR. Jeżeli został ustawiony odpowiedni bit w rejestrze TIMSK (oraz odblokowano globalnie przerwania) MCU przejdzie do obsługi przerwania.
- **ICPn** – *Input Capture*, czyli informacja o pojawieniu się sygnału wyzwalającego na wejściu ICP1 (lub odpowiedniego wewnętrznego sygnału wyzwalającego). Pojawia się na odpowiednim bicie w rejestrze TIFR. Jeżeli został ustawiony odpowiedni bit w rejestrze TIMSK oraz odblokowano globalnie przerwania MCU przejdzie do obsługi przerwania.

## 4. Tryby pracy liczników

Zgrubnie tryby pracy można podzielić na 3 rodzaje – normalny, CTC, PWM (tryby PWM dzielą się na 3 rodzaje nieco różniące się sposobem działania; szczegółowe informacje można znaleźć w dokumentacji). Każdy z nich można wykorzystać w dowolnym celu (zliczanie, formowanie sygnału), jednak standardowo wykorzystuje się je następująco:

- **normalny** – zliczenie impulsów wewnętrznych (zegarowych) LUB zewnętrznych (wejście Tn);
- **CTC** – formowanie przebiegu prostokątnego o współczynniku wypełnienia 50% i o zadanej częstotliwości;
- **PWM** – formowanie przebiegu prostokątnego o zadany współczynnik wypełnienia i zgrubnie zadanej częstotliwości.

### 4.1. Tryb normalny

Licznik zlicza doprowadzone impulsy. Dla licznika 8-bitowego zakres pracy to  $0 \div 255$ . Aktualny wynik dostępny jest w rejestrze TCNTn. Zliczane mogą być impulsy (Tab. 43/85 – dla T/C0 i T/C1):

- zegarowe – sygnał taktujący mikrokontroler (np. 1 MHz) jest dzielony w układzie *prescalera* (dzielnika częstotliwości) i doprowadzany do układu zliczającego;
- doprowadzone z zewnątrz przez wejście Tn (np. T1) – należy wówczas określić zbczce.

## 4.2. Tryb CTC (*Clear Timer on Compare*)

Licznik działa jak w trybie normalnym, z jedną różnicą – rejestr zliczający TCNTn jest zerowany za każdym razem, gdy jego stan zrówna się ze stanem rejestru OCRn. Powoduje to ograniczenie zakresu pracy licznika (np. dla 8-bitowego T/C0 już nie  $0 \div 255$ , ale  $0 \div \text{OCR0}$ ). Podstawowa zaleta to możliwość zliczania do zadanej wartości (OCR0). O jej osiągnięciu licznik informuje ustawiając flagę przerwania OCn w rejestrze TIFR.

Najważniejszą funkcją tego trybu pracy jest generowanie przebiegu prostokątnego na wyjściu OCn licznika. Jeżeli rejestr kontrolny TCCRn zostanie odpowiednio skonfigurowany (Tab. 40/84; sposób działania pinu OCn – *toggle*) to każde zrównanie stanu rejestrów TCNTn oraz OCRn będzie powodowało zmianę stanu na wyjściu OCn na przeciwny. Schematycznie zostało to przedstawione na Fig. 31/75. Częstotliwość generowanego wówczas przebiegu prostokątnego będzie opisana wzorem przedstawionym w podrozdziale *ClearTimer on CompareMatch (CTC) Mode* dokumentacji (s. 76), gdzie:

$f_{OCn}$  – częstotliwość generowanego przebiegu;

$f_{clk\_I/O}$  – częstotliwość taktowania mikrokontrolera;

$N$  – wybrany dzielnik częstotliwości taktującej licznik (*prescaler*);

$\text{OCRn}$  – wartość wpisana do rejestru OCRn.

**Uwaga:** dla liczników T/C0 oraz T/C2 zakres pracy ustala się w odpowiednich rejestrach OCRn. Dla T/C1 służy do tego jeden z dwóch rejestrów (do wyboru) – OCR1A lub ICR1 (rejestr OCR1B nie może być użyty w trybie CTC).

Przykładowo, mikrokontroler taktowany częstotliwością 8 MHz, po ustawieniu *prescalera* na 1024 i wpisaniu do rejestru OCRn wartości „123” będzie formował na wyjściu OCn przebieg prostokątny o częstotliwości 31,5 Hz. Należy pamiętać o wysterowaniu OCn jako wyjście (odpowiedni rejestr DDRn).

## 4.3. Tryb PWM (*Pulse Width Modulation*)

Wyróżniamy 2 rodzaje trybów PWM – jednozbozowe (tylko zbocze narastające – np. tryb FAST PWM) oraz dwuzbozowe (narastające i opadające – np. tryb PC PWM). W kontekście laboratorium różnice między nimi są nieznaczne i ograniczają się do wzorów określających częstotliwość generowanego przebiegu. Tryby PWM służą do formowania na wyjściu OCn przebiegu prostokątnego o zadanej częstotliwości (jedynie zgrubnie) oraz o zadanym współczynniku wypełnienia (0-100%, dokładność nawet do 0,0015%). Sposób uruchomienia trybu oraz formowania sygnału na wyjściu zachodzi analogicznie jak w trybie CTC (zob. Fig. 32/77 lub Fig. 33/78). Również tutaj należy dokonać odpowiednich ustawień w rejestrach TCCRn oraz OCRn, należy pamiętać o wysterowaniu

OCn jako wyjście. Częstotliwość generowanego sygnału PWM określana jest jednym z 5 wzorów podanych w określonym podrozdziale dokumentacji:

- podrozdział Fast PWM Mode (F-PWM) dla T/C0 lub T/C2;
- podrozdział Fast PWM Mode (F-PWM) dla T/C1;
- podrozdział PhaseCorrect PWM Mode (PC-PWM) dla T/C0 lub T/C2;
- podrozdział PhaseCorrect PWM Mode (PC-PWM) dla T/C1;
- podrozdział Phase and Frequency Correct PWM Mode (PFC-PWM) dla T/C1.

Wykorzystując licznik T/C0 można ustalić jedynie 5 różnych częstotliwości dla F-PWM i 5 dla PC-PWM. Przy użyciu T/C2 – po 7 (w T/C0 i T/C2 decyduje o tym liczba możliwych ustawień *prescalera*). Licznik T/C1 pozwala uzyskać znacznie lepszą dokładność, ponieważ częstotliwość ustalana jest nie tylko na podstawie *prescalera*, ale również rejestru określanego w dokumentacji [2] jako TOP (OCR1A, ICR1 lub wartość stała, w zależności od wybranego trybu pracy).

Współczynnik wypełnienia dla sygnału PWM określa się dokonując wpisu do rejestru OCRn. Liczniki mogą pracować w zwykłym lub odwróconym trybie PWM. W przypadku trybu zwykłego wpisanie do OCRn wartości '0' powoduje formowanie sygnału o współczynniku wypełnienia 0%. Wpisanie wartości równej rejestrowi określonemu jako TOP – o współczynniku wypełnienia 100%. Pośrednie współczynniki wypełnienia można wyznaczyć z prostej proporcji. W przypadku trybu odwróconego sytuacja wygląda odwrotnie:  $OCRn = 0 \Rightarrow 100\%$ ,  $OCRn = TOP \Rightarrow 0\%$ .

Przykładowo, mikrokontroler taktowany częstotliwością 4 MHz, po ustawieniu *prescalera* na 8, wpisaniu do rejestru OCRn wartości „50” i wybraniu trybu pracy „odwrócony” F-PWM licznika T/C2 będzie formował na wyjściu OCn przebieg prostokątny o częstotliwości 1953 Hz oraz współczynniku wypełnienia 80,4%. Należy pamiętać o wysterowaniu OCn jako wyjście (odpowiedni rejestr DDRn).

## 5. Opis rejestrów sterujących

Poniżej przedstawiono skrócony opis poszczególnych bitów. Szczegółowe informacje zawarte są w odpowiednich w tabelkach i akapitach w podrozdziale *8-bit Timer/Counter Register Description* dokumentacji.

Bity oznaczone kolorem szarym nie są użyteczne w kontekście liczników.

Bity są numerowane (0÷7).

Wyróżniamy bity tylko do odczytu (R), tylko do zapisu (W) lub do zapisu/odczytu (R/W).

### 5.1. Rejestry kontrolne TCCR (Timer/Counter Control Register):

#### 5.1.1. Liczniki T/C0 oraz T/C2

Oba liczniki mają po jednym rejestrze kontrolnym. Układ i funkcje bitów w TCCR0 oraz TCCR2 są identyczne. Pełny opis można znaleźć w dokumentacji ATmega8535, odpowiednio na stronach 82-84 (T/C0) oraz 127-129 (T/C2).



**TCCR0, TCCR2**

| 7    | 6     | 5     | 4     | 3     | 2    | 1    | 0    |
|------|-------|-------|-------|-------|------|------|------|
| FOCn | WGMn0 | COMn1 | COMn0 | WGMn1 | CSn2 | CSn1 | CSn0 |
| W    | R/W   | R/W   | R/W   | R/W   | R/W  | R/W  | R/W  |

**FOCn** – ręczne sterowanie stanem wyjścia OCn (działa tylko w trybach normalnych i CTC).

**WGMn1:0** – wybór trybu pracy (normalny, CTC, Fast PWM, PhaseCorrect PWM – zob. tab. w dokumentacji).

**COMn1:0** – wybór sposobu działania wyjścia OCn (zob. tab. w dokumentacji); dla generowania przebiegu prostokątnego w trybie CTC sugerowana opcja to „toggle” – zmiana stanu na przeciwny.

**CSn2:0** – wybór sposobu taktowania licznika (zob. tab. w dokumentacji): jeden z kilku możliwych dzielników częstotliwości lub taktowanie sygnałem dochodzącym przez wejście T0 (tylko T/C0!)

**5.1.2. Licznik T/C1**

Licznik 1 jest 16-bitowy. Oferuje więcej możliwości niż pozostałe dwa. Wiąże się z tym większa liczba bitów konfiguracyjnych, które rozmieszczono w dwóch rejestrach kontrolnych – TCCR1A oraz TCCR1B. Ich pełny opis można znaleźć w dokumentacji ATmega8535 na stronach 109-112.

**TCCR1A**

| 7      | 6      | 5      | 4      | 3     | 2     | 1     | 0     |
|--------|--------|--------|--------|-------|-------|-------|-------|
| COM1A1 | COM1A0 | COM1B1 | COM1B0 | FOC1A | FOC1B | WGM11 | WGM10 |
| R/W    | R/W    | R/W    | R/W    | W     | W     | R/W   | R/W   |

**TCCR1B**

| 7     | 6     | 5 | 4     | 3     | 2    | 1    | 0    |
|-------|-------|---|-------|-------|------|------|------|
| ICNC1 | ICES1 | - | WGM13 | WGM12 | CS12 | CS11 | CS10 |
| R/W   | R/W   | R | R/W   | R/W   | R/W  | R/W  | R/W  |

**FOC1A,B** – jak w T/C0. W T/C1 są 2 wyjścia (OC1A, OC1B), zatem 2 bity FOC1.

**WGMn3:0** – jak w T/C0. Jest 15 trybów pracy, które możemy podzielić na: normalne, CTC, PWM.

**COM1A1:0,COM1B1:0** – jak w T/C0. Są 2 wyjścia (OC1A, OC1B), zatem 2 pary bitów COM1.

**CSn2:0** – jak w T/C0.

**ICES1** – wybór zbocza aktywującego układ *Input Capture* – wejście IC1.

**ICNC1** – aktywacja zabezpieczenia przed zakłóceniami na wejściu IC1 dla układu *Input Capture*.

**5.2. Rejestry zliczające TCNT (Timer/CouNter Register):**

Dla każdego z trzech liczników mamy jeden rejestr, w którym odbywa się zliczanie:

- 8-bitowy TCNT0 dla T/C0,
- 16-bitowy TCNT1 dla T/C1 (podzielony na TCNT1H, TCNT1L),
- 8-bitowy TCNT2 dla T/C2.

### 5.3. Rejestry porównawcze OCR (Output Compare Register):

Dla każdego z wyjść OCn liczników mamy po jednym rejestrze OCR. W umieszcza się wartość, do której licznik ma doliczyć. Wykorzystywane m.in. do określania częstotliwości generowanego przebiegu prostokątnego (tryby CTC) lub współczynnika wypełnienia sygnału (tryby PWM):

- 8-bitowy OCR0 dla T/C0 (wyjście OC0),
- dwa 16-bitowe OCR1A i OCR1B dla T/C1 (podzielone na OCR1nH i OCR1nL) – wyjścia OC1A i OC1B,
- 8-bitowy OCR2 dla T/C2 (wyjście OC2).

### 5.4. Rejestr maski (aktywacji) przerw TIMSK (TimerInterruptMaSK register):

|       |       |        |        |        |       |       |       |
|-------|-------|--------|--------|--------|-------|-------|-------|
| 7     | 6     | 5      | 4      | 3      | 2     | 1     | 0     |
| OCIE2 | TOIE2 | TICIE1 | OCIE1A | OCIE1B | TOIE1 | OCIE0 | TOIE0 |
| R/W   | R/W   | R/W    | R/W    | R/W    | R/W   | R/W   | R/W   |

**TOIE<sub>n</sub>** – aktywacja przerwania informującego o przepelnieniu rejestru zliczającego TCNT<sub>n</sub>.

**OCIE<sub>n</sub>** – aktywacja przerwania informującego o zrównaniu stanu rejestrów TCNT<sub>n</sub> i OCR<sub>n</sub>.

**TICIE1** – aktywacja przerwania związanego z pinem IC1 i układem *Input Capture*.

### 5.5. Pozostałe rejestry

**ICR1** (ICR1H, ICR1L) – dla układu *Input Capture* licznika 1 (dokładny opis – s. 115 dokumentacji),

**ASSR** – dla pracy asynchronicznej licznika 2 (dokładny opis – s. 131 dokumentacji),

**TIFR** – rejestr flag przerw liczników (znaczenie poszczególnych bitów analogiczne do TIMSK).

## 6. Urządzenia wspomagające zajęcia laboratoryjne

### 6.1. Brzęczyk (buzzer)

Wyposażony w dwa wyprowadzenia, które należy podłączyć między wyjście licznika (OCn) a pin GND. Sposób podłączenia nie ma znaczenia. Może zostać wykorzystany do zamiany przebiegu prostokątnego o zadanej przez prowadzącego częstotliwości (np. 1 kHz) na sygnał akustyczny.

### 6.2. Serwomechanizm

Serwomechanizm wykorzystywany w trakcie zajęć laboratoryjnych jest sterowany sygnałem PWM generowanym przy użyciu licznika T/C1. W zależności od współczynnika wypełnienia doprowadzonego przebiegu wskazówka wychyla się w zakresie 0-90° w prawo lub w lewo.

Zadanie polega na doprowadzeniu do serwomechanizmu sygnału o odpowiedniej częstotliwości oraz o zadanym przez prowadzącego współczynniku wypełnienia (ew. na późniejszej zmianie wartości tego współczynnika). W efekcie wskazówka powinna wychylić się o ściśle określony kąt (z dokładnością do ułamków procenta).

Serwomechanizm jest wyposażony w 3 wyprowadzenia:

- Vcc – kolor czerwony, należy podłączyć do napięcia 5 V (pin VCC).
- GND – kolor czarny (brązowy), należy podłączyć do masy (pin GND).
- PWM – kolor żółty, sygnał sterujący, należy podłączyć do źródła sygnału PWM (wyjście OCn).

**UWAGA: błędne podłączenie doprowadzenia Vcc lub GND może skutkować uszkodzeniem mechanizmu. Wymagane sprawdzenie połączenia przez prowadzącego zajęcia.**

Sygnał PWM doprowadzony do serwomechanizmu powinien mieć okres 20 ms. Współczynnik wypełnienia determinuje kąt wychylenia wskazówki. Zakres ruchu to 180°:

- pozycja „minimum” -90° – dla współczynnika wypełnienia 2,7%,
- pozycja „neutralna” 0° – dla współczynnika wypełnienia 8%,
- pozycja „maksimum” +90° – dla współczynnika wypełnienia 13,3%.

Nie należy przekraczać zakresu „minimum÷maksimum”, ponieważ może to doprowadzić do uszkodzenia serwomechanizmu.

## 7. Programy przykładowe

### 7.1. T/C0, tryb pracy normalny

zliczanie impulsów zewnętrznych (zbocze narastające, pin T0) i kopiowanie aktualnego stanu licznika do rejestru R31.

```
ldi R16,0
out DDRB,R16
ldi R16,1
out PORTB,R16 //pull-up na wejściu T0

ldi R16,0b00000111
out TCCR0,R16 //tryb pracy normalny,takt.: zbocze narastające na T0

skacz:
in R31, TCNT0 //kopiowanie aktualnego stanu licznika do rejestru R31
rjmp skacz
```

## 7.2. T/C2, tryb pracy CTC

generowanie przebiegu prostokątnego o częstotliwości  $\sim 1$  kHz.

Określamy wartość prescalera ( $N$ ) oraz wartość, jaką należy wpisać do rejestru OCR2 (założenie – częstotliwość taktująca mikrokontroler to 8 MHz) zgodnie ze wzorem ze strony 122 dokumentacji. Przyjmując prescaler  $N = 8$  wyliczamy:  $OCR2 = 124$ .

```
ldi R16, $80
out DDRD, R16          //pin OC2 wysterowany jako wyjście

ldi R16, 124
out OCR2, R16         //OCR2 = 124

ldi R16, 0b00011011
out TCCR2, R16        //tryb pracy CTC, działanie wyjścia OC2 - toggle,
                    //taktowanie: prescaler N = 32

petla:
nop                   //pusta instrukcja (No Operation)
rjmp petla
```

## 8. Przykładowe zadania

- przy pomocy licznika T/C2 odlicz czas 50 ms (załóż częstotliwość taktującą 1 MHz);
- przy pomocy licznika T/C0 odlicz 500 impulsów zegarowych;
- przy pomocy licznika T/C1 odlicz 50 impulsów zewnętrznych, doprowadzanych przez wejście T1;
- przy pomocy licznika T/C0 wygeneruj przebieg prostokątny o częstotliwości 1,5 kHz, doprowadź go do buzzera;
- przy pomocy licznika T/C2 wygeneruj przebieg PWM o współczynniku wypełnienia 20% i częstotliwości około 1 kHz;
- przy pomocy licznika T/C1 wygeneruj przebieg PWM o współczynniku wypełnienia 5% i okresie 20 ms, wykorzystaj go do sterowania serwomechanizmem.