

Programowanie mikrokontrolerów

Modulacja czasu trwania impulsów

Marcin Engel Marcin Peczarski

Instytut Informatyki Uniwersytetu Warszawskiego

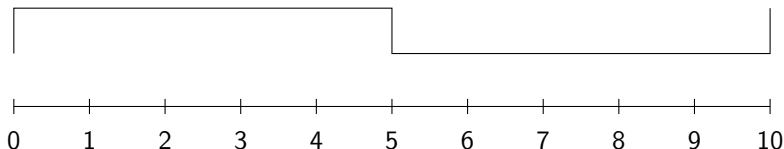
27 listopada 2012

Pulse Width Modulation

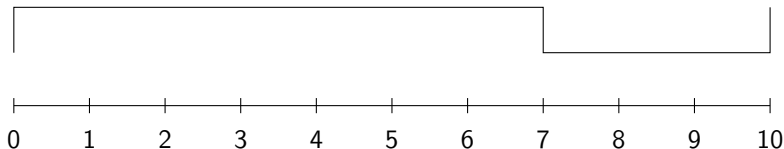
- ▶ Pulse Width Modulation
- ▶ Modulacja czasu trwania impulsów
- ▶ Ważna technika sterowania mocą odbiorników
- ▶ Zastosowania:
 - ▶ regulacja prędkości obrotowej silnika,
 - ▶ regulacja jasności źródła światła,
 - ▶ ustawianie kąta wychylenia serwomechanizmu,
 - ▶ 1-bitowy przetwornik cyfrowo-analogowy,
 - ▶ wzmacniacz akustyczny klasy D,
 - ▶ ...

Czas trwania impulsu

- ▶ Dotychczas generowaliśmy sygnały o współczynniku wypełnienia 50%.
- ▶ Czas trwania fazy wysokiej = czas trwania fazy niskiej sygnału:



- ▶ Można też inaczej, np. współczynnik wypełnienia 70% oznacza, że poziom sygnału przez 70% czasu jest wysoki:



Szybki PWM licznika 0

- ▶ Licznik zlicza zawsze od 0 do 255.
- ▶ Stan wyjścia OC0 zmienia się (w sposób określony przez konfigurację) po osiągnięciu wartości minimalnej oraz po osiągnięciu zgodności.
- ▶ Dokumentacja nie precyzuje, kiedy **dokładnie** dochodzi do zmiany stanu wyjścia OC0 („*at bottom*” i „*on compare match*”), ale nie ma to wielkiego znaczenia.
- ▶ Uzyskiwana częstotliwość:

$$\frac{\text{clk}}{256 \cdot N},$$

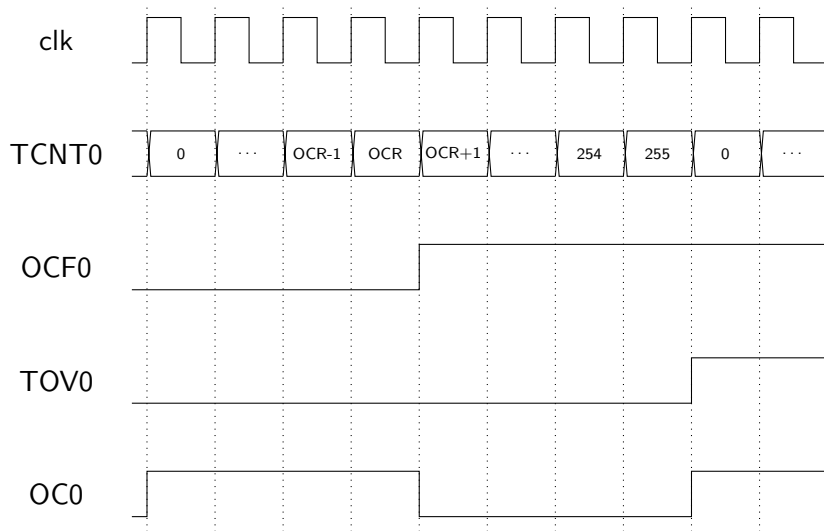
clk – częstotliwość zegara, N – współczynnik podziału preskalera.

- ▶ Współczynnik wypełnienia:

$$\frac{\text{OCR0} + 1}{256}.$$

- ▶ VMLAB **nie** symuluje tego trybu!

Przebieg czasowy w trybie szybkiego PWM



PWM poprawny fazowo

- ▶ Licznik zlicza w górę od 0 do 255, a potem w dół do 0.
- ▶ Stan wyjścia OC0 zmienia się (w sposób określony przez konfigurację) przy osiągnięciu zgodności.
- ▶ Uzyskiwana częstotliwość:

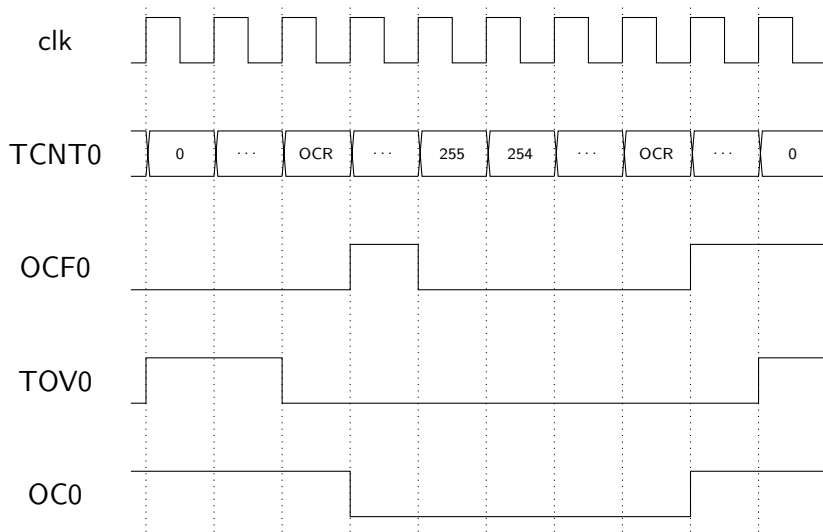
$$\frac{\text{clk}}{510 \cdot N},$$

clk – częstotliwość zegara, N – współczynnik podziału preskalera.

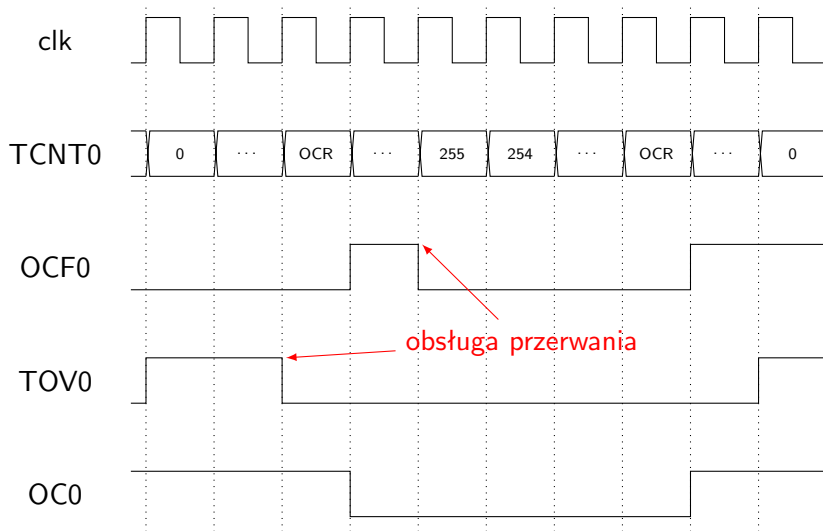
- ▶ Współczynnik wypełnienia:

$$\frac{\text{OCR0}}{255}.$$

Przebieg czasowy — PWM poprawny fazowo



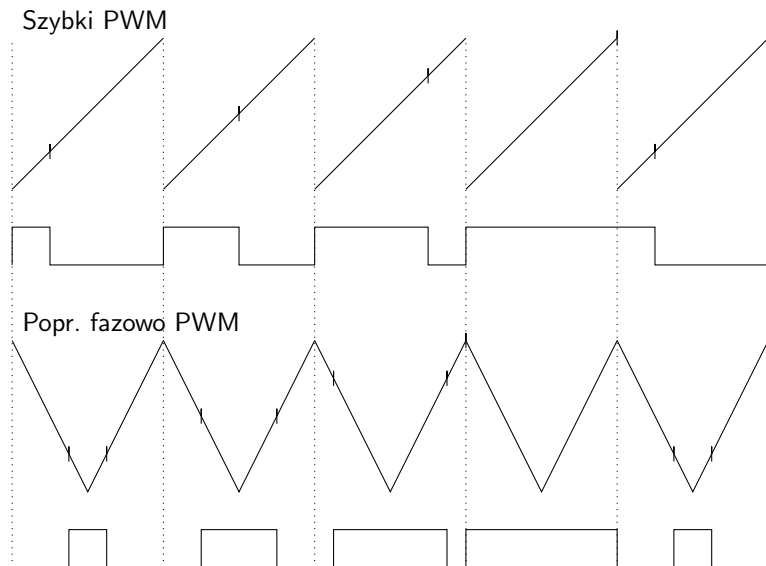
Przebieg czasowy — PWM poprawny fazowo



Buforowanie OCR0

- ▶ Zmiany OCR0 są buforowane (niezależnie od tego, czy licznik pracuje!) i mają wpływ na licznik dopiero gdy:
 - ▶ licznik osiągnie wartość 0, jeśli trybem pracy jest szybki PWM,
 - ▶ licznik osiągnie wartość 255, jeśli trybem pracy jest PWM poprawny fazowo.

Różnica między trybami PWM



Dwa specjalne przypadki zmiany stanu OC0 w PWM poprawnym fazowo

- ▶ Gdy OCR0 zmienia wartość z 255 na inną.
- ▶ Gdy licznik zaczyna zliczać od wartości większej niż OCR0.

Konfiguracja licznika do pracy w trybie szybkiego PWM

7	6	5	4	3	2	1	0
FOC0	WGM00 (PWM0)	COM01	COM00	WGM01 (CTC0)	CS02	CS01	CS00

- ▶ $WGM01 = 1$, $WGM00 = 1$ — szybki PWM,
- ▶ $WGM01 = 0$, $WGM00 = 1$ — PWM poprawny fazowo,
- ▶ FOC0 musi być = 0,
- ▶ Bity preskalera CS00, CS01, CS02 jak poprzednio.

Przebieg na nodze OC0 w trybie szybkiego PWM

COM01	COM00	stan wyjścia OC0
0	0	OC0 nie zmienia się
0	1	zarezerwowane
1	0	OC0 jest zerowany przy zgodności, a ustawiany, gdy licznik osiąga 0
1	1	OC0 jest ustawiany, gdy licznik osiąga 0, a zerowany przy zgodności

- ▶ We wszystkich trybach z wyjątkiem pierwszego OC0 jest przyłączony do PB3.

Przebieg na nodze OC0 w trybie PWM poprawnego fazowo

COM01	COM00	stan wyjścia OC0
0	0	OC0 nie zmienia się
0	1	OC0 zarezerwowane
1	0	OC0 jest zerowany przy $TCNT0=OCR0$, gdy licznik zlicza w górę, a ustawiany, gdy zlicza w dół
1	1	OC0 jest ustawiany przy $TCNT0=OCR0$, gdy licznik zlicza w górę, a zerowany, gdy zlicza w dół

- ▶ We wszystkich trybach z wyjątkiem pierwszego OC0 jest przyłączony do PB3.

Eksperyment

- ▶ Wygenerujemy sygnał o zmienianym współczynniku wypełnienia.
- ▶ Jako odbiornik podłączmy diodę.
- ▶ Zaczniemy od niedużych częstotliwości.

Kod

```
sbi ddrb, 3      ; PB3 = OC0 jako wyjście

; współczynnik wypełnienia ok. 50%
ldi r16, 127
out ocr0, r16

; konfigurujemy licznik 0
; tryb PWM poprawny fazowo
; ustawiamy OC0 przy liczeniu w dół
; zerujemy OC0 przy liczeniu w górę
; preskaler 1024
ldi r16, 1 << wgm00 | 1 << com01 |
          1 << cs00  | 1 << cs02
out tccr0, r16
```


Kod, cd.

```
forever:
    ; naiwne skanowanie klawiszy
    sbis pinc, pc0
    rjmp zwieksz          ; jeśli pc0 = 0, to zwiększ
    sbis pinc, pc1
    rjmp zmniejsz        ; jeśli pc1 = 0, to zmniejsz
    rjmp forever
```

Kod, cd.

zwiększ:

```
in r16, ocr0      ; zwiększamy OCR0 o 10
subi r16, -10
out ocr0, r16
```

petla1:

```
sbis pinc, pc0    ; czekamy na zwolnienie przycisku
rjmp petla1
rjmp forever
```

- Kod zmniejszający wartość OCR0 po wciśnięciu przycisku podłączonego do PC1 jest analogiczny do powyższego.

Wnioski

- ▶ Dioda zgodnie z oczekiwaniami miga, przy czym zmienia się długość fazy świecenia i długość przerw między mignięciami.
- ▶ Ale co się stanie, gdy zwiększymy częstotliwość?

Sterowanie jasnością diody

- ▶ Podajemy na diodę impulsy o dużej częstotliwości.
- ▶ Jasność diody zależy (nieliniowo) od współczynnika wypełnienia.
- ▶ Można próbować uzyskać liniowość jasności, zmieniając nieliniowo współczynnik wypełnienia.

Przykłady innych zastosowań

- ▶ Sterowanie kątem wychylenia serwa modelarskiego:
 - ▶ kąty między -90 stopni a 90 stopni,
 - ▶ kąt 0 stopni przy długości impulsu $1,5$ ms,
 - ▶ dopuszczalny zakres długości impulsu od $0,3$ ms do $2,7$ ms,
 - ▶ częstotliwość 50 Hz.
- ▶ Sterowanie prędkością obrotową silnika:
 - ▶ konieczny układ pośredniczący,
 - ▶ gwarancja właściwego zabezpieczenia,
 - ▶ możliwość regulacji prędkości obrotowej i kierunku obrotów.
- ▶ Konwersja cyfrowo-analogowa:
 - ▶ sygnał PWM podany na filtr dolnoprzepustowy,
 - ▶ napięcie wyjściowe proporcjonalne do współczynnika wypełnienia,
 - ▶ filtr dolnoprzepustowy jest niepotrzebny, jeśli membrana głośnika ma odpowiednio dużą bezwładność,
 - ▶ wychylenie membrany jest proporcjonalne do współczynnika wypełnienia.

Sterowanie serwomechanizmem modelarskim

- ▶ Założenia:
 - ▶ częstotliwość sygnału sterującego wynosi 50 Hz,
 - ▶ szerokość impulsu z zakresu 0,3 ms do 2,7 ms ustala położenie ramienia,
 - ▶ chcemy ustawiać ramię w pozycjach między -90 a 90 stopni z krokiem co 5 stopni.
- ▶ Obliczenia:
 - ▶ liczba pozycji: $180/5 = 36$,
 - ▶ zmiana długości impulsu przypadająca na jedną pozycję: $(2,7 - 0,3)/36 \approx 67 \mu s$,
 - ▶ wartość współczynnika preskalera w trybie PWM poprawnym fazowo: $8 \text{ MHz}/(50 \text{ Hz} \cdot 510) \approx 313,7$,
 - ▶ wartość współczynnika preskalera w trybie szybkiego PWM: $8 \text{ MHz}/(50 \text{ Hz} \cdot 256) = 625$.
- ▶ Problem: nie dysponujemy takimi wartościami współczynników.

Licznik 1

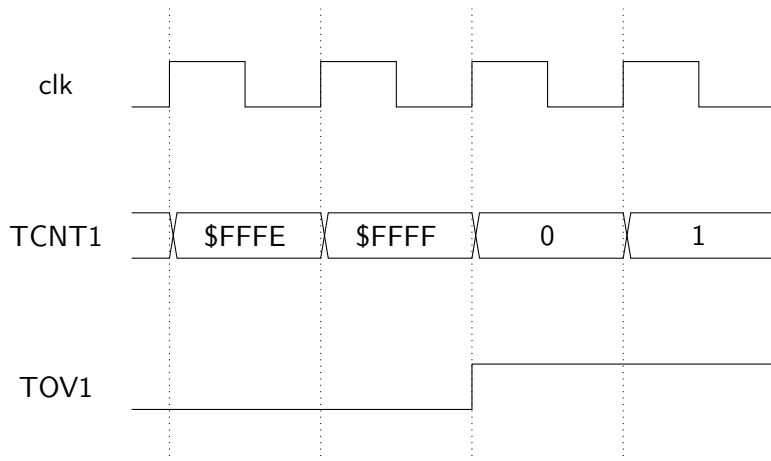
- ▶ Jest 16-bitowy.
- ▶ Ma 15 trybów pracy.
- ▶ Może być porównywany z dwoma rejestrami: OCR1A i OCR1B.
- ▶ Przy zgodności może wyzwać dwa rodzaje przerwań o adresach OC1Aaddr, OC1Baddr.
- ▶ Może sterować stanem wyprowadzeń: OC1A=PD5 i OC1B=PD4.
- ▶ Znaczniki zgodności mają nazwy OCF1A i OCF1B.
- ▶ Znacznik przepełnienia ma nazwę TOV1.

Tryb zwykły

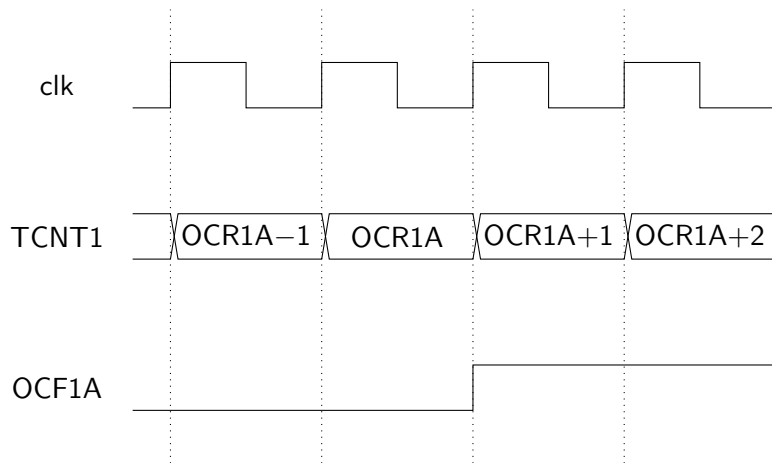
- ▶ Jak tryb zwykły licznika 0.
- ▶ Różnica: licznik zlicza do 0xFFFF.
- ▶ Częstotliwość przebiegu na wyprowadzeniach OC1A i OC1B:

$$\frac{\text{clk}}{N \cdot 65536}.$$

Przebieg czasowy, tryb zwykły bez preskalera



Przebieg czasowy, tryb zwykły bez preskalera



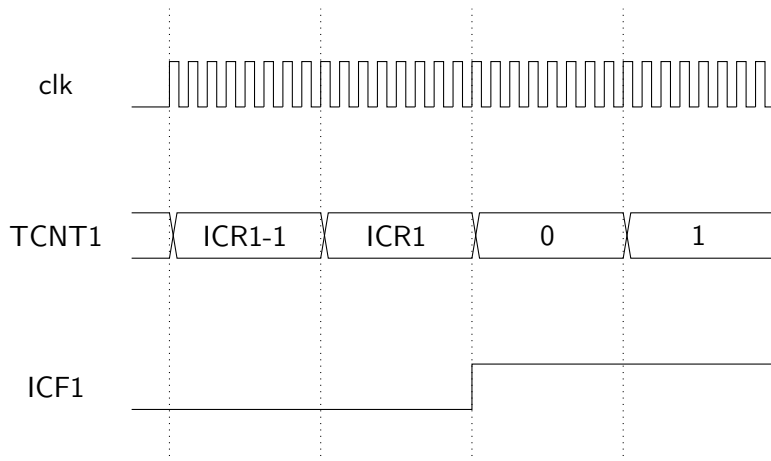
Analogicznie dla OCF1B.

Tryby CTC

- ▶ Jak tryb CTC licznika 0.
- ▶ W zależności od konfiguracji licznik zlicza do:
 - ▶ OCR1A lub
 - ▶ ICR1.
- ▶ Jeśli licznik zlicza do ICR1, to w chwili przejścia między wartością ICR1 a 0 jest ustawiany znacznik ICF1.
- ▶ Ustawienie znacznika ICF1 może wyzwolić przerwanie o adresie ICP1addr.
- ▶ Niezależnie od konfiguracji znaczniki zgodności są ustawiane w standardowy sposób (przy zmianie wartości licznika z OCR1A/B na kolejną).
- ▶ Częstotliwość przebiegów na OC1A/B (przy zliczaniu do ICR1):

$$\frac{\text{clk}}{N \cdot (\text{ICR1} + 1)}.$$

Przebieg czasowy, tryb CTC, preskaler /8



Szybki PWM licznika 1

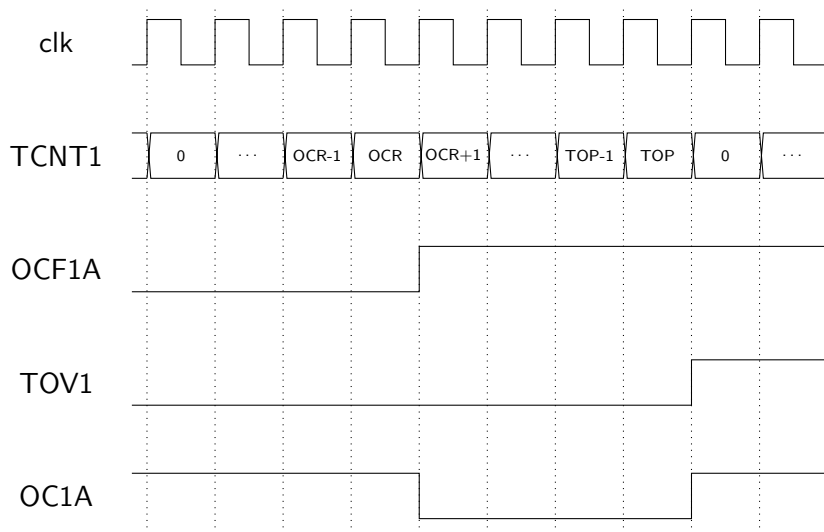
- ▶ Jak szybki PWM licznika 0.
- ▶ Jednocześnie można wygenerować dwa różne przebiegi o tych samych częstotliwościach.
- ▶ Jest pięć trybów szybkiego PWM dla licznika 1, różniących się wartością (TOP), do której zlicza licznik:
 - ▶ PWM 8-bitowy, TOP=255,
 - ▶ PWM 9-bitowy, TOP=511,
 - ▶ PWM 10-bitowy, TOP=1023,
 - ▶ TOP = ICR1A (musi być równe co najmniej 3),
 - ▶ TOP = OCR1A (musi być równe co najmniej 3).
- ▶ Częstotliwość przebiegów na OC1A/B:

$$\frac{\text{clk}}{N \cdot (\text{TOP} + 1)}.$$

- ▶ Współczynnik wypełnienia przebiegu na OC1A:

$$\frac{\text{OCR1A} + 1}{\text{TOP} + 1}.$$

Przebieg czasowy w trybie szybkiego PWM



Poprawny fazowo tryb PWM licznika 1

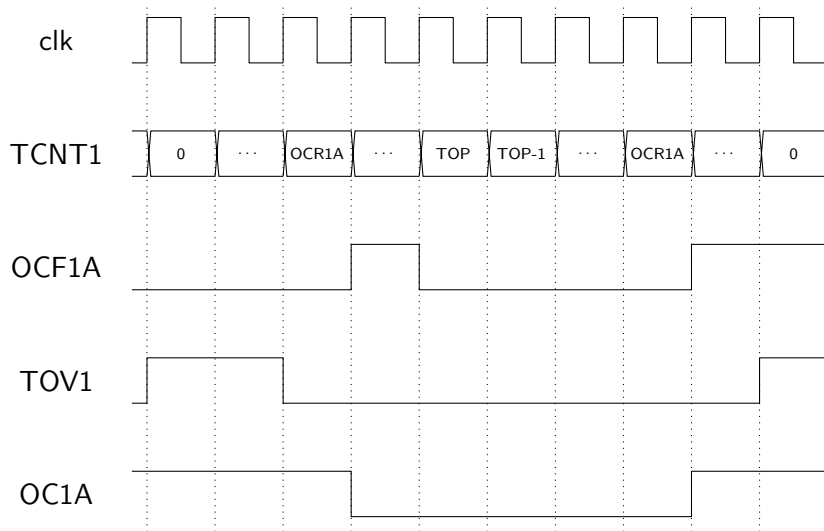
- ▶ Jak poprawny fazowo PWM licznika 0.
- ▶ Jednocześnie można wygenerować dwa różne przebiegi o tych samych częstotliwościach.
- ▶ Jest pięć trybów poprawnego fazowo PWM licznika 1, różniących się wartością (TOP), do której zlicza licznik:
 - ▶ PWM 8-bitowy, TOP=255,
 - ▶ PWM 9-bitowy, TOP=511,
 - ▶ PWM 10-bitowy, TOP=1023,
 - ▶ TOP = ICR1 (musi być równe co najmniej 3),
 - ▶ TOP = OCR1A (musi być równe co najmniej 3).
- ▶ Częstotliwość przebiegów na OC1A/B:

$$\frac{\text{clk}}{2 \cdot N \cdot \text{TOP}}.$$

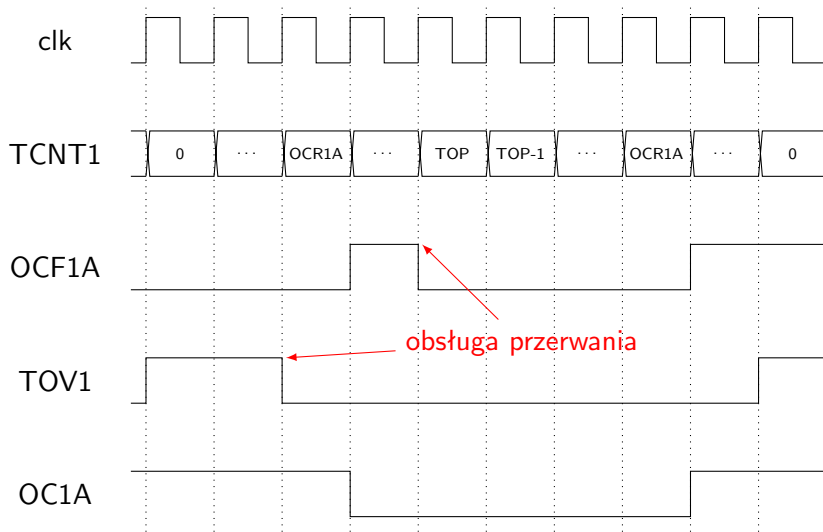
- ▶ Współczynnik wypełnienia przebiegu na OC1A:

$$\frac{\text{OCR1A}}{\text{TOP}}.$$

Przebieg czasowy — PWM poprawny fazowo



Przebieg czasowy — PWM poprawny fazowo



Poprawny fazowo i częstotliwościowo tryb PWM licznika 1

- ▶ Licznik zlicza w górę od 0 do TOP, a potem w dół do 0.
- ▶ Są dwa tryby szybkiego PWM dla licznika 1, różniące się wartością (TOP), do której zlicza licznik:
 - ▶ $TOP = ICR1$ (musi być równe co najmniej 3),
 - ▶ $TOP = OCR1A$ (musi być równe co najmniej 3).
- ▶ Jednocześnie można wygenerować dwa różne przebiegi o tych samych częstotliwościach, wykorzystując OCR1A i OCR1B.
- ▶ Stan wyjścia OC1A/1B zmienia się (w sposób określony przez konfigurację) przy osiągnięciu zgodności.
- ▶ Rejestr OCR1A jest uaktualniany, gdy licznik osiąga wartość 0.

Szesnastobitowe rejestry licznika 1

- ▶ Rejestry TCNT1, OCR1A, OCR1B i ICR1 są 16-bitowe.
- ▶ Zapisuje się je i odczytuje w specjalny sposób:
 - ▶ licznik jest wyposażony w jeden rejestr pomocniczy,
 - ▶ zapis do starszego bajtu: TCNT1H, OCR1AH, OCR1BH lub ICR1H umieszcza wartość w rejestrze pomocniczym,
 - ▶ zapis do młodszej bajtu: TCNT1L, OCR1AL, OCR1BL lub ICR1L wyzwala właściwą 16-bitową operację zapisu,
 - ▶ odczyt starszego bajtu powoduje odczyt rejestru pomocniczego,
 - ▶ odczyt młodszej bajtu powoduje skopiowanie w tym samym cyklu zegara wartości starszego bajtu do rejestru pomocniczego.

Szesnastobitowe rejestry licznika 1, cd.

- ▶ Właściwa kolejność zapisu:

```
out TCNT1H, r17
```

```
out TCNT1L, r16
```

- ▶ Właściwa kolejność odczytu:

```
in r16, TCNT1L
```

```
in r17, TCNT1H
```

- ▶ Rejestr pomocniczy jest wspólny dla wszystkich 16-bitowych rejestrów licznika.
- ▶ Powyższe dwie operacje odczytu i zapisu nie mogą się przepleść z odczytem lub zapisem tego samego lub innego rejestru 16-bitowego — uwaga na przerwania!

Przechwytywanie wartości licznika

- ▶ Zmiana stanu na wyprowadzeniu ICP1 lub zmiana stanu wyjścia komparatora może wyzwolić przechwytywanie wartości licznika.
- ▶ Aktualna wartość licznika jest wówczas kopiowana do rejestru ICR1.
- ▶ Po zakończeniu kopiowania jest ustawiany znacznik ICF1.
- ▶ Ustawienie znacznika ICF1 może wyzwolić przerwanie o adresie ICP1addr.

Rejestr TCCR1A

7	6	5	4
COM1A1	COM1A0	COM1B1	COM1B0

- ▶ Bity 7, 6 – sposób sterowania wyjściem OC1A
- ▶ Bity 5, 4 – sposób sterowania wyjściem OC1B

3	2	1	0
FOC1A	FOC1B	WGM11	WGM10

- ▶ Bity 3, 2 – wymuszenie zgodności (jak w liczniku 0)
- ▶ Bity 1, 0 – w połączeniu w bitami z TCCR1B ustalają tryb pracy licznika

Rejestr TCCR1B

7	6	5	4	3	2	1	0
ICNC1	ICES1	—	WGM13	WGM12	CS12	CS11	CS10

- ▶ Bit 7 – włącza filtr na ICP1
- ▶ Bit 6 – sposób wyzwalania przechwytywania
- ▶ Bity 4, 3 – starsze bity trybu pracy
- ▶ Bity 2, 1, 0 – preskaler

Rejestr TIMSK ponownie

7	6	5	4	3	2	1	0
		TICIE1	OCIE1A	OCIE1B	TOIE1		

- ▶ Bit 5 – aktywacja przerwania po przechwyceniu wartości licznika tzn., gdy ustawiony znacznik ICF1:
 - ▶ 1 – przerwanie włączone (jeśli ustawiony znacznik I w SREG),
 - ▶ 0 – przerwanie wyłączone.
- ▶ Bit 4 (3) – aktywacja przerwania, gdy porównanie rejestrów TCNT1 i OCR1A (OCR1B) wypadło pozytywnie, tzn. gdy ustawiony znacznik OCF1A (OCF1B):
 - ▶ 1 – przerwanie włączone (jeśli ustawiony znacznik I w SREG),
 - ▶ 0 – przerwanie wyłączone.
- ▶ Bit 2 – aktywacja przerwania, gdy wystąpiło przepełnienie licznika, tzn. gdy ustawiony znacznik TOV1:
 - ▶ 1 – przerwanie włączone (jeśli ustawiony znacznik I w SREG),
 - ▶ 0 – przerwanie wyłączone.
- ▶ Pozostałe bity nie dotyczą licznika 1.

Rejestr TIFR ponownie

7	6	5	4	3	2	1	0
		ICF1	OCF1A	OCF1B	TOV1		

- ▶ Bit 5 – znacznik ustawiany, gdy nastąpiło przechwycenie wartości licznika.
- ▶ Bity 4, 3 – znaczniki ustawiane, gdy porównanie rejestrów TCNT1 i OCR1A/OCR1B wypadło pozytywnie.
- ▶ Bit 2 – znacznik ustawiany, gdy wystąpiło przepełnienie licznika.
- ▶ Jeśli odpowiednie przerwanie jest włączone, to znaczniki są automatycznie zerowane w trakcie obsługi przerwania.
- ▶ Znaczniki mogą być zerowane programowo, przez zapisanie do nich wartości 1.
- ▶ Pozostałe bity nie dotyczą licznika 1.

Tryby pracy

WGM13	WGM12	WGM11	WGM10	Tryb
0	0	0	0	zwykły
0	0	0	1	popr. faz. PWM, 8-bitów
0	0	1	0	popr. faz. PWM, 9-bitów
0	0	1	1	popr. faz. PWM, 10-bitów
0	1	0	0	CTC, TOP=OCR1A
0	1	0	1	szybki PWM, 8-bitów
0	1	1	0	szybki PWM, 9-bitów
0	1	1	1	szybki PWM, 10-bitów
1	0	0	0	popr. faz. i częst. PWM, ICR1
1	0	0	1	popr. faz. i częst. PWM, OCR1A
1	0	1	0	popr. faz. PWM, TOP=ICR1
1	0	1	1	popr. faz. PWM, TOP=OCR1A
1	1	0	0	CTC, TOP=ICR1
1	1	1	0	szybki PWM, TOP=ICR1
1	1	1	1	szybki PWM, TOP=OCR1A

Preskaler licznika 1

CS12	CS11	CS10	źródło zegara
0	0	0	licznik zatrzymany
0	0	1	clk
0	1	0	clk/8
0	1	1	clk/64
1	0	0	clk/256
1	0	1	clk/1024
1	1	0	zewnętrzny z nogi T0, zbocze opadające
1	1	1	zewnętrzny z nogi T0, zbocze narastające

- ▶ Tak naprawdę licznik 0 i licznik 1 współdzielą ten sam preskaler.

Przebieg na nogach OC1A i OC1B — tryb zwykły i CTC

COM1A1 COM1B1	COM1A0 COM1B0	stan wyjścia OC1A stan wyjścia OC1B
0	0	OC0 nie zmienia się
0	1	OC0 zmienia swój stan po wykryciu zgodności
1	0	OC0 jest zerowany po wykryciu zgodności
1	1	OC0 jest ustawiany po wykryciu zgodności

Przebieg na nogach OC1A i OC1B — szybki PWM

COM1A1 COM1B1	COM1A0 COM1B0	stan wyjścia OC1A stan wyjścia OC1B
0	0	nie zmienia się
0	1	tylko w szybkim PWM z TOP = OCR1A OC1A zmienia swój stan po wykryciu zgodności OC1B nie zmienia się
1	0	jest zerowany po wykryciu zgodności i ustawiany w zerze
1	1	jest ustawiany po wykryciu zgodności i zerowany w zerze

Przebieg na nogach OC1A i OC1B — pozostałe tryby PWM

COM1A1 COM1B1	COM1A0 COM1B0	stan wyjścia OC1A stan wyjścia OC1B
0	0	nie zmienia się
0	1	tylko w trybach 9 i 14: OC1A zmienia swój stan po wykryciu zgodności OC1B nie zmienia się
1	0	jest zerowany przy zgodności, gdy liczy w górę a ustawiany przy zgodności, gdy liczy w dół
1	1	jest ustawiany przy zgodności, gdy liczy w górę a zerowany przy zgodności, gdy liczy w dół

Sterowanie serwomechanizmem, cd.

- ▶ Chcemy wygenerować przebieg o częstotliwości 50 Hz i rozdzielczości co najmniej $67 \mu s$.
- ▶ Użyjemy poprawnego fazowo trybu PWM.
- ▶ Częstotliwość

$$f = \frac{clk}{2 \cdot N \cdot TOP}.$$

- ▶ Po podstawieniu wartości i przekształceniu:

$$N \cdot TOP = \frac{8 \text{ MHz}}{100 \text{ Hz}} = 8 \cdot 10^4.$$

- ▶ Można zatem przyjąć, np.: $TOP = 10^4$ i $N = 8$.
- ▶ Rozdzielczość wynosi wówczas $20 \text{ ms} \cdot 10^{-4} = 2 \mu s$.
- ▶ Zmiana szerokości impulsu o $67 \mu s$ odpowiada zmianie wartości OCR1B o ok. 33.

Sterowanie serwomechanizmem, cd.

```
.cseg
.org 0

; użyjemy nogi OC1B, czyli PD4
; konfiguracja jako wyjście
sbi ddrd, 4

; TOP = OCR1A
; Zgodnie z obliczeniami 10000
ldi r17, high (10000)
ldi r16, low (10000)
out OCR1AH, r17
out OCR1AL, r16
```


Sterowanie serwomechanizmem, cd.

```
; Początkowo szerokość = 1.5ms  
; czyli OCR1B = 750  
; rejestry r19:r18 cały czas  
; przechowują aktualną wartość OCR1B  
ldi r19, high (750)  
ldi r18, low (750)  
out OCR1BH, r19  
out OCR1BL, r18
```

Sterowanie serwomechanizmem, cd.

```
; tryb pracy licznika: popr. faz. PWM do OCR1A
; COM1B1 podłącza OC1B do PD4
ldi r16, 1 << WGM10 | 1 << WGM11 | 1 << COM1B1
out TCCR1A, r16
```

```
; dalszy ciąg konfiguracji i start
; licznika z preskalerem 8
ldi r16, 1 << WGM13 | 1 << CS11
out TCCR1B, r16
```

```
forever: ; naiwne skanowanie klawiszy
sbis pina, 0
rjmp zwieksz
sbis pina, 1
rjmp zmniejsz
rjmp forever
```

Sterowanie serwomechanizmem, cd.

zwiększ:

```
subi r18, low (-33)
sbci r19, high (-33)
out OCR1BH, r19
out OCR1BL, r18
```

czekaj: ; czekamy na puszczenie klawisza

```
sbis pina, 0
rjmp czekaj
rjmp forever
```

Zmniejszanie szerokości impulsu wygląda podobnie.

Licznik 2

- ▶ Jest także 8-bitowy.
- ▶ Ma te same tryby pracy, co licznik 0.
- ▶ Dodatkowo może pracować asynchronicznie w stosunku do zegara systemowego.
- ▶ Ma osobny preskaler z innymi współczynnikami.

Praca w trybie asynchronicznym

- ▶ Domyślnie licznik 2 jest taktowany (przeskalowanym) sygnałem zegara systemowego.
- ▶ Można to jednak zmienić, ustawiając bit AS2 w rejestrze ASSR.
- ▶ Powoduje to:
 - ▶ odłączenie wyprowadzeń PC6 (TOSC1) i PC7 (TOSC2) od portu C,
 - ▶ możliwość podłączenia między nie kwarcu,
 - ▶ optymalizację do pracy z oscylatorem zegarkowym (32768 Hz).
- ▶ Przy przełączaniu trybu pracy należy wyłączyć przerwania licznika 2 i zainicjować rejestry TCNT2, TCCR2 i OCR2.

Rejestry licznika 2

- ▶ Wartość licznika: TCNT2
- ▶ Wartość porównywana z licznikiem: OCR2
- ▶ Rejestr kontrolny: TCCR2

Rejestr TCCR2

7	6	5	4	3	2	1	0
FOC2	WGM20 (PWM2)	COM21	COM20	WGM21 (CTC2)	CS22	CS21	CS20

- ▶ Bity 6, 3 – ustalają tryb pracy, jak w przypadku licznika 0,
- ▶ Bity 5, 4 – sposób sterowania wyjściem OC2 (PD7), jak w przypadku licznika 0,
- ▶ Bity 2, 1, 0 – konfiguracja preskalera,
- ▶ Bit 7 – jak w przypadku licznika 0.

Preskaler licznika 2

CS22	CS21	CS20	źródło zegara
0	0	0	licznik zatrzymany
0	0	1	clk2
0	1	0	clk2/8
0	1	1	clk2/32
1	0	0	clk2/64
1	0	1	clk2/128
1	1	0	clk2/256
1	1	1	clk2/1024

Rejestr TIMSK ponownie

7	6	5	4	3	2	1	0
OCIE2	TOIE2					OCIE0	TOIE0

- ▶ Bit 7 – aktywacja przerwania, gdy porównanie rejestrów TCNT2 i OCR2 wypadło pozytywnie, tzn. gdy ustawiony znacznik OCF2:
 - ▶ 1 – przerwanie włączone (jeśli ustawiony znacznik I w SREG),
 - ▶ 0 – przerwanie wyłączone.
- ▶ Bit 6 – aktywacja przerwania, gdy wystąpiło przepełnienie licznika, tzn. gdy ustawiony znacznik TOV2:
 - ▶ 1 – przerwanie włączone (jeśli ustawiony znacznik I w SREG),
 - ▶ 0 – przerwanie wyłączone.
- ▶ Pozostałe bity dotyczą licznika 1.

Rejestr TIFR ponownie

7	6	5	4	3	2	1	0
OCF2	TOV2					OCF0	TOV0

- ▶ Bit 7 – znacznik ustawiany, gdy porównanie rejestrów TCNT2 i OCR2 wypadło pozytywnie.
- ▶ Bit 6 – znacznik ustawiany, gdy wystąpiło przepełnienie licznika.
- ▶ Jeśli odpowiednie przerwanie jest włączone, to znacznik jest automatycznie zerowany w trakcie obsługi przerwania.
- ▶ Znaczniki mogą być zerowane programowo, przez zapisanie do nich wartości 1.
- ▶ Pozostałe bity dotyczą licznika 1.

Rejestr ASSR

7	6	5	4	3	2	1	0
—	—	—	—	AS2	TCN2UB	OCR2UB	TCR2UB

- ▶ Bit 3 - uaktywnia tryb asynchroniczny,
- ▶ Bit 2 - ustawiany sprzętowo w chwili zapisu do TCNT2, zerowany sprzętowo, gdy TCNT2 zostanie uaktualniony,
- ▶ Bit 1 - ustawiany sprzętowo w chwili zapisu do OCR2, zerowany sprzętowo, gdy OCR2 zostanie uaktualniony,
- ▶ Bit 0 - ustawiany sprzętowo w chwili zapisu do TCCR2, zerowany sprzętowo, gdy TCCR2 zostanie uaktualniony.

Bity 0, 1, 2 mają znaczenie tylko, jeśli licznik pracuje w trybie asynchronicznym.

Zapis i odczyt rejestrów TCNT2, OCR2 i TCCR2 w trybie asynchronicznym

- ▶ Zapis odbywa się do rejestrów roboczych.
- ▶ Sprzęt ustawia odpowiedni bit zajętości.
- ▶ Bit ten jest zerowany, gdy wartość z rejestru roboczego zostanie wpisana do właściwego rejestru.
- ▶ Odczyt TCNT2 powoduje odczytanie faktycznej wartości rejestru.
- ▶ Odczyt pozostałych rejestrów powoduje odczyt rejestru roboczego.