# EIE3105: Wave Generating and Capturing (Chapter 16 and 17)
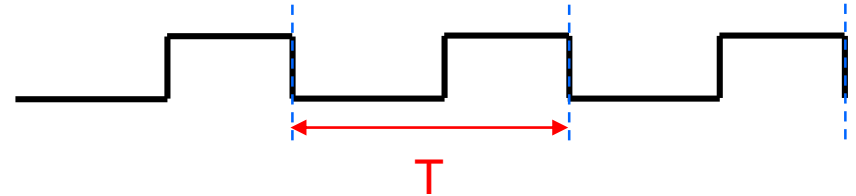
Dr. Lawrence Cheung

Semester 1, 2021/22

# Topics

- Wave characteristics
- Timer 0 review
- Wave generating using Timer 0
- Wave generating using Timer 2
- Wave generating using Timer 1
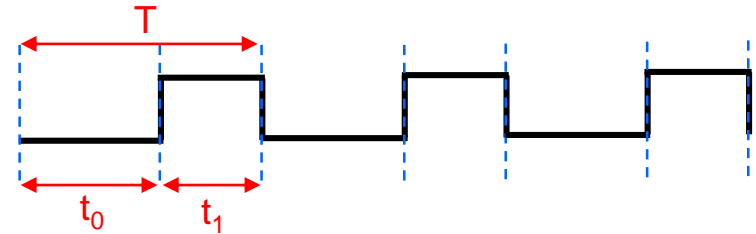- Capturing

# Wave characteristics

- Period
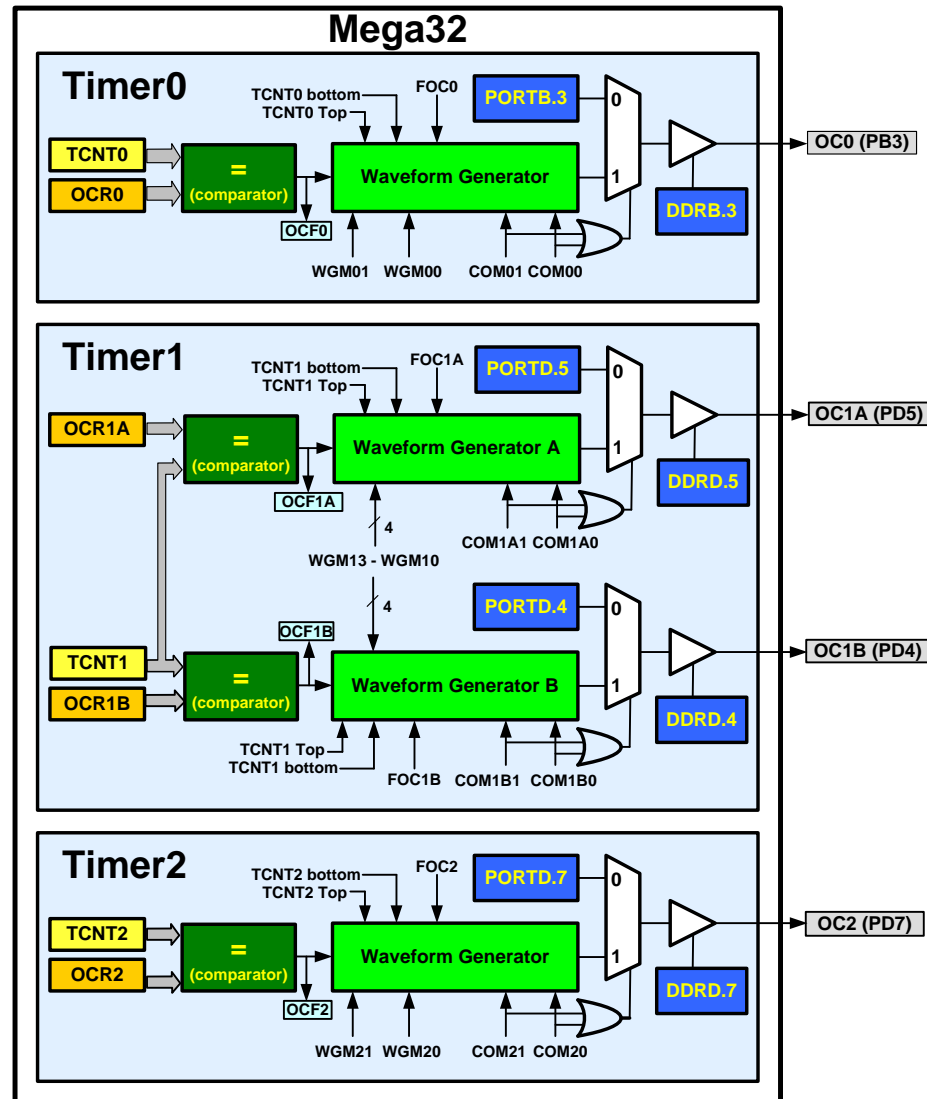  - Frequency

$$f = \frac{1}{T}$$

- Duty cycle

$$\text{duty cycle} = \frac{t_1}{T} \times 100 = \frac{t_1}{t_0 + t_1} \times 100$$
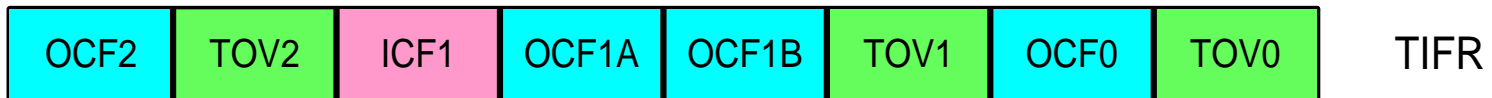
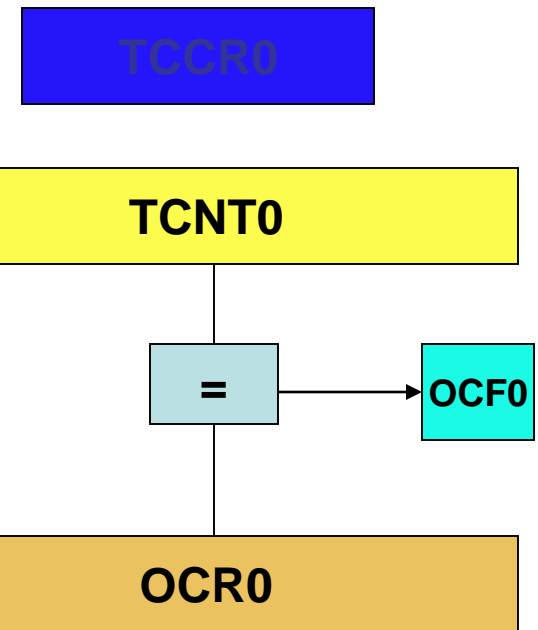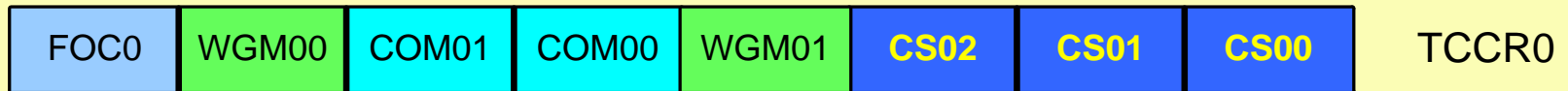- Amplitude

# Waveform generators in ATmega32

4

# Timer 0 Review

TCCR0

TCNT0

TOV0

= → OCF0

OCR0

| OCF2 | TOV2 | ICF1 | OCF1A | OCF1B | TOV1 | OCF0 | TOV0 | TIFR |
|------|------|------|-------|-------|------|------|------|------|

| FOC0 | WGM00 | COM01 | COM00 | WGM01 | **CS02** | **CS01** | **CS00** | TCCR0 |

**Clock Selector (CS)**

| CS02 | CS01 | CS00 | Comment |
|------|------|------|---------|
| 0 | 0 | 0 | No clock source (Timer/Counter stopped) |
| 0 | 0 | 1 | clk (No Prescaling) |
| 0 | 1 | 0 | clk / 8 |
| 0 | 1 | 1 | clk / 64 |
| 1 | 0 | 0 | clk / 256 |
| 1 | 0 | 1 | clk / 1024 |
| 1 | 1 | 0 | External clock source on T0 pin. Clock on falling edge |
| 1 | 1 | 1 | External clock source on T0 pin. Clock on rising edge |

| FOC0 | WGM00 | COM01 | COM00 | WGM01 | CS02 | CS01 | CS00 | TCCR0 |
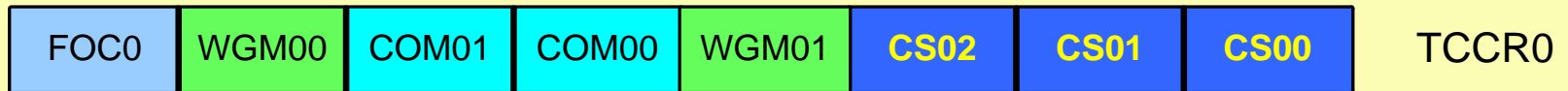
**Timer Mode (WGM)**

| WGM00 | WGM01 | Comment |
|-------|-------|---------|
| 0 | 0 | Normal |
| 0 | 1 | CTC (Clear Timer on Compare Match) |
| 1 | 0 | PWM, phase correct |
| 1 | 1 | Fast PWM |

# Normal mode

TCNT0

0xFF

0

**TOV**  **TOV**  **TOV**

time

**TOV0:** 1

FF

FE

2

1

0

**TOV0 = 1**

# CTC mode

TCNT0

0xFF

OCR0

OCF0    OCF0    OCF0

0                      time

OCR0

xx

2

1

0

**TOV0 = no change**
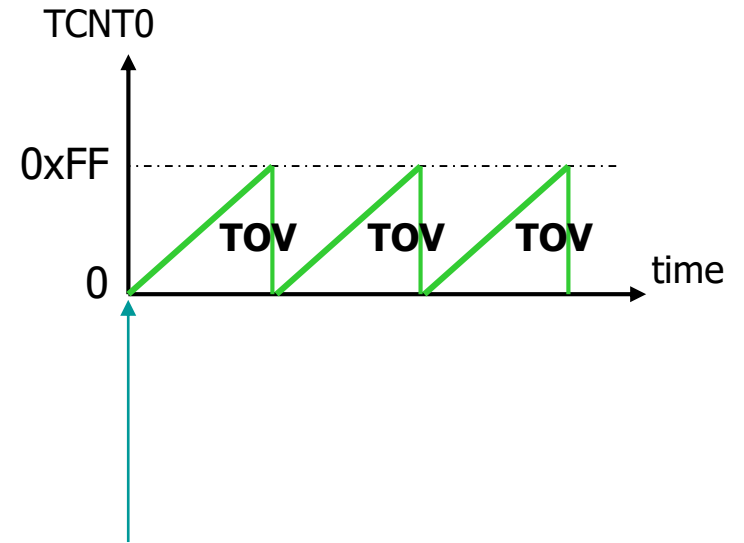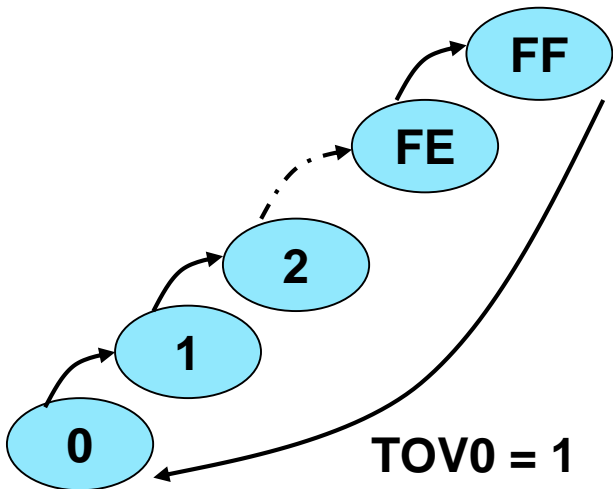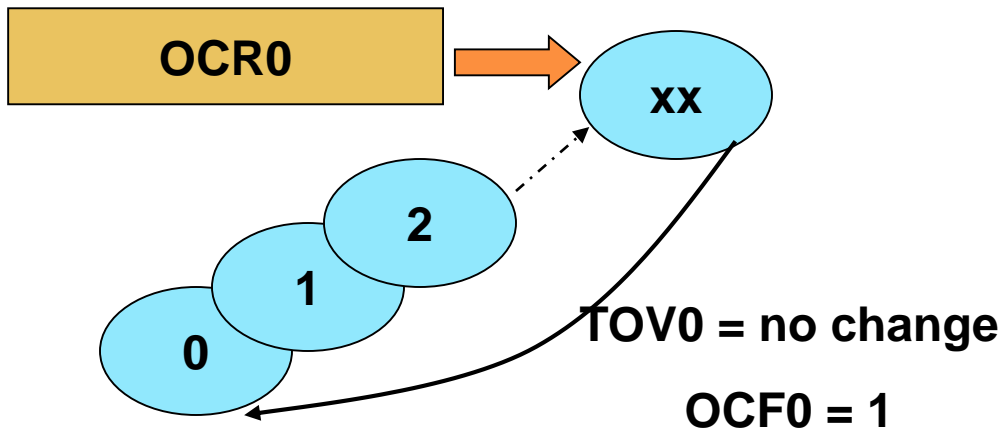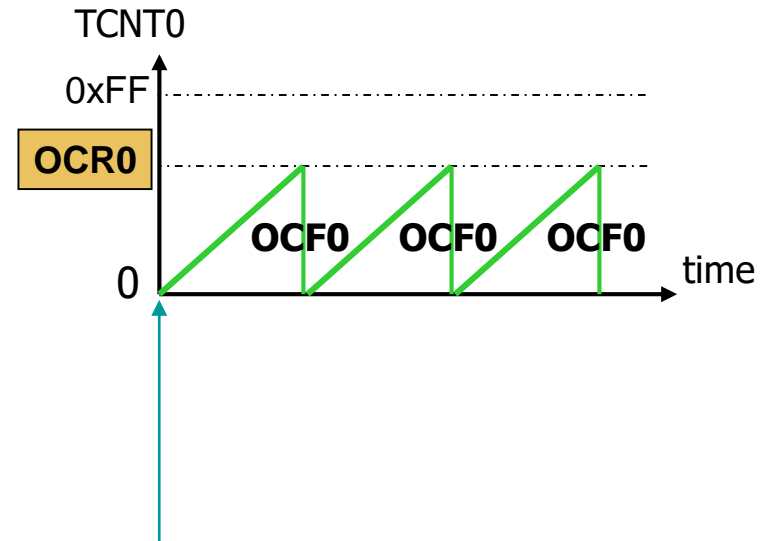
**OCF0 = 1**

**TOV0:**    0

**OCF0:**    1

# Waveform Generator

# Waveform Generator

| FOC0 | WGM00 | COM01 | COM00 | WGM01 | CS02 | CS01 | CS00 | TCCR0 |
|------|-------|-------|-------|-------|------|------|------|-------|
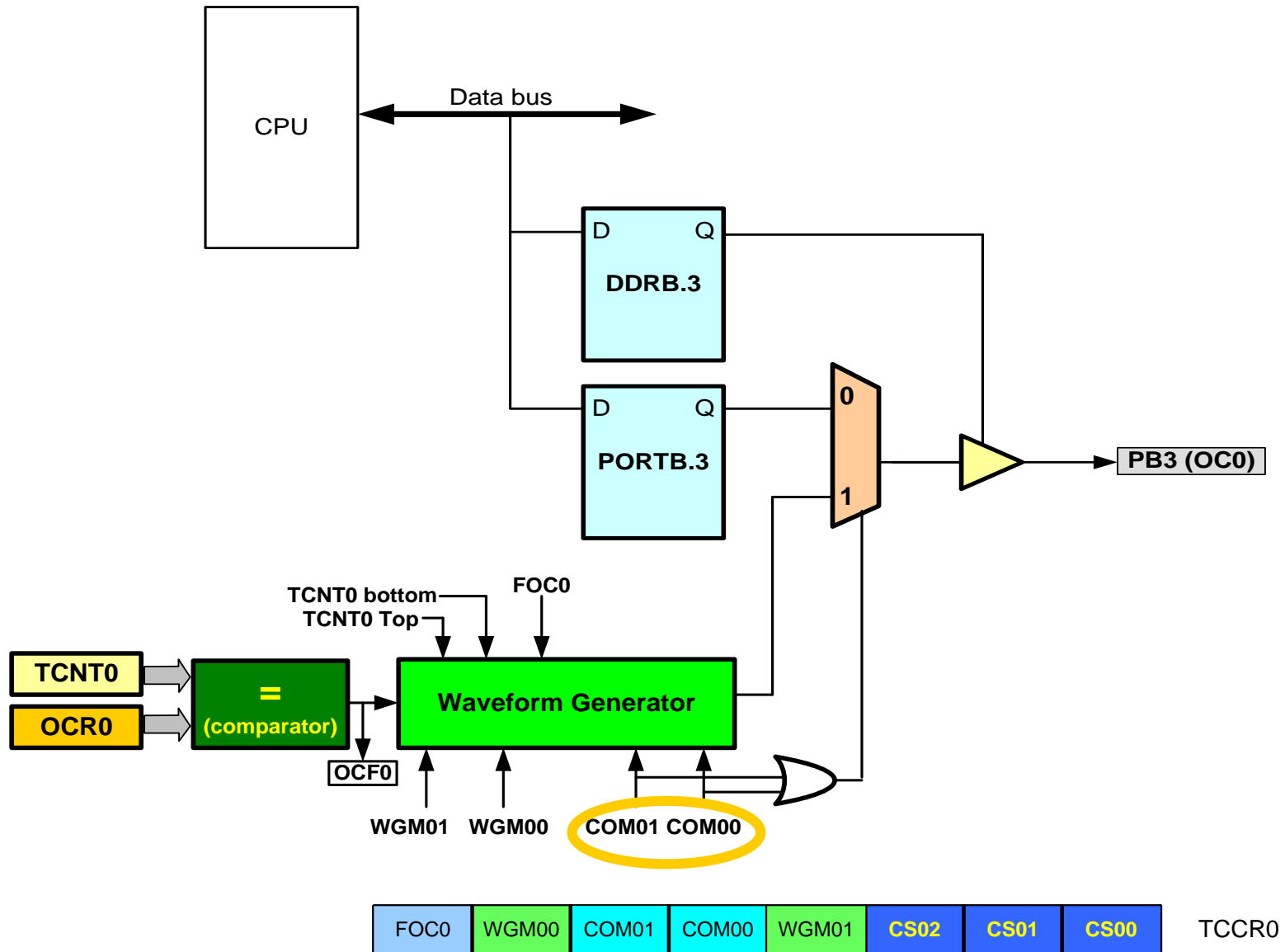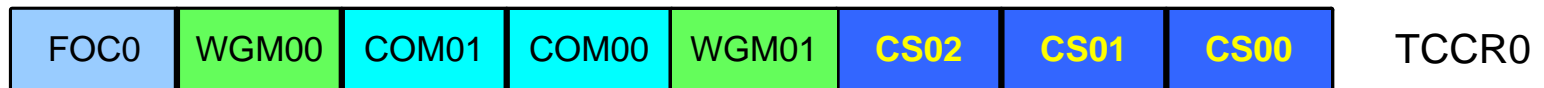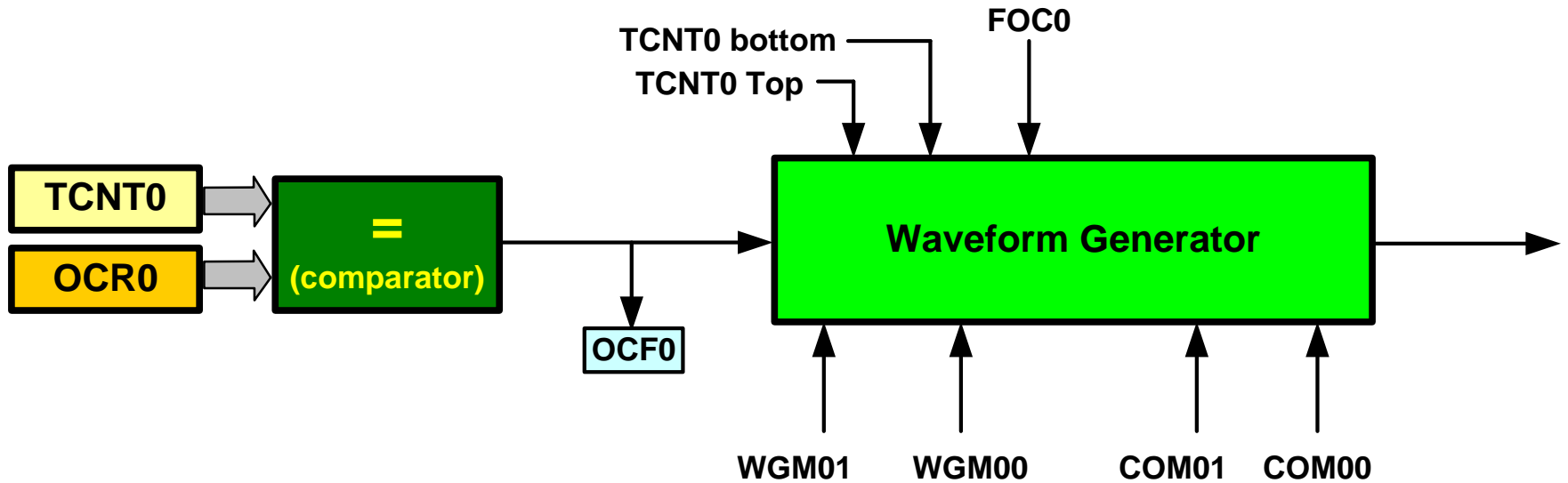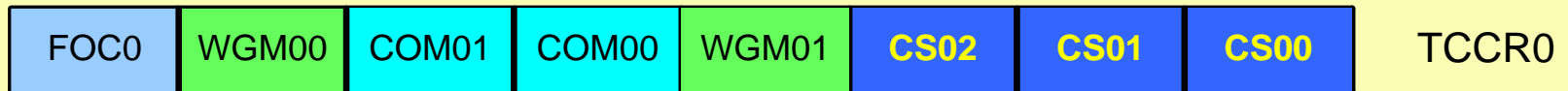
## Compare Output Mode (COM)

| WGM00 | WGM01 | Mode |
|-------|-------|------|
| 0 | 0 | Normal |
| 0 | 1 | CTC (Clear Timer on Compare Match) |
| 1 | 0 | PWM, phase correct |
| 1 | 1 | Fast PWM |

| COM01 | COM00 | Description |
|-------|-------|-------------|
| 0 | 0 | Normal port operation, OC0 disconnected. |
| 0 | 1 | Toggle OC0 on compare match |
| 1 | 0 | Clear OC0 on compare match |
| 1 | 1 | Set OC0 on compare match |

**CTC, Toggle Mode
Duty cycle = 50%
Frequency = changeable**

$$F_{OC0} = \frac{f_{clk}}{2N(OCR0+1)}$$

TCNT0

0xFF

OCR0

OCF0  OCF0  OCF0

0

OC0

TCNT0

0xFF

OCR0

OCF0  OCF0

0

OC0
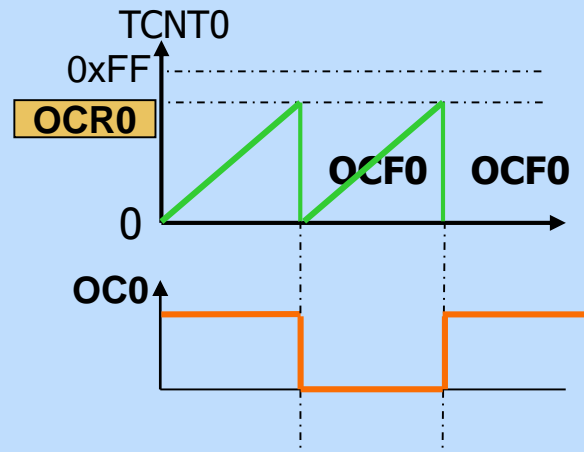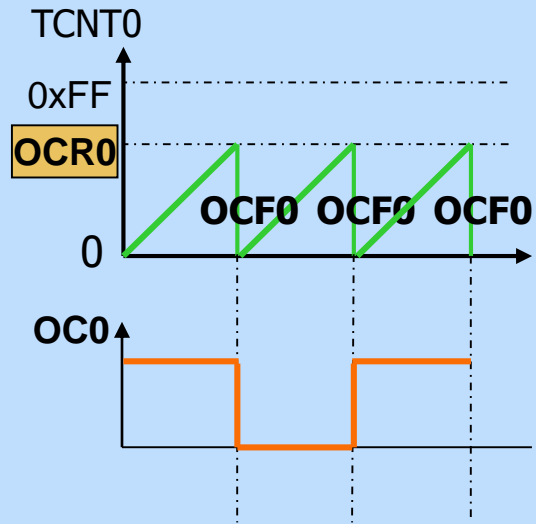
# Waveform Generator

$$F_{OC0} = \frac{f_{clk}}{2N(OCR0+1)}$$ ➡ $$500\text{KHz} = \frac{8\text{MHz}}{2N(OCR0+1)}$$ ➡ $$N(OCR0+1) = \frac{8\text{MHz}}{1\text{MHz}}$$

➡ $N(OCR0+1) = 8$ ➡ $\begin{cases} N = 1 \text{ and } OCR0 = 7 \\ N = 8 \text{ and } OCR0 = 0 \end{cases}$

Assuming XTAL = 8 MHz, make a pulse with duty cycle = 50% and frequency = 500KHz

```
LDI R20,7
OUT OCR0,R20
LDI R20,0x19
OUT TCCR0,R20
```

```
OCR0 = 7;
TCCR0 = 0x19; //prescaler = 1
```

```
LDI R20,0
OUT OCR0,R20
LDI R20,0x1A
OUT TCCR0,R20
```

```
OCR0 = 0;
TCCR0 = 0x1A; //prescaler = 8
```

# Wave generating in Timer 2

- Like Timer0
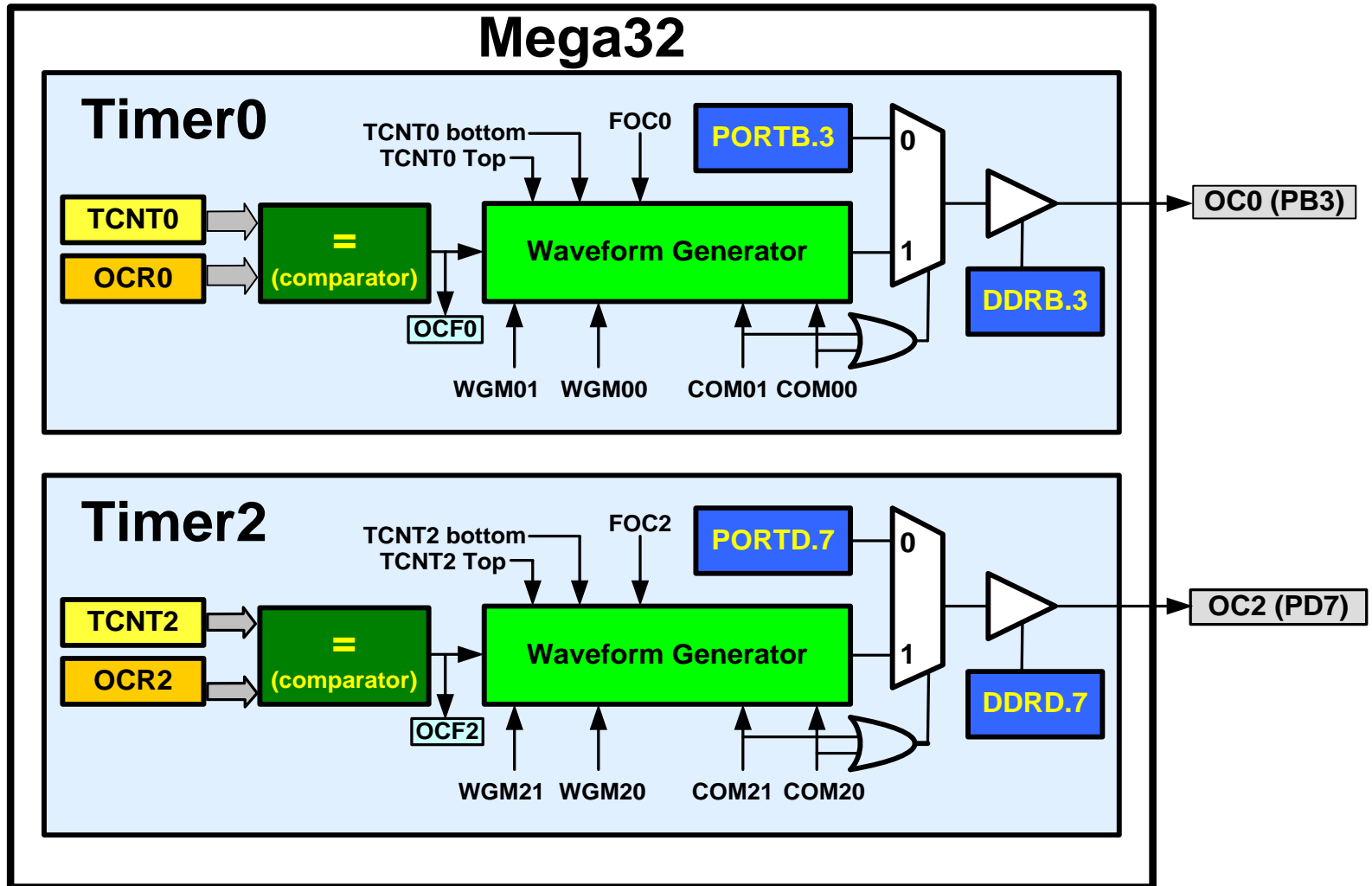
# The difference between Timer 0 and Timer 2

- Timer 0

- Timer 2
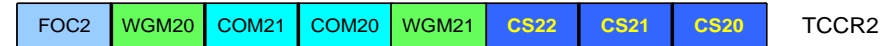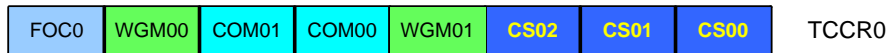
| FOC0 | WGM00 | COM01 | COM00 | WGM01 | CS02 | CS01 | CS00 | TCCR0 |
|------|-------|-------|-------|-------|------|------|------|-------|

| FOC2 | WGM20 | COM21 | COM20 | WGM21 | CS22 | CS21 | CS20 | TCCR2 |
|------|-------|-------|-------|-------|------|------|------|-------|

| CS02 | CS01 | CS00 | Comment |
|------|------|------|---------|
| 0 | 0 | 0 | Timer/Counter stopped |
| 0 | 0 | 1 | clk (No Prescaling) |
| 0 | 1 | 0 | clk / 8 |
| 0 | 1 | 1 | clk / 64 |
| 1 | 0 | 0 | clk / 256 |
| 1 | 0 | 1 | clk / 1024 |
| 1 | 1 | 0 | External clock (falling edge) |
| 1 | 1 | 1 | External clock (rising edge) |

| CS22 | CS21 | CS20 | Comment |
|------|------|------|---------|
| 0 | 0 | 0 | Timer/Counter stopped |
| 0 | 0 | 1 | clk (No Prescaling) |
| 0 | 1 | 0 | clk / 8 |
| 0 | 1 | 1 | clk / 32 |
| 1 | 0 | 0 | clk / 64 |
| 1 | 0 | 1 | clk / 128 |
| 1 | 1 | 0 | clk / 256 |
| 1 | 1 | 1 | clk / 1024 |

# Timer 1

- Timer 1 has two waveform generators.
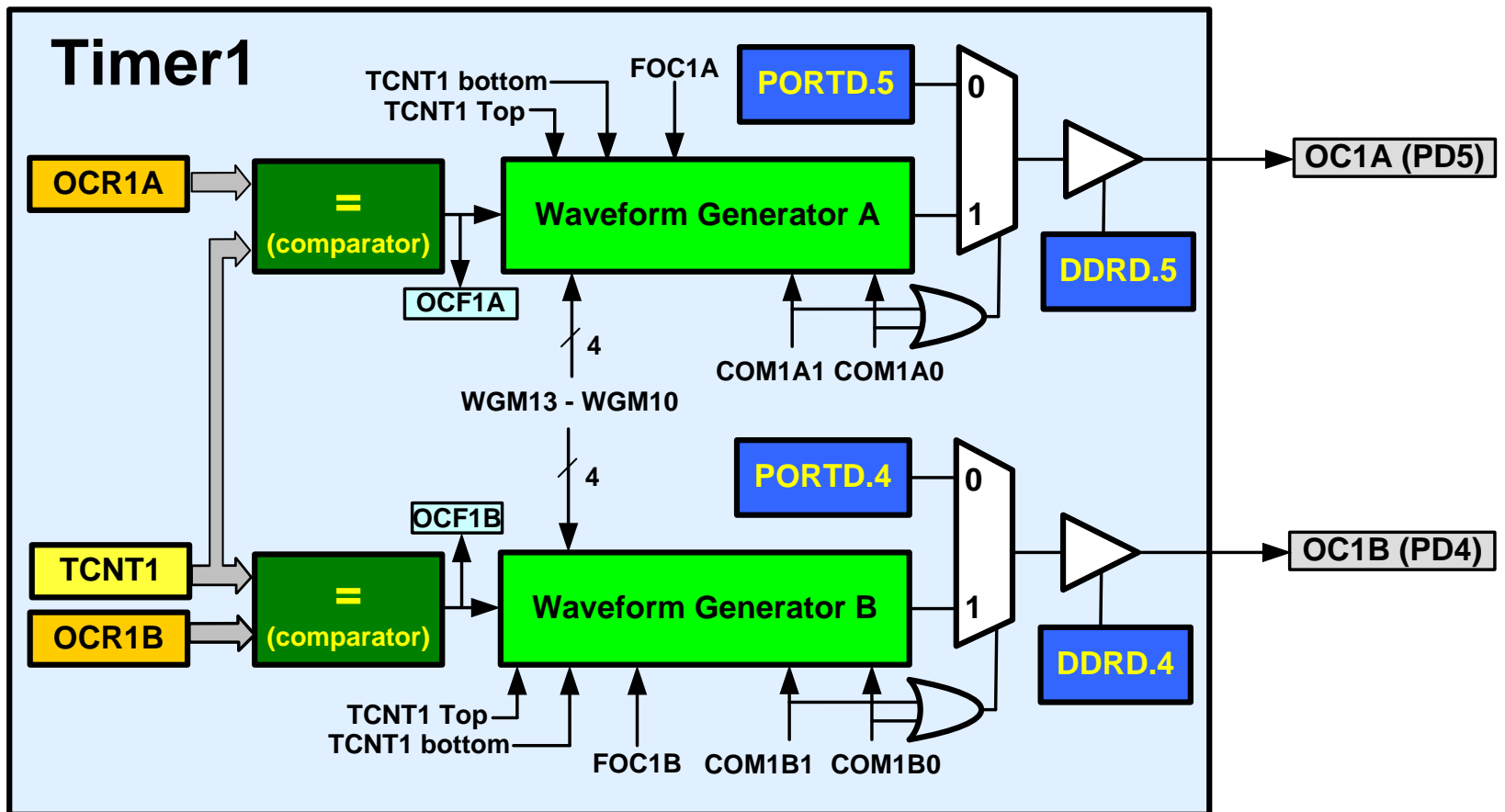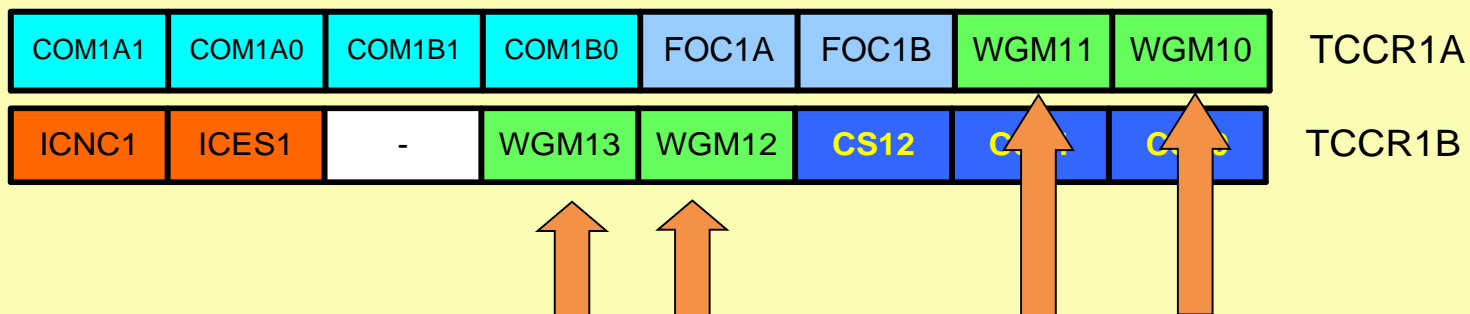
| COM1A1 | COM1A0 | COM1B1 | COM1B0 | FOC1A | FOC1B | WGM11 | WGM10 | TCCR1A |

| ICNC1 | ICES1 | - | WGM13 | WGM12 | CS12 | C | C | TCCR1B |

| Mode | WGM13 | WGM12 (CTC1) | WGM11 (PWM11) | WGM10 (PWM10) | Timer/Counter Mode of Operation | TOP | Update of OCR1x | TOV1 Flag Set on |
|------|-------|--------------|---------------|---------------|----------------------------------|-----|-----------------|------------------|
| 0 | 0 | 0 | 0 | 0 | Normal | 0xFFFF | Immediate | MAX |
| 1 | 0 | 0 | 0 | 1 | PWM, Phase Correct, 8-bit | 0x00FF | TOP | BOTTOM |
| 2 | 0 | 0 | 1 | 0 | PWM, Phase Correct, 9-bit | 0x01FF | TOP | BOTTOM |
| 3 | 0 | 0 | 1 | 1 | PWM, Phase Correct, 10-bit | 0x03FF | TOP | BOTTOM |
| 4 | 0 | 1 | 0 | 0 | CTC | OCR1A | Immediate | MAX |
| 5 | 0 | 1 | 0 | 1 | Fast PWM, 8-bit | 0x00FF | TOP | TOP |
| 6 | 0 | 1 | 1 | 0 | Fast PWM, 9-bit | 0x01FF | TOP | TOP |
| 7 | 0 | 1 | 1 | 1 | Fast PWM, 10-bit | 0x03FF | TOP | TOP |
| 8 | 1 | 0 | 0 | 0 | PWM, Phase and Frequency Correct | ICR1 | BOTTOM | BOTTOM |
| 9 | 1 | 0 | 0 | 1 | PWM, Phase and Frequency Correct | OCR1A | BOTTOM | BOTTOM |
| 10 | 1 | 0 | 1 | 0 | PWM, Phase Correct | ICR1 | TOP | BOTTOM |
| 11 | 1 | 0 | 1 | 1 | PWM, Phase Correct | OCR1A | TOP | BOTTOM |
| 12 | 1 | 1 | 0 | 0 | CTC | ICR1 | Immediate | MAX |
| 13 | 1 | 1 | 0 | 1 | Reserved | – | – | – |
| 14 | 1 | 1 | 1 | 0 | Fast PWM | ICR1 | TOP | TOP |
| 15 | 1 | 1 | 1 | 1 | Fast PWM | OCR1A | TOP | TOP |

| COM1A1 | COM1A0 | COM1B1 | COM1B0 | FOC1A | FOC1B | WGM11 | WGM10 | TCCR1A |

| | | | | WGM13 | WGM12 | CS12 | CS11 | CS10 | TCCR1B |
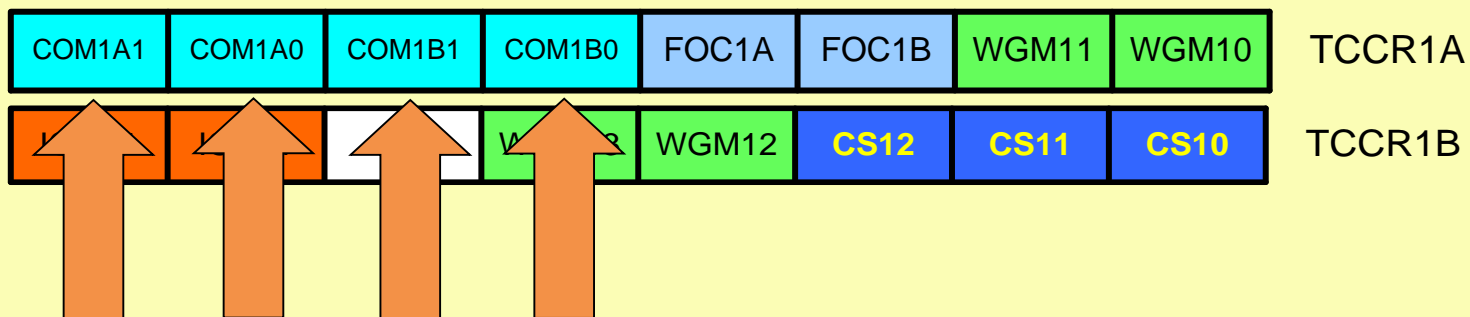
## In non PWM modes

COM1A1:COM1A0    D7 D6  Compare Output Mode for Channel A

| COM1A1 | COM1A0 | Description |
|--------|--------|-------------|
| 0 | 0 | Normal port operation, OC1A disconnected |
| 0 | 1 | Toggle OC1A on compare match |
| 1 | 0 | Clear OC1A on compare match |
| 1 | 1 | Set OC1A on compare match |

COM1B1:COM1B0    D5 D4  Compare Output Mode for Channel B

| COM1B1 | COM1B0 | Description |
|--------|--------|-------------|
| 0 | 0 | Normal port operation, OC1B disconnected |
| 0 | 1 | Toggle OC1B on compare match |
| 1 | 0 | Clear OC1B on compare match |
| 1 | 1 | Set OC1B on compare match |

# PWM modes in ATmega328p

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|-----|---|---|---|---|---|---|---|---|---|
| (0x80) | COM1A1 | COM1A0 | COM1B1 | COM1B0 | – | – | WGM11 | WGM10 | **TCCR1A** |
| Read/Write | R/W | R/W | R/W | R/W | R | R | R/W | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|-----|---|---|---|---|---|---|---|---|---|
| (0x81) | ICNC1 | ICES1 | – | WGM13 | WGM12 | CS12 | CS11 | CS10 | **TCCR1B** |
| Read/Write | R/W | R/W | R | R/W | R/W | R/W | R/W | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|-----|---|---|---|---|---|---|---|---|---|
| 0x16 (0x36) | – | – | ICF1 | – | – | OCF1B | OCF1A | TOV1 | **TIFR1** |
| Read/Write | R | R | R/W | R | R | R/W | R/W | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |



TOV1 ← | 15 ... 8 | 7 ... 0 | TCNT1H | TCNT1L ← $f_{clkT1}$ ← Prescaler ← $f_{clkI/O}$

Table 13-5. Clock Select Bit Description

| CS12 | CS11 | CS10 | Description |
|------|------|------|-------------|
| 0 | 0 | 0 | No clock source (Timer/Counter stopped). |
| 0 | 0 | 1 | clk$_{I/O}$/1 (No prescaling) |
| 0 | 1 | 0 | clk$_{I/O}$/8 (From prescaler) |
| 0 | 1 | 1 | clk$_{I/O}$/64 (From prescaler) |
| 1 | 0 | 0 | clk$_{I/O}$/256 (From prescaler) |
| 1 | 0 | 1 | clk$_{I/O}$/1024 (From prescaler) |
| 1 | 1 | 0 | External clock source on T1 pin. Clock on falling edge. |
| 1 | 1 | 1 | External clock source on T1 pin. Clock on rising edge. |

# PWM modes in ATmega328p

| BOTTOM | The counter reaches the BOTTOM when it becomes zero (0x00). |
|--------|-----------------------------------------------------------|
| MAX | The counter reaches its MAXimum when it becomes 0xFF (decimal 255). |
| TOP | The counter reaches the TOP when it becomes equal to the highest value in the count sequence. The TOP value can be assigned to be the fixed value 0xFF (MAX) or the value stored in the OCR2A Register. The assignment is dependent on the mode of operation. |

# PWM modes in ATmega328p

**Table 1**  Waveform Generation Mode Bit Description

| Mode | WGM2 | WGM1 | WGM0 | Timer/Counter Mode of Operation | TOP | Update of OCRx at | TOV Flag Set on[1][2] |
|------|------|------|------|---------------------------------|-----|-------------------|----------------------|
| 0 | 0 | 0 | 0 | Normal | 0xFF | Immediate | MAX |
| 1 | 0 | 0 | 1 | PWM, Phase Correct | 0xFF | TOP | BOTTOM |
| 2 | 0 | 1 | 0 | CTC | OCRA | Immediate | MAX |
| 3 | 0 | 1 | 1 | Fast PWM | 0xFF | BOTTOM | MAX |
| 4 | 1 | 0 | 0 | Reserved | – | – | – |
| 5 | 1 | 0 | 1 | PWM, Phase Correct | OCRA | TOP | BOTTOM |
| 6 | 1 | 1 | 0 | Reserved | – | – | – |
| 7 | 1 | 1 | 1 | Fast PWM | OCRA | BOTTOM | TOP |

Notes:

1. MAX = 0xFF
2. BOTTOM = 0x00
3. In normal operation the Timer/Counter Overflow Flag (TOV0) will be set in the same timer clock cycle as the TCNT0 becomes zero.
4. Whenever TCNT0 equals OCR0A or OCR0B, the comparator signals a match. A match will set the Output Compare Flag (OCF0A or OCF0B) at the **next timer clock cycle**.
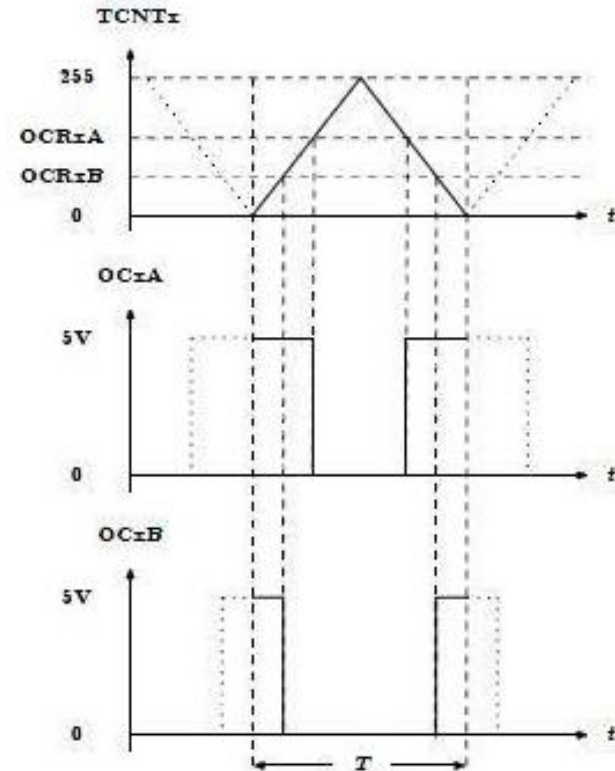
# PWM modes in ATmega328p

## Timer Modes 3 and 1
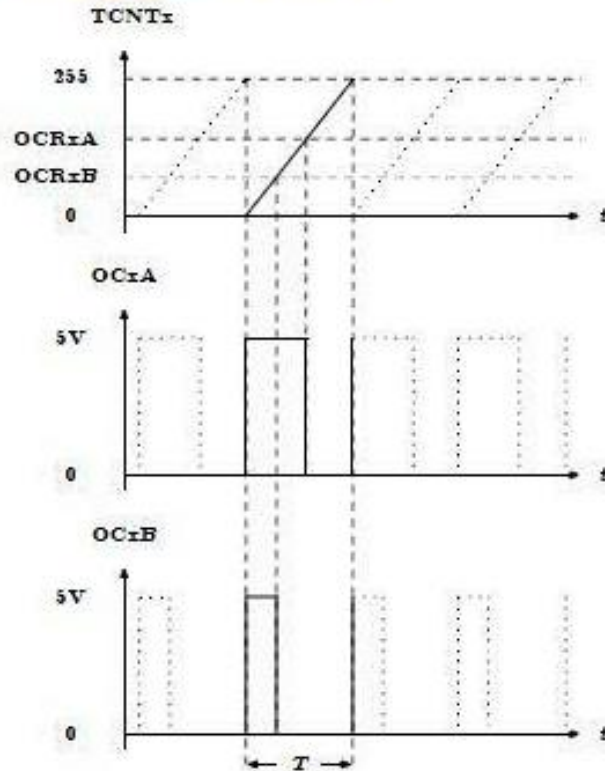


$$f_{OCnxPWM} = \frac{f_{clk\_I/O}}{N \cdot 256}$$

(a) Fast PWM

(b) Phase-Correct PWM

$$f_{OCnxPCPWM} = \frac{f_{clk\_I/O}}{N \cdot 510}$$
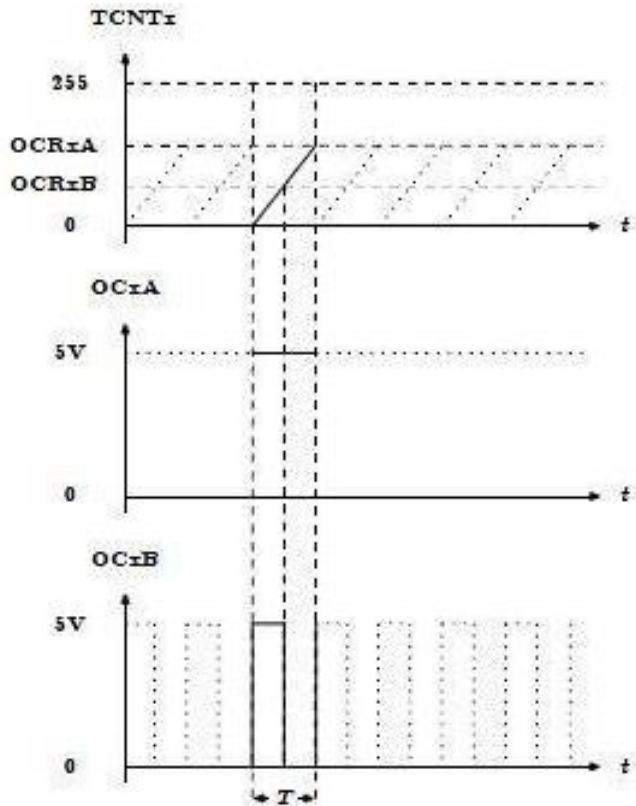
$f_{OC2B} = f_{CLK} / 256 = 16\ \text{MHz} / 256 = 62.5\ \text{KHz} \simeq 64\ \text{KHz}$

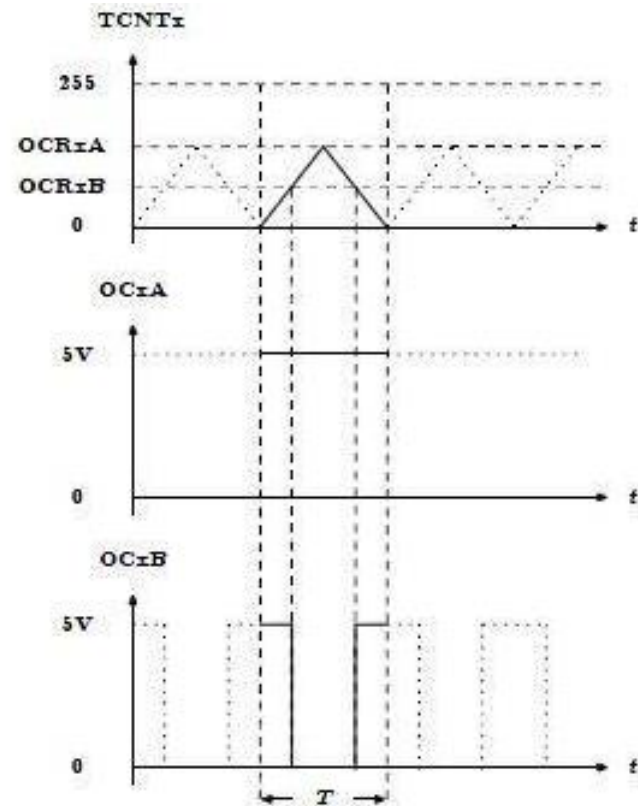$f_{OC2B} = f_{CLK} / 510 = 16\ \text{MHz} / 510 = 31.3725\ \text{MHz}$

For N = 1

# PWM modes in ATmega328p

**Timer Modes 7 and 5**



(a) Fast PWM         (b) Phase-Correct PWM

OCRxA = Frequency of oscillator / (Frequency of generated wave $\times$ N) – 1
(N = prescaler)
Duty cycle = (OCRxB + 1) / (OCRxA + 1) $\times$ 100%.

Lawrence.Cheung@EIE3105

# PWM modes in ATmega328p
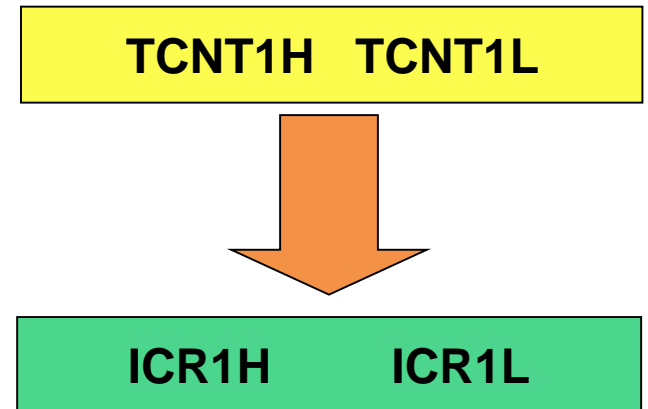
```
#include "avr/io.h"
int main(void)
{
        TCCR0A = (1 << COM0A1) | (0 << COM0A0) | //00
                (1 << COM0B1) | (0 << COM0B0) |
                (1 << WGM01)  | (1 << WGM00);
        TCCR0B = (1 << WGM02) |
                (0 << CS02)   | (0 << CS01)   | (1 << CS00);

        OCR0A = 249;  //64kHz
        OCR0B = 49;   //20% duty cycle
        DDRD = 0b00100000; // PD5 (OC0B)

        while(1);
}
```
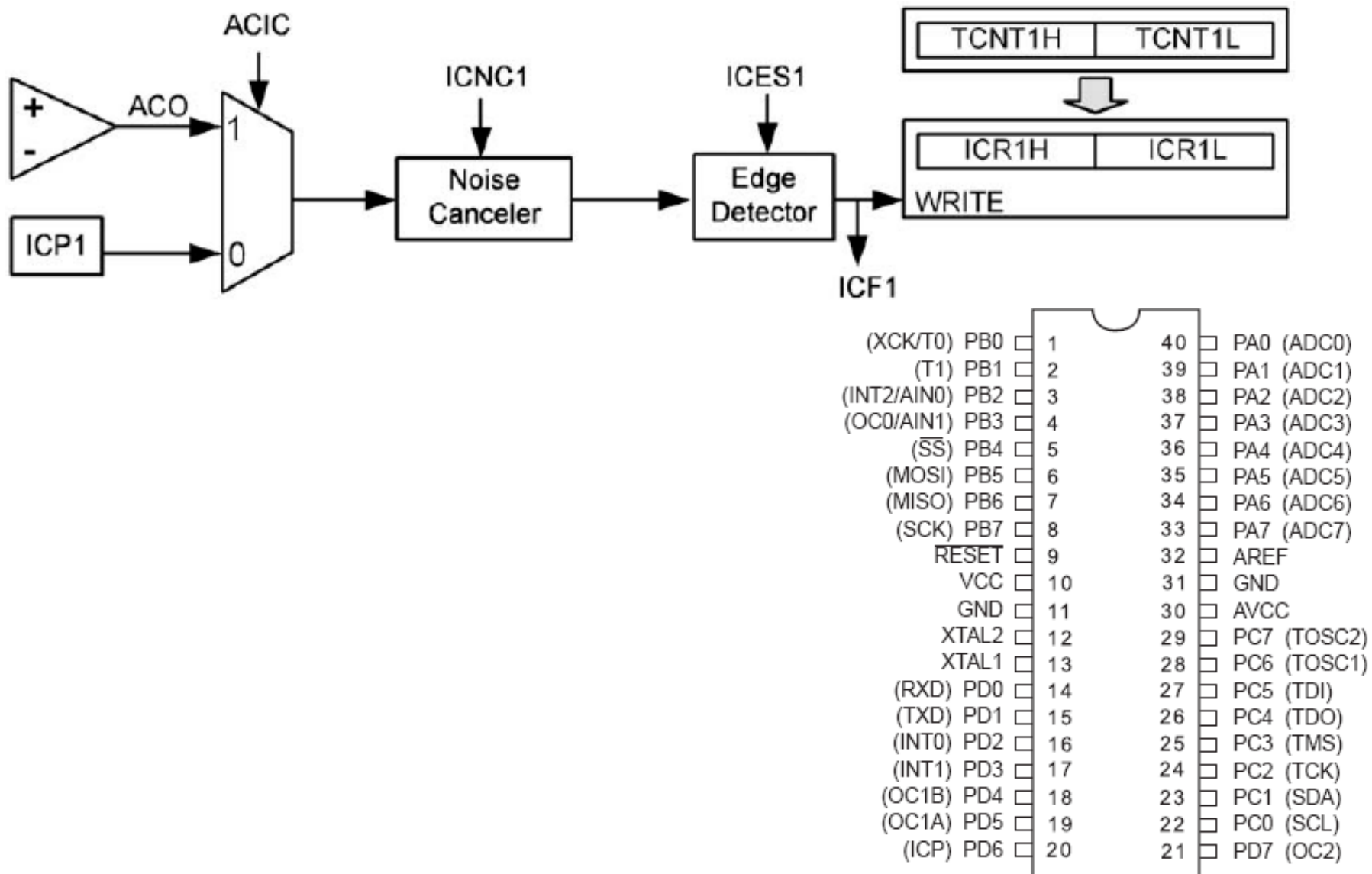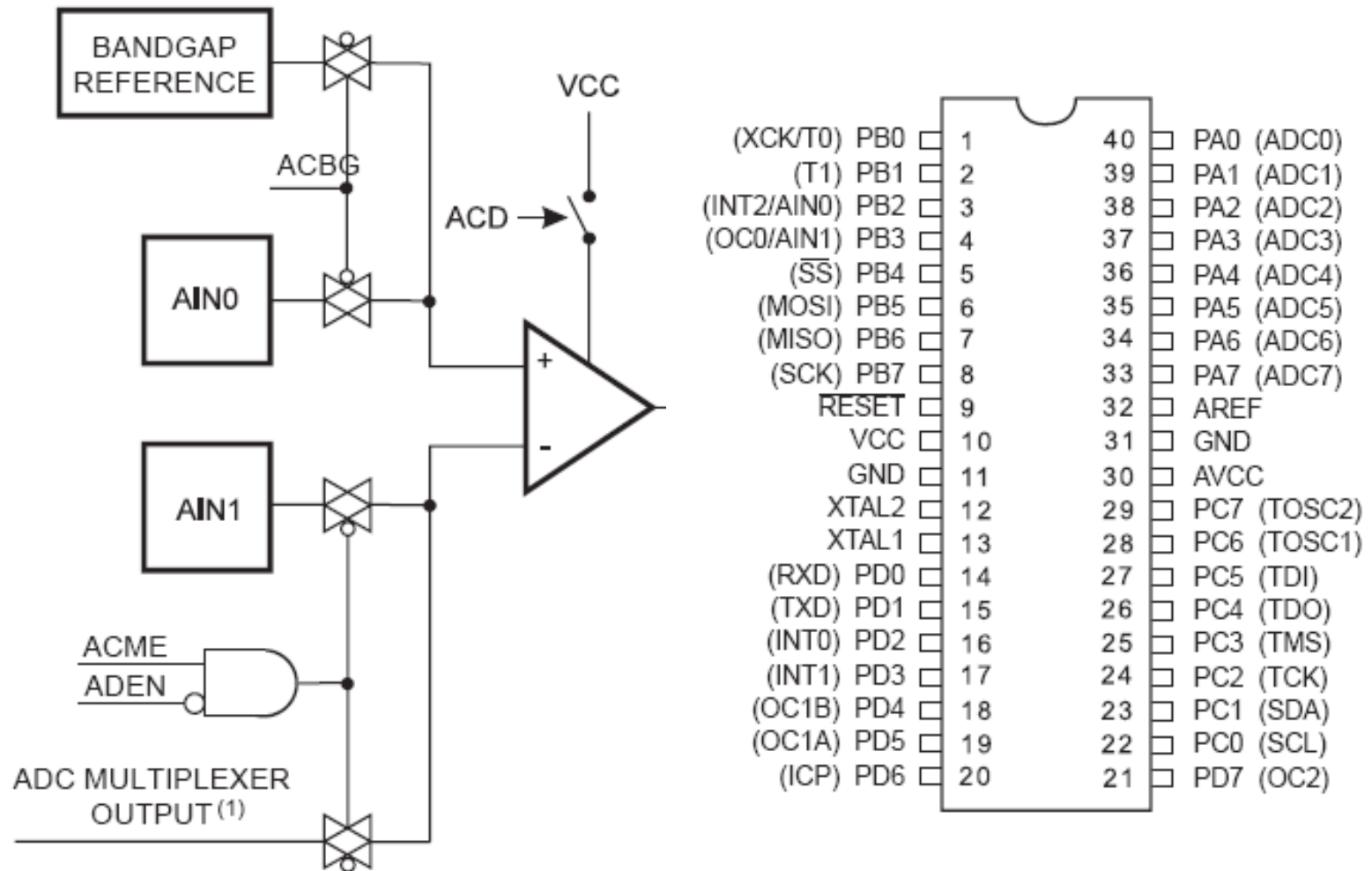
# Capturing
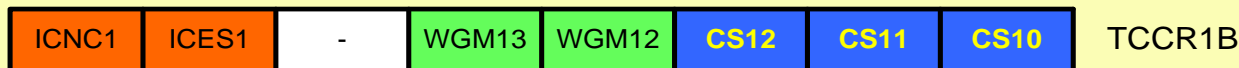
- Usages
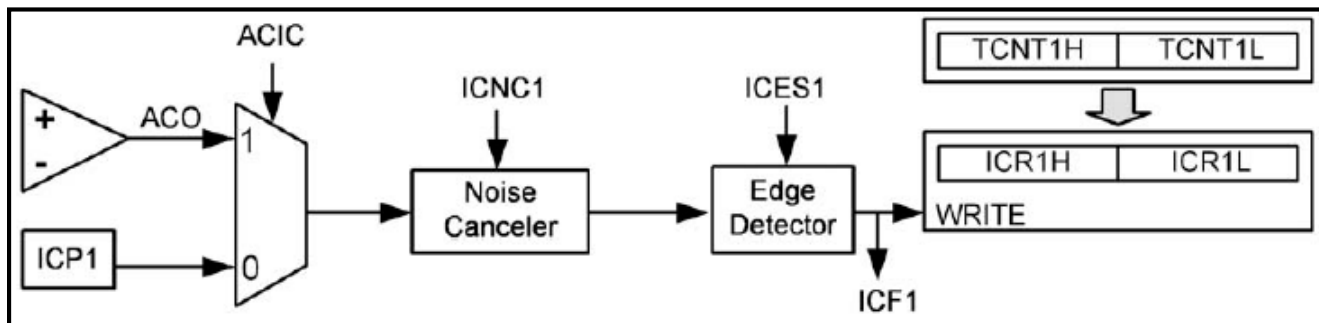  - Measuring duty cycle
  - Measuring period

| TCNT1H   TCNT1L |
|---|

$$\Downarrow$$

| ICR1H        ICR1L |
|---|

# Capturing

# Comparator

ICNC1 | ICES1 | - | WGM13 | WGM12 | CS12 | CS11 | CS10 | TCCR1B

**ICNC1:** Input Capture Noise Canceller
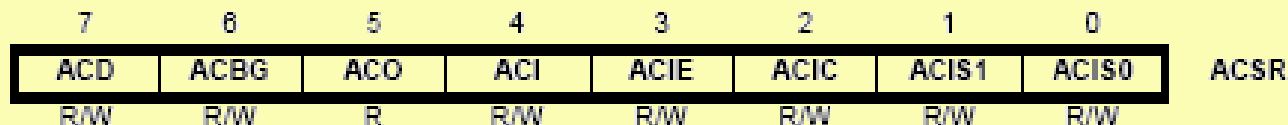     0:disabled
     1:Enabled (captures after 4 successive equal valued samples)
**ICSES1:** Input Capture Edge Select
     0: Falling edge
     1: Rising edge

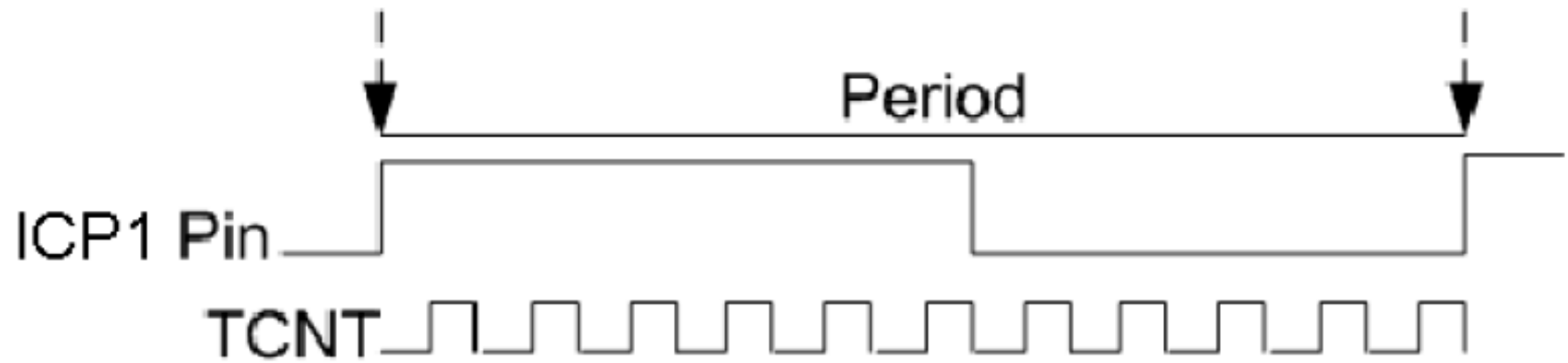| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|
| ACD | ACBG | ACO | ACI | ACIE | ACIC | ACIS1 | ACIS0 | ACSR |
| R/W | R/W | R | R/W | R/W | R/W | R/W | R/W | |

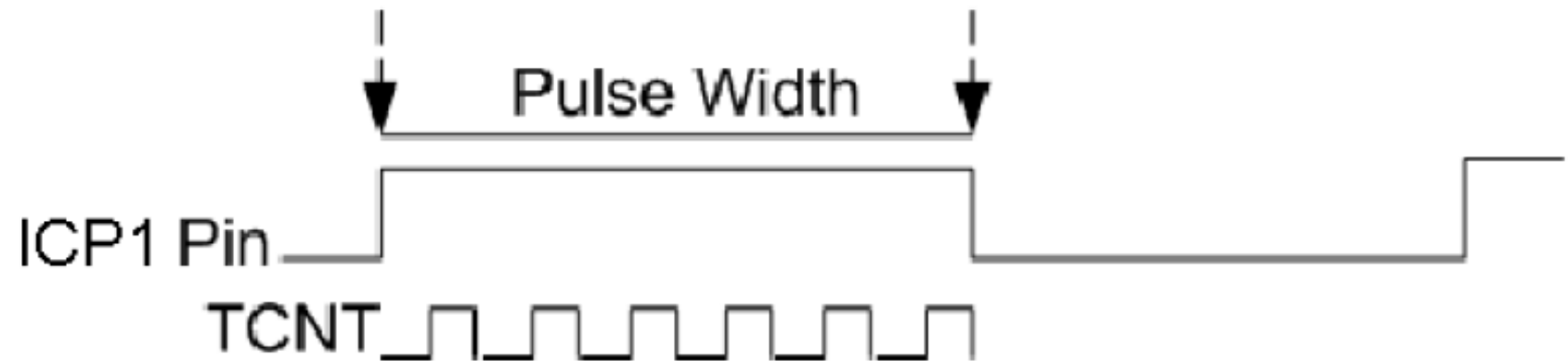**ACIC:** Analog Comparator Input Capture Enable
     0: ICP1 provides the capture signal
     1: analog comparator is connected to the capturer

# Measuring duty cycle and period



Measuring Period in Terms of the Number of Clocks Counted By TCNT

Measuring Pulse Width in Terms of the Number of Clocks Counted By TCNT

# Capturing in ATmega328p

```
#include "avr/io.h"
int main()
{
        //measure the pulse width of a pulse
        unsigned char t1;
        DDRD = 0xFF;        //PORTD as output
        PORTB = 0xFF;
        TCCR1A = 0;         //Timer Mode = Normal
        TCCR1B = (1 <<ICES1) |
                            (1 << CS12) | (0 << CS11) | (0 << CS10);
        //rising edge, prescaler = 256, no noise canceller

        TIFR1 = (1<<ICF1);          //clear ICF1 (The Input Capture Flag)
        while ((TIFR1&(1<<ICF1)) == 0); //wait while ICF1 is clear
        t1 = ICR1L;        //first edge value (Input Capture Register,
low byte)
        TIFR1 = (1<<ICF1);          //clear ICF1
        TCCR1B = (0 <<ICES1) | (1 << CS12) | (0 << CS11) | (0 << CS10);
        //falling edge
        while ((TIFR1&(1<<ICF1)) == 0); //wait while ICF1 is clear
        PORTD = ICR1L - t1;        //pulse width = falling - rising
        TIFR1 = (1<<ICF1);          //clear ICF1
        while (1);
}
```

# Capturing in ATmega328p

```
#include "avr/io.h"
int main()
{
        //measure the period of a pulse
        unsigned char t1;
        DDRD = 0xFF;        //PORTD as output
        PORTB = 0xFF;
        TCCR1A = 0;         //Timer Mode = Normal
        TCCR1B = (1 <<ICES1) |
                            (1 << CS12) | (0 << CS11) | (0 << CS10);
        //rising edge, prescaler = 256, no noise canceller

        TIFR1 = (1<<ICF1);          //clear ICF1 (The Input Capture Flag)
        while ((TIFR1&(1<<ICF1)) == 0); //wait while ICF1 is clear
        t1 = ICR1L;         //first edge value (Input Capture Register,
low byte)
        TIFR1 = (1<<ICF1);          //clear ICF1
        while ((TIFR1&(1<<ICF1)) == 0); //wait while ICF1 is clear
        PORTD = ICR1L – t1;         //period = second edge – first edge
        TIFR1 = (1<<ICF1);          //clear ICF1
        while (1);
}
```

# Reference Readings

- Chapter 16 and 17 – *The AVR Microcontroller and Embedded Systems : Using Assembly and C*, M. A. Mazidi, S. Naimi, and S. Naimi, Pearson, 2014.

End