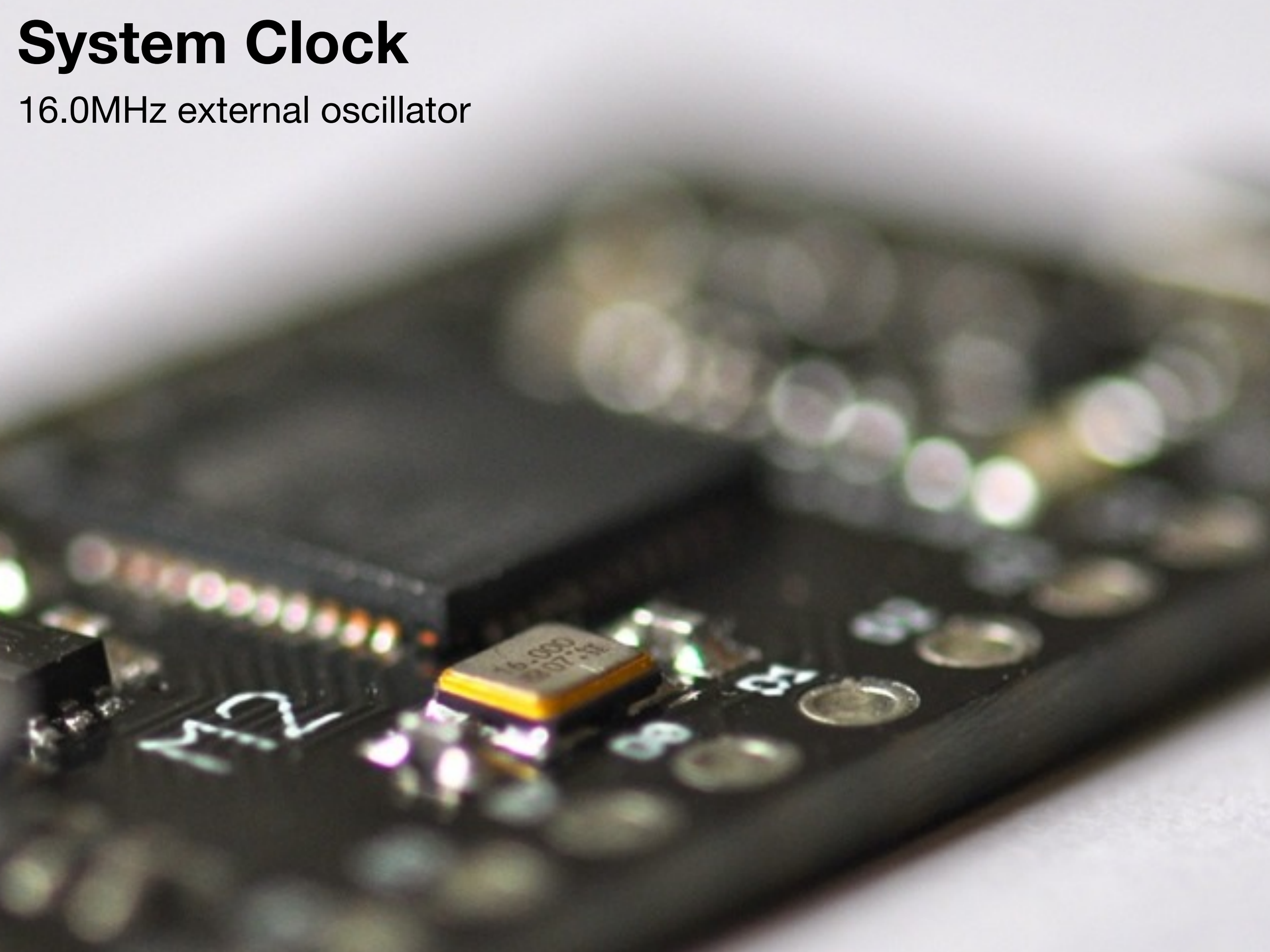
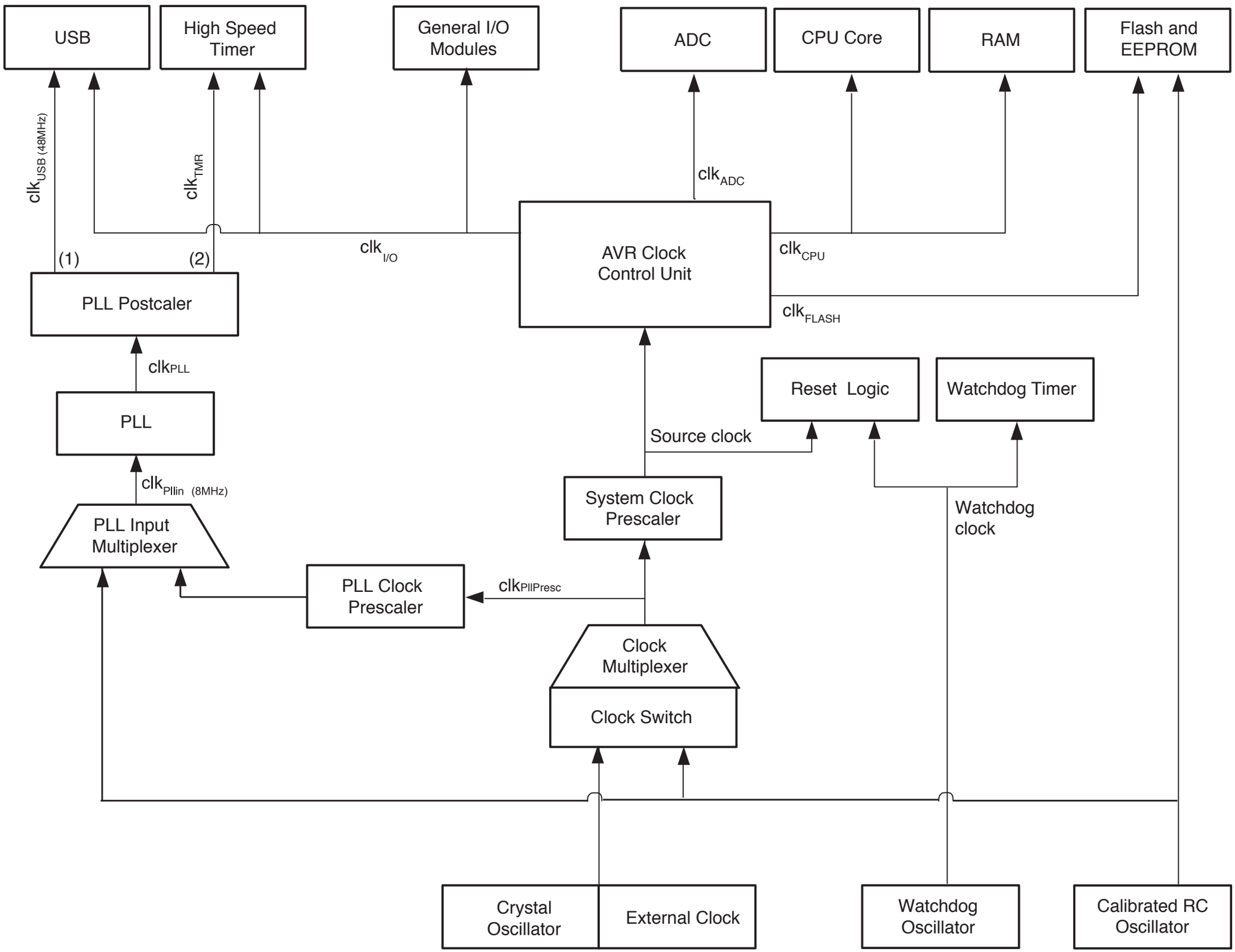
ATmega32U4 Clock & Timers

System Clock

16.0MHz external oscillator



System Clock



System Clock

```
m_clockdivide(N);
// prescale the 16MHz system clock
// by 2^N (N=0..8)
// default is N=3
```

Timer 0

8-bit timer/counter

2 compare outputs (OC0A, OC0B)

PWM capability

timer overflow flag

6 timer modes

Timers 1 & 3

8/10/16-bit timer/counter

4 compare outputs (OC1A,OC1B,OC1C & OC3A)

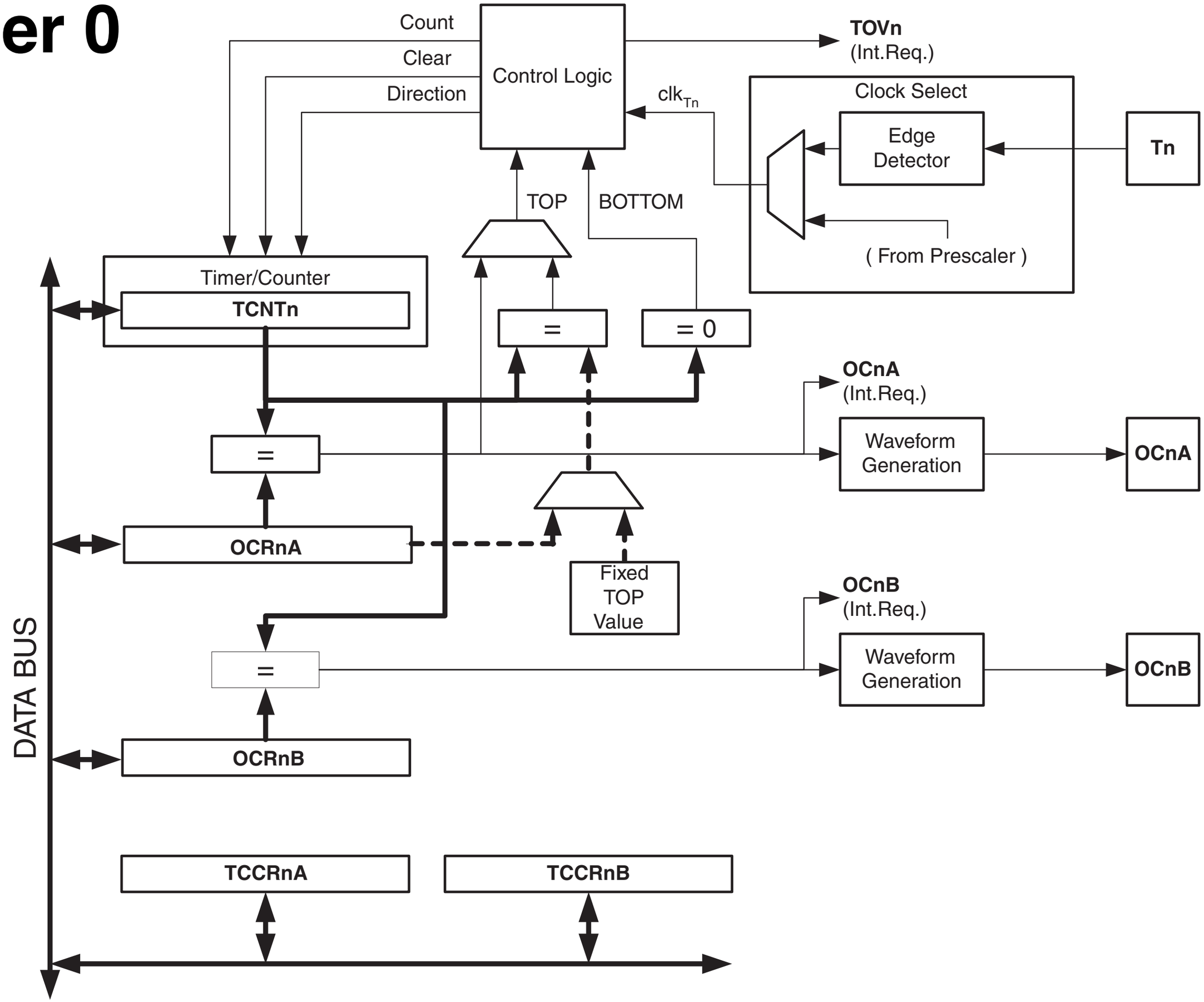
2 capture inputs (IPC1 & IPC3)

PWM capability

timer overflow flag

many timer modes

Timer 0


Count
Clear
Direction
Control Logic
TOVn
(Int.Req.)
clk$_{Tn}$
Clock Select
Edge
Detector
Tn
(From Prescaler)
TOP
BOTTOM
Timer/Counter
TCNTn
=
= 0
OCnA
(Int.Req.)
Waveform
Generation
OCnA
=
OCRnA
Fixed
TOP
Value
OCnB
(Int.Req.)
Waveform
Generation
OCnB
=
OCRnB
DATA BUS
TCCRnA
TCCRnB

Timer 0 : registers

`TCNT0`	timer value
`TCCR0A`	control register A
`TCCR0B`	control register B
`OCR0A`	compare register A
`OCR0B`	compare register B
`TIFR0`	interrupt flags

Timer 0 : bits

select the clock prescaler

```
TCCR0B : CS02
TCCR0B : CS01
TCCR0B : CS00
```

set the timer mode

```
TCCR0B : WGM02
TCCR0A : WGM01
TCCR0A : WGM00
```

set the compare options

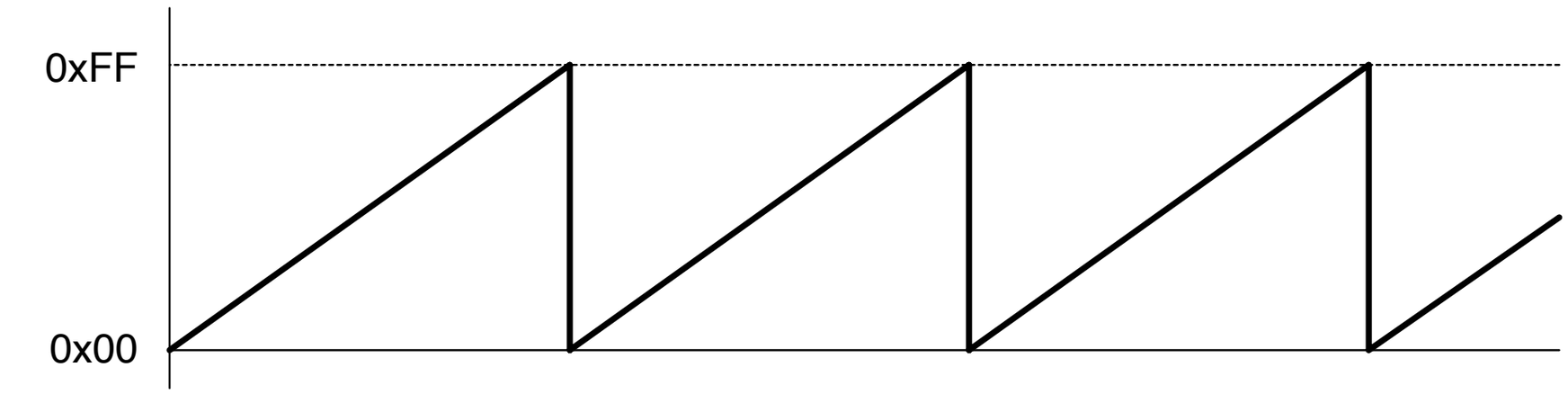
```
TCCR0A : COM0A0
TCCR0A : COM0A1
TCCR0B : COM0B0
TCCR0B : COM0B1
```

watch the flags

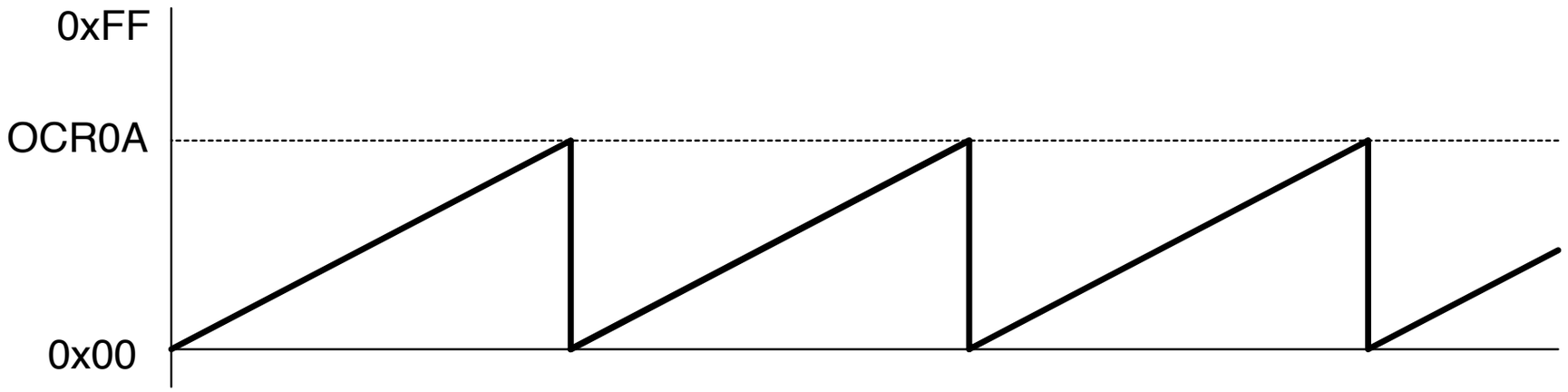
```
TIFR0 : OCF0A
TIFR0 : OCF0B
TIFR0 : TOV0
```

Timer 0 : modes

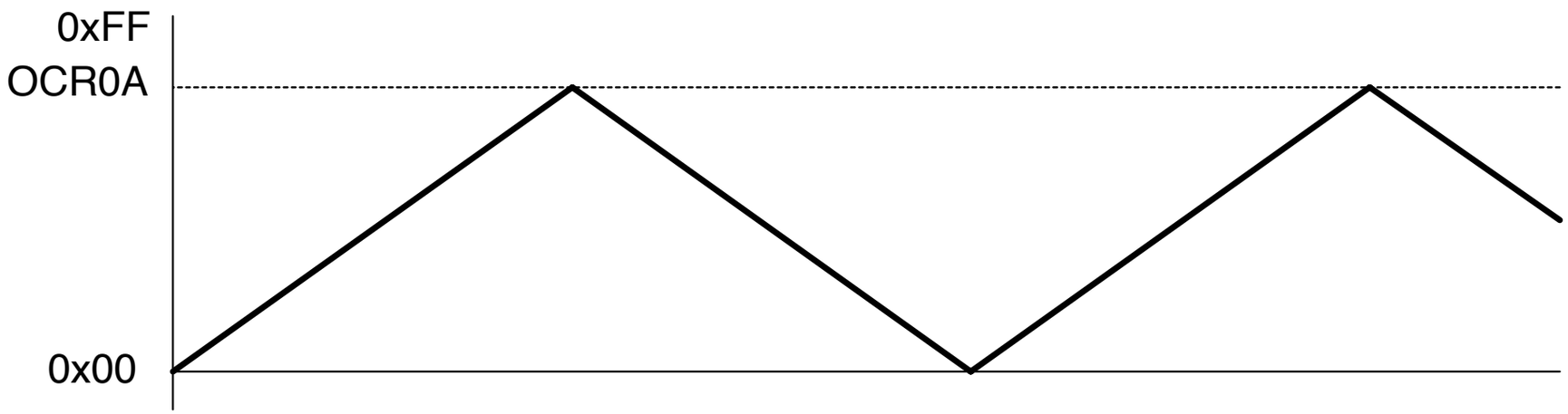
WGMxx



0,0,0



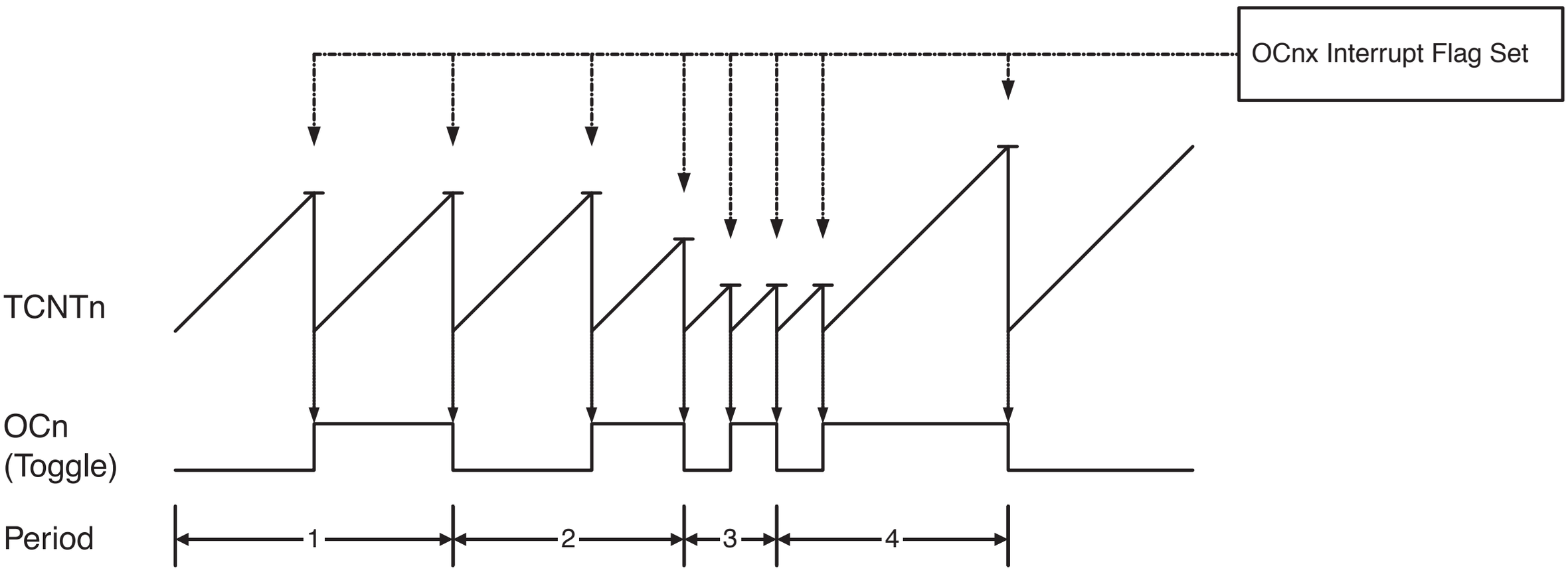
0,1,0



1,0,1

```
TCCR0B : WGM02 = 0
TCCR0A : WGM01 = 1
TCCR0A : WGM00 = 0
```

```
TCCR0A : COM0A1 = 0
TCCR0A : COM0A0 = 1
```



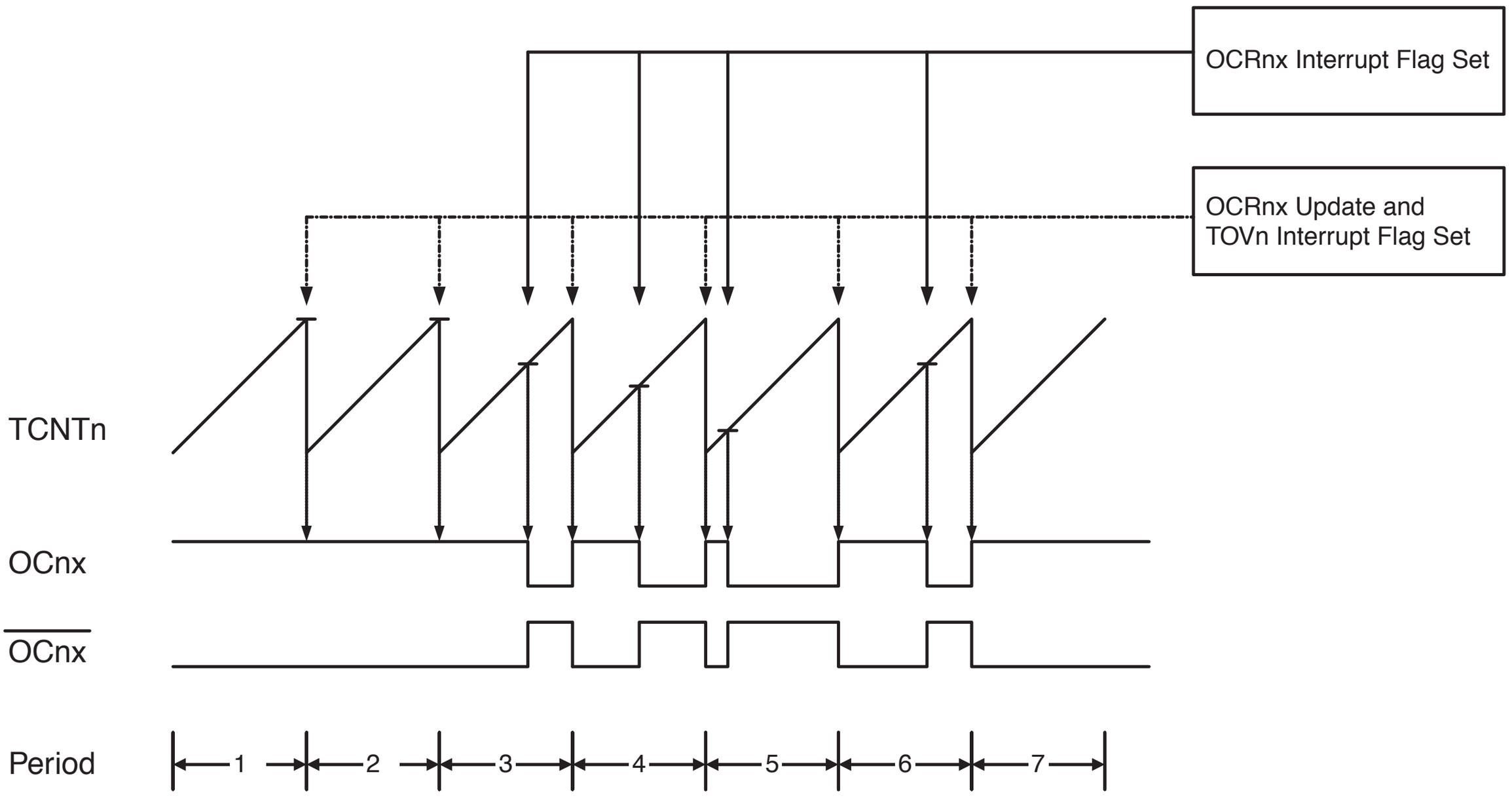
`TCCR0B : WGM02 = 1`

`TCCR0A : WGM01 = 1`

`TCCR0A : WGM00 = 1`

`TCCR0A : COM0B1 = 1`

`TCCR0A : COM0B0 = 0`



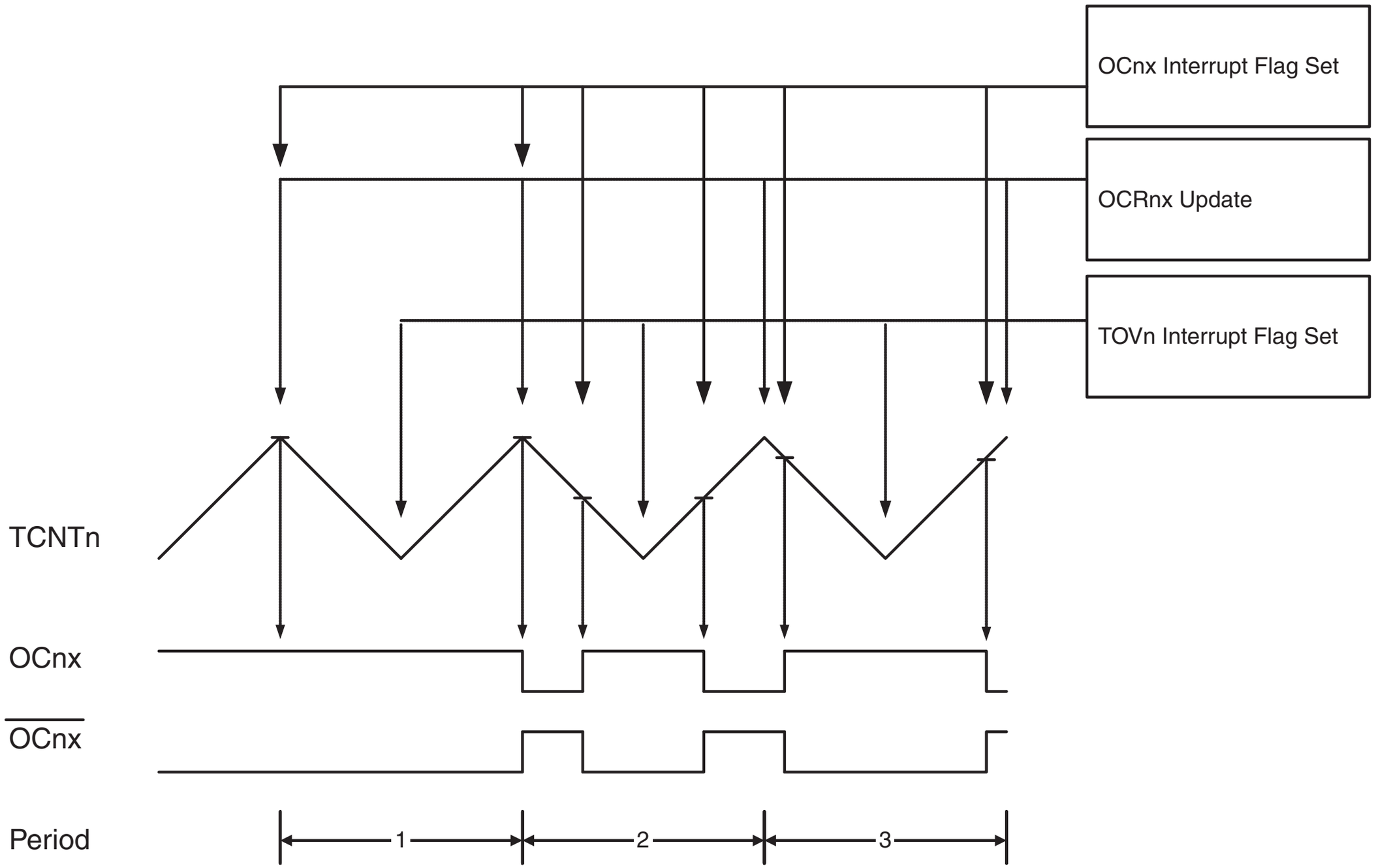
`TCCR0B : WGM02 = 1`

`TCCR0A : WGM01 = 0`

`TCCR0A : WGM00 = 1`

`TCCR0A : COM0B1 = 1`

`TCCR0A : COM0B0 = 1`



Timer 0 : example

```
OCR0B = 0xA3;
OCR0B = 0x84;


set(TCCR0B,WGM02);          // MODE: up to OCR0A
set(TCCR0A,WGM01);          // ^
set(TCCR0A,WGM00);          // ^

set(TCCR0A,COM0B1);         // clear at OCR0B, set at OCR0A
clear(TCCR0A,COM0B0);       // ^

clear(TCCR0B,CS02);         // set prescaler to /1
clear(TCCR0B,CS01);         // ^
set(TCCR0B,CS00);           // ^

set(DDRD,0);                // set D0 as output
```