Timers:

- The main idea is that: register increases or decreases its value automatically at a predefined rate (supplied by user).

- Timers are necessary to:

  - generate time delays, waveforms or to count events.
  - PWM generation, capturing events,…etc.

Prescaler:

- The Prescaler is used to divide the CPU clock frequency to be able to control the clock frequency of the timer itself.
- The Prescaler is used when it is necessary to measure longer periods of time.
- The Prescaler can be used to get the following clock for timer:
  - No Clock (Timer Stop)
  - No Prescaler (Clock = FCPU)
  - FCPU/8, FCPU/64, FCPU/256 and FCPU/1024.
- We don't reduce the FCPU, the actual processor frequency remains the same, but we use it to feed a divider that then feeds the timer/counter register.

ATmega16/32 have 3 Timer/counter each one with Separate Prescaler.

- Timer 0 → 8-bit timer.
- Timer 1 → 16-bit timer.
- Timer 2 → 8 bit timer.

Apart from normal operation, these three timers can be either operated in:

- Normal mode (Overflow).
- Clear Timer on Compare mode (CTC) or Compare Match mode.
- Pulse Width Modulation mode (PWM).

Basic registers and flags of the Timers

TCNTn: Timer / Counter Register

Every timer has a timer/counter register. It is zero upon reset. We can access value or write a value to this register. It counts up with each clock pulse.

TOVn: Timer Overflow Flag

Each timer has a Timer Overflow flag. When the timer overflows, this flag will get set.

TCCRn: Timer Counter Control Register

This register is used for setting the modes of timer/counter.

OCRn: Output Compare Register

The value in this register is compared with the content of the TCNTn register. When they are equal, the OCFn flag will get set.