Timers

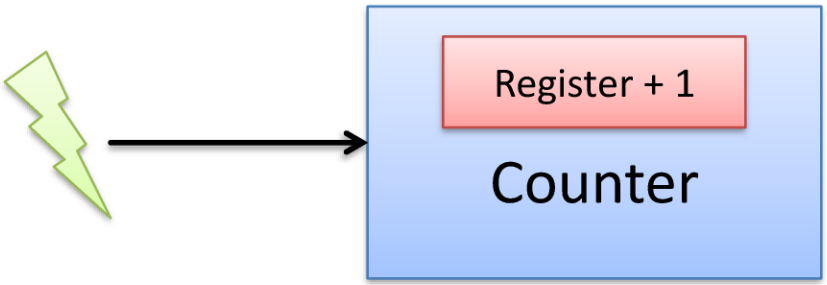
Networks and Embedded Software

Module 4.2.4

by Wolfgang Neff

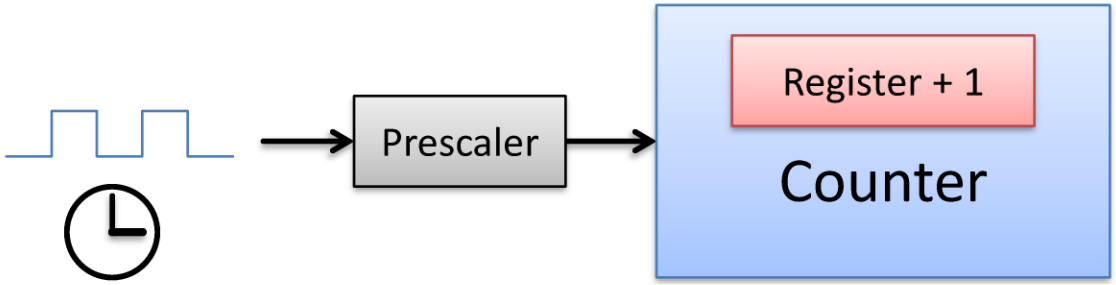
Counters

- Counters count events
  - Number of events stored in a register
  - Each event increments this register



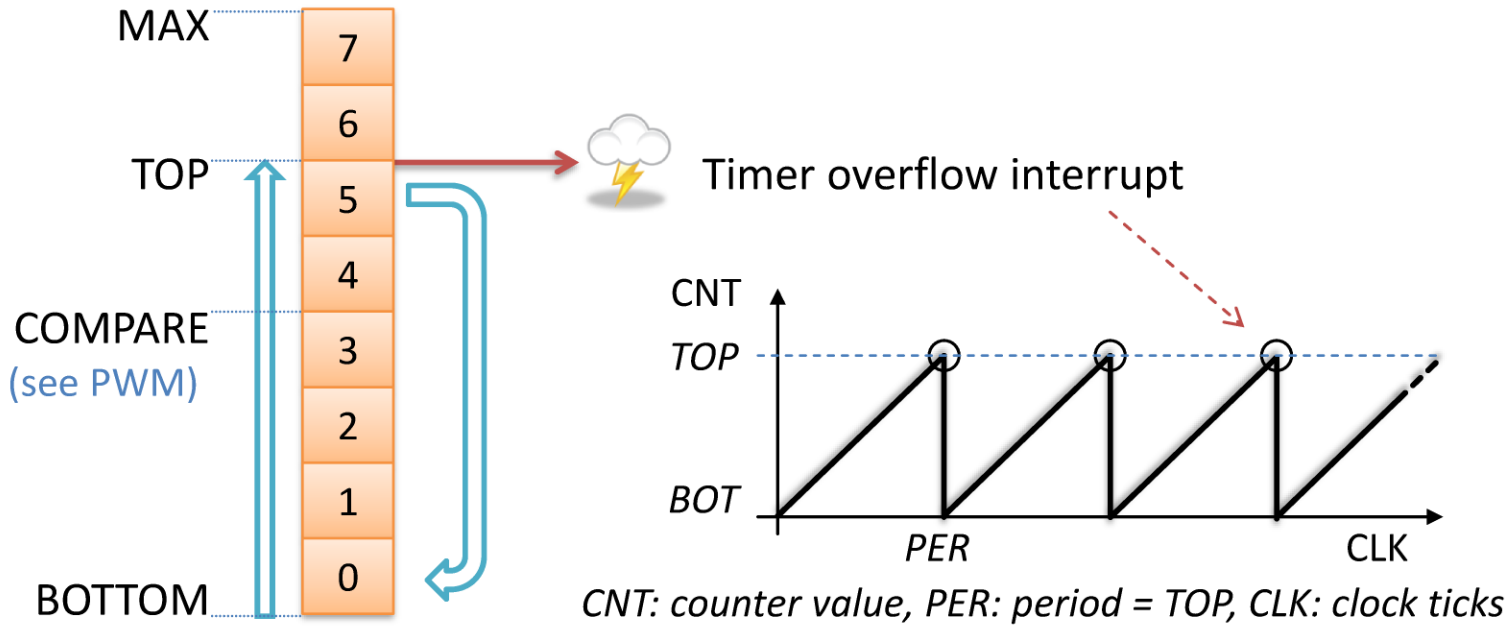
Timers (1)

- Timers count clock ticks
  - Primary source is system clock (f_{CPU})
  - Clock speed is reduced by a prescaler



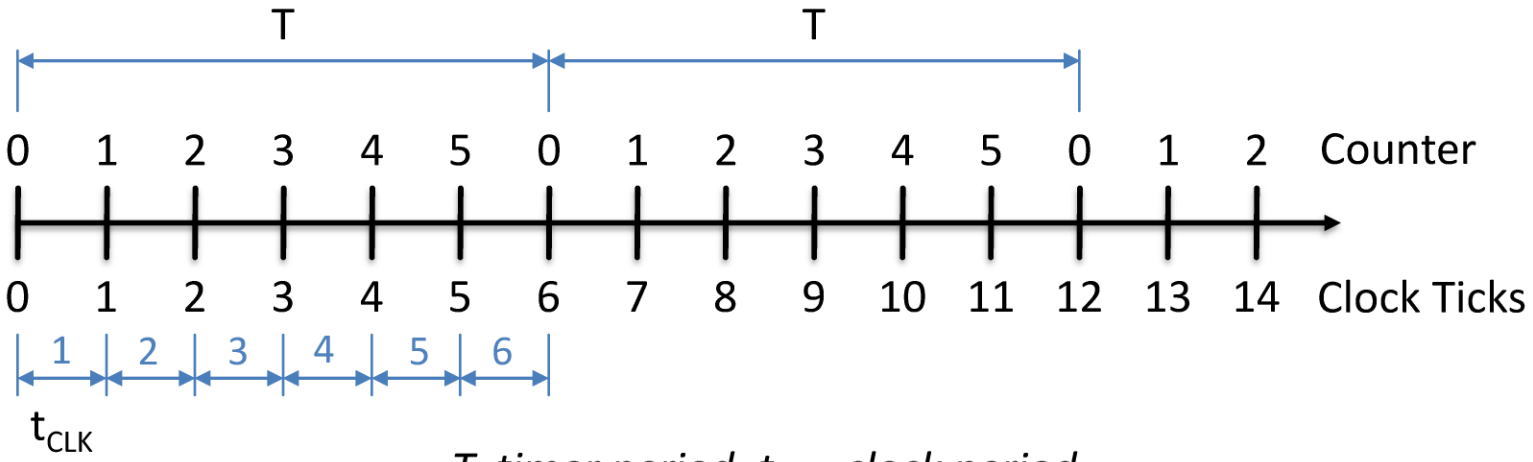
Timers (2)

- Operating mode



Timers (3)

- Operating mode (continue)
  - Example: TOP = 5



T: timer period, t_{CLK}: clock period

$$T = (TOP+1) \cdot t_{CLK}$$

Timers (4)

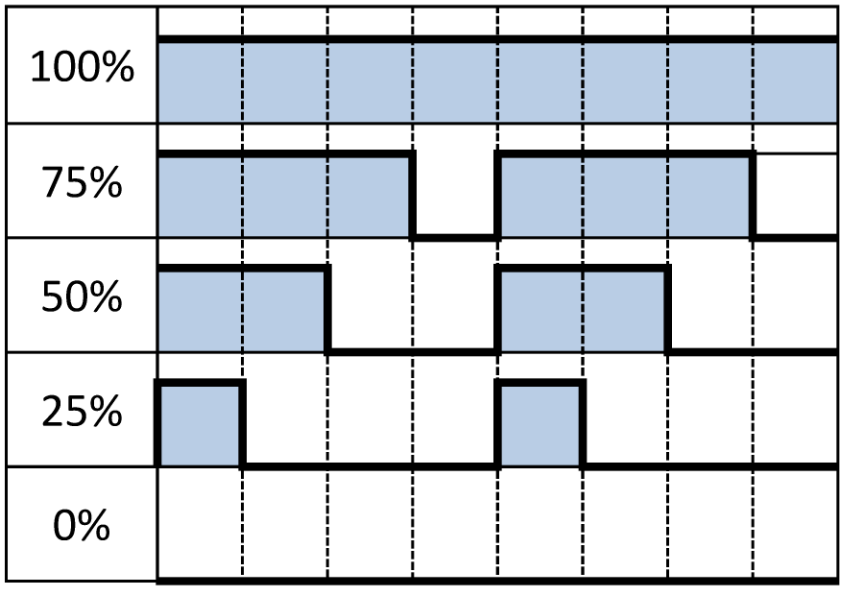
- Basic Formulas
  - Prescaler
    - $f_{CLK} = \frac{f_{CPU}}{n}$
  - Period
    - $T = (TOP + 1) \cdot t_{CLK} = \frac{TOP+1}{f_{CLK}} = \frac{n \cdot (TOP+1)}{f_{CPU}}$
  - Ticks per second (TPS)
    - $TPS = f = \frac{1}{T} = \frac{f_{CLK}}{TOP+1} = \frac{f_{CPU}}{n \cdot (TOP+1)}$

Timers (5)

- Example
  - f_{CPU} = 2 MHz, Prescale value = 8, TOP = 24999
    - Frequency of CPU: **2000000** Hz
    - Frequency of clock: 2000000 Hz / **8** = 250,000 Hz
    - Frequency of timer: 250,000 Hz / (**24999**+1) = 10 Hz
  - Ticks per second: 10
  - Period: 0,1 s

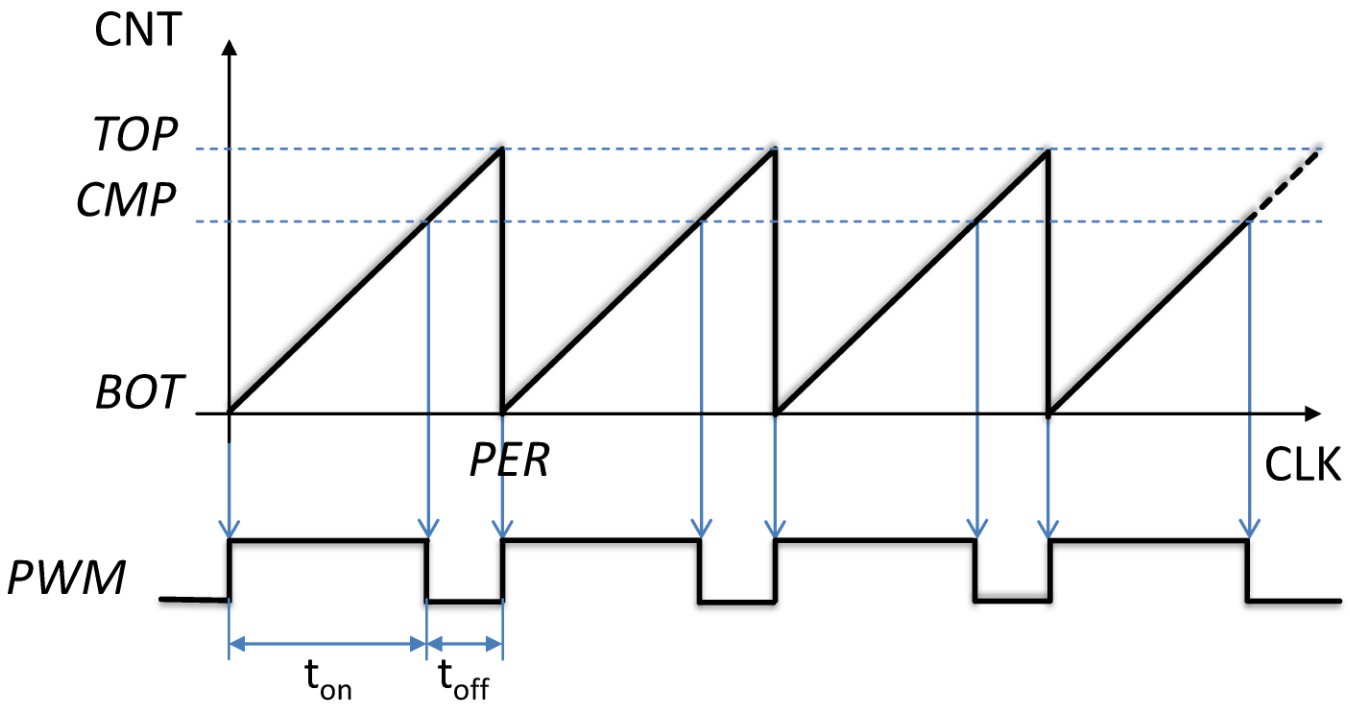
PWM (1)

- Pulse-Width Modulation
  - Digital pins are either high or low
  - Time enables inter-mediate values
  - Inertia for averaging necessary
  - Alternative: Digital Analog Converter



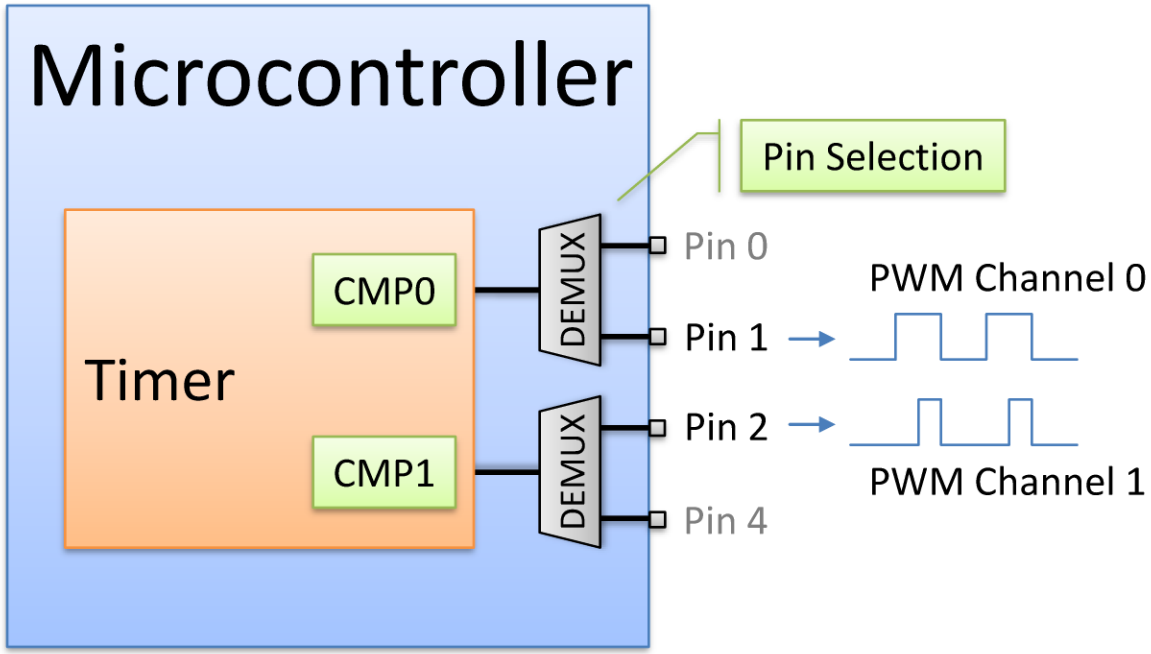
PWM (2)

- Generation



PWM (3)

- Architecture



PWM (4)

- Basic Formulas
  - Frequency
    - $f_{PWM} = \frac{f_{CPU}}{n \cdot (TOP+1)} = \frac{f_{CLK}}{TOP+1}$
  - Duty cycle ratio
    - $d = \frac{t_{on}}{t_{on}+t_{off}} = \frac{t_{on}}{T} = \frac{CMP+1}{TOP+1}$
  - Resolution
    - R = $\frac{\log(TOP+1)}{\log 2}$