Timers and Interrupts

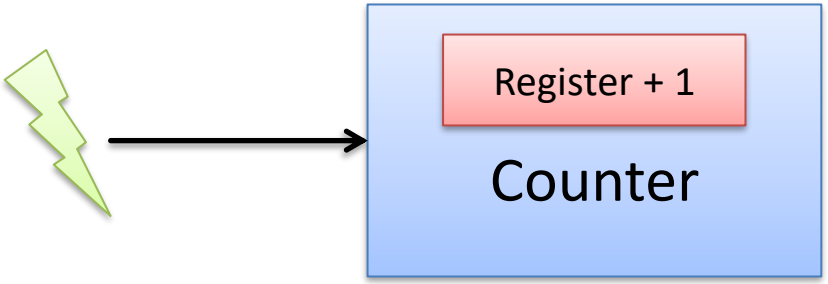
Networks and Embedded Systems

Second Grade Level

Wolfgang Neff

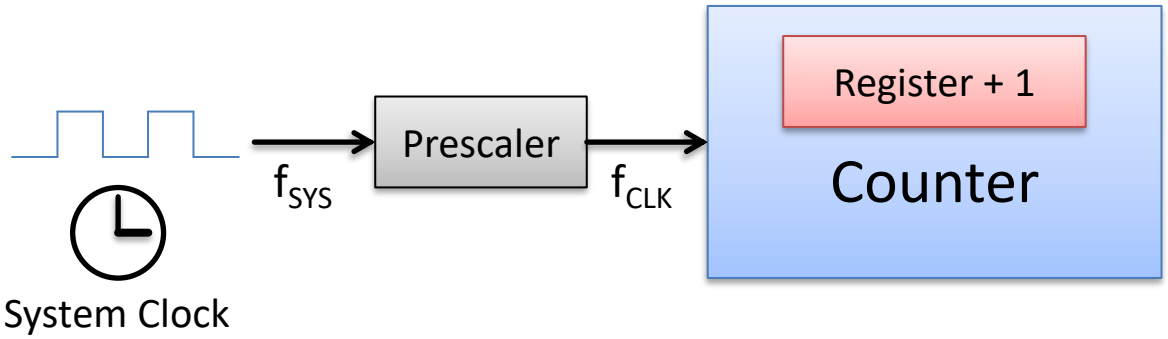
Counters

- 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
  - Clock speed is reduced by a prescaler

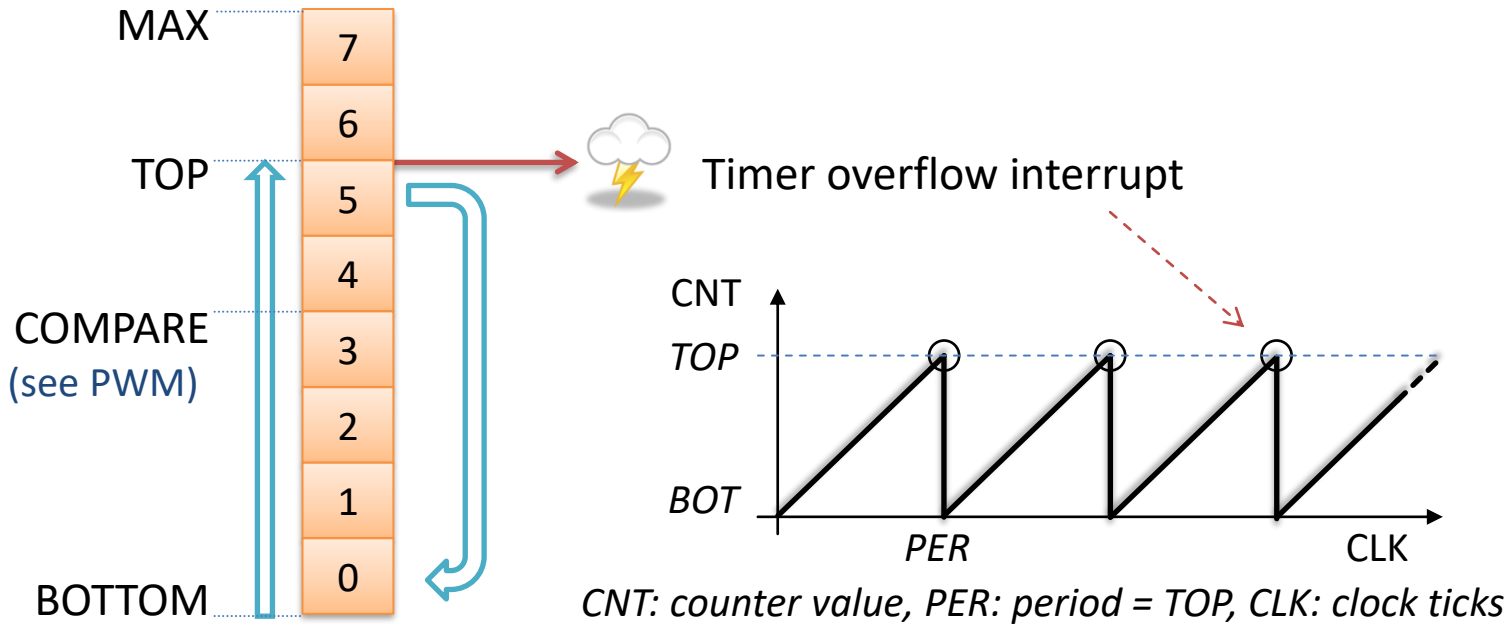


$$f_{CLK} = \frac{f_{SYS}}{n}$$

f_{SYS}: System Frequency
f_{CLK}: Clock Frequency
n: Prescaler Value

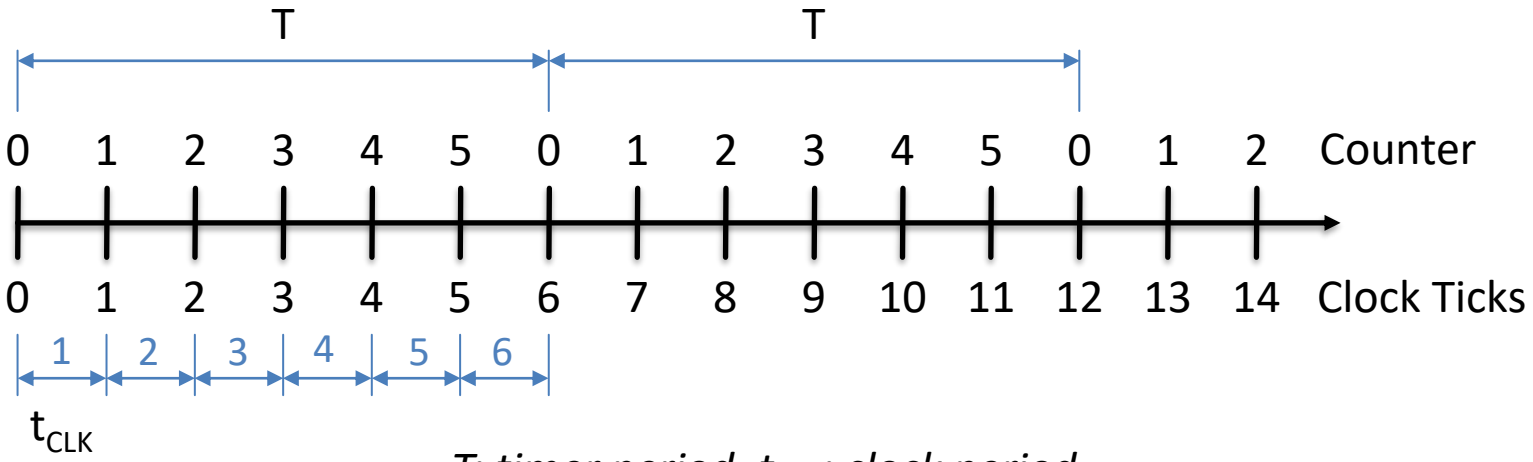
Timers (2)

- Operating mode



Timers (3)

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

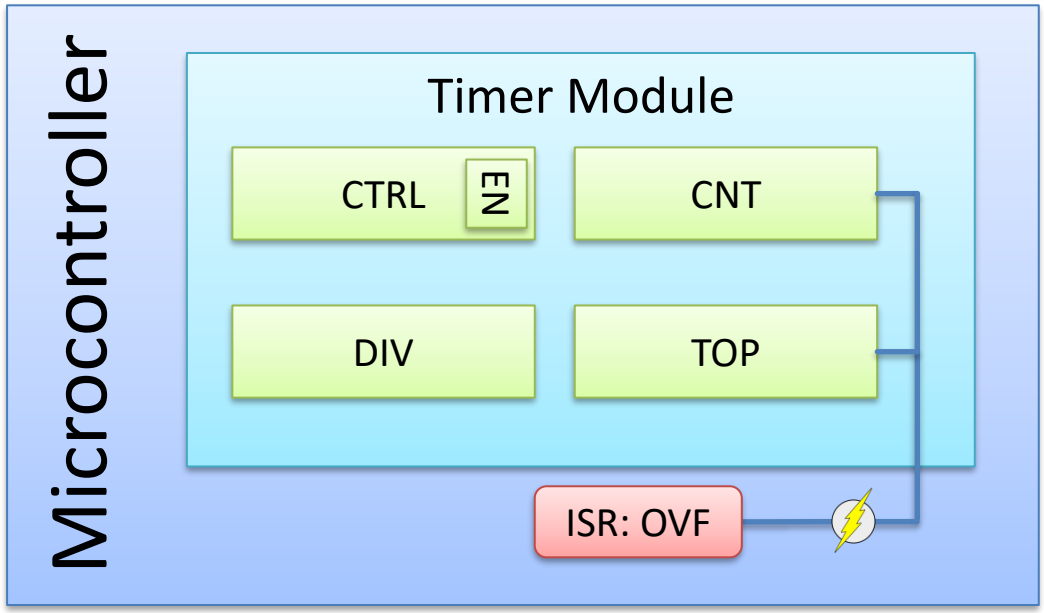


T: timer period, t_{CLK}: clock period

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

Timers (4)

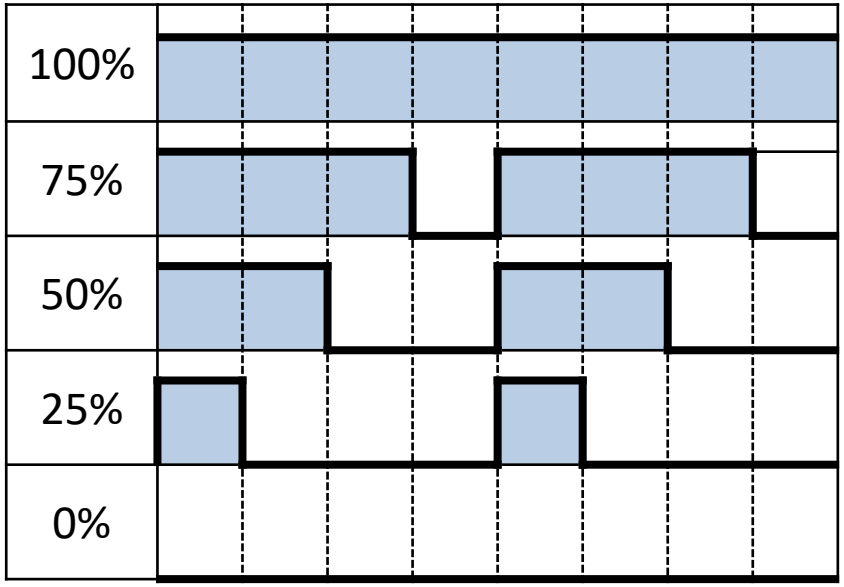
- Timer Module



CTRL: Control Register
CTRL.EN: Enable Bit
CNT: Counter Value
DIV: Clock Divider
TOP: Top Value
OVF: Overflow

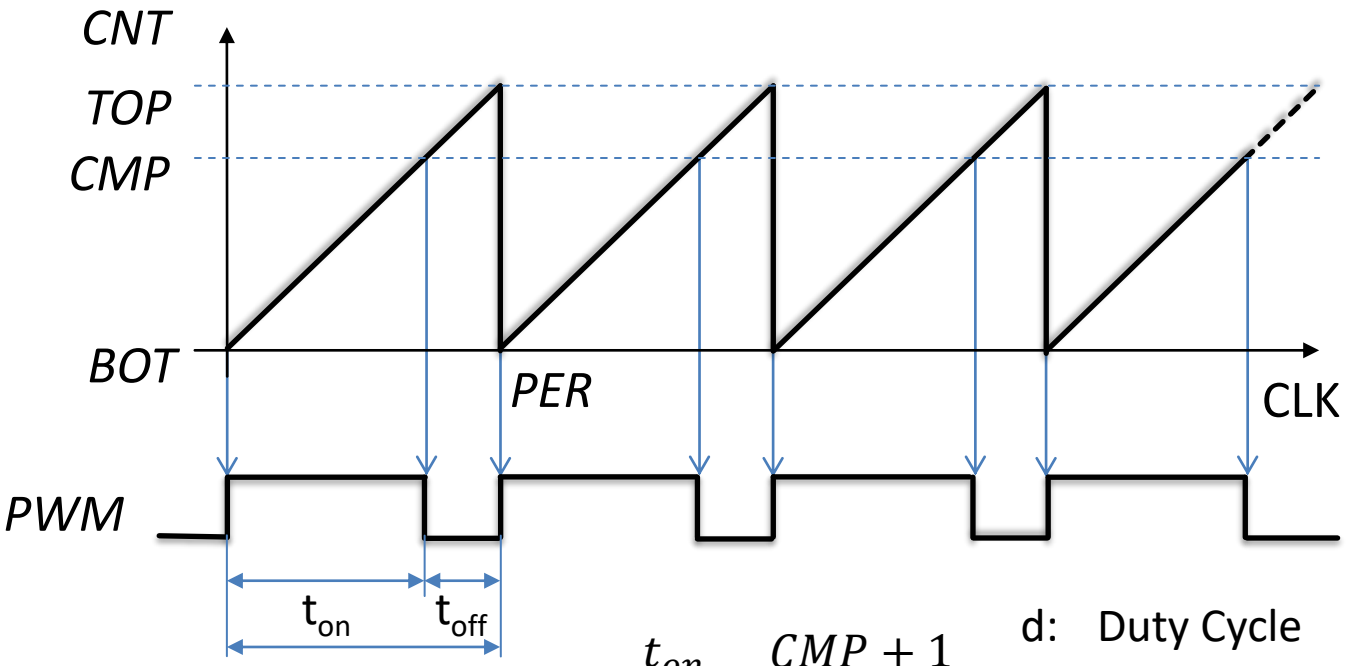
PWM (1)

- 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

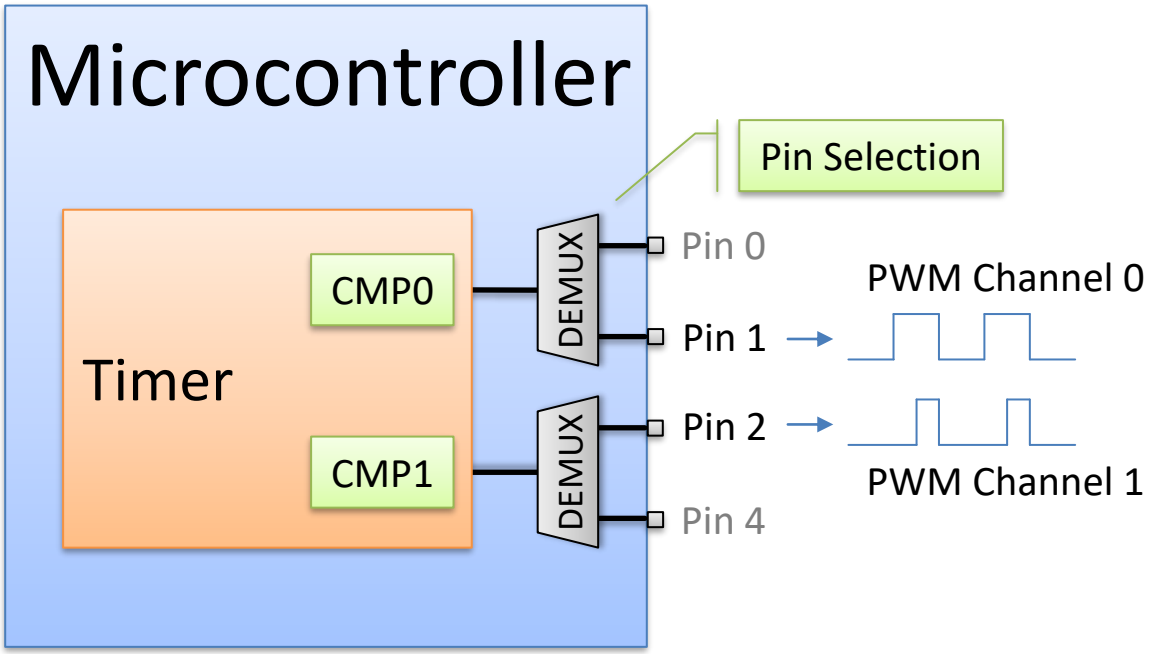


$$d = \frac{t_{on}}{T} = \frac{CMP + 1}{PER + 1}$$

d: Duty Cycle
t_{on}: On Time
T: Period (PER)

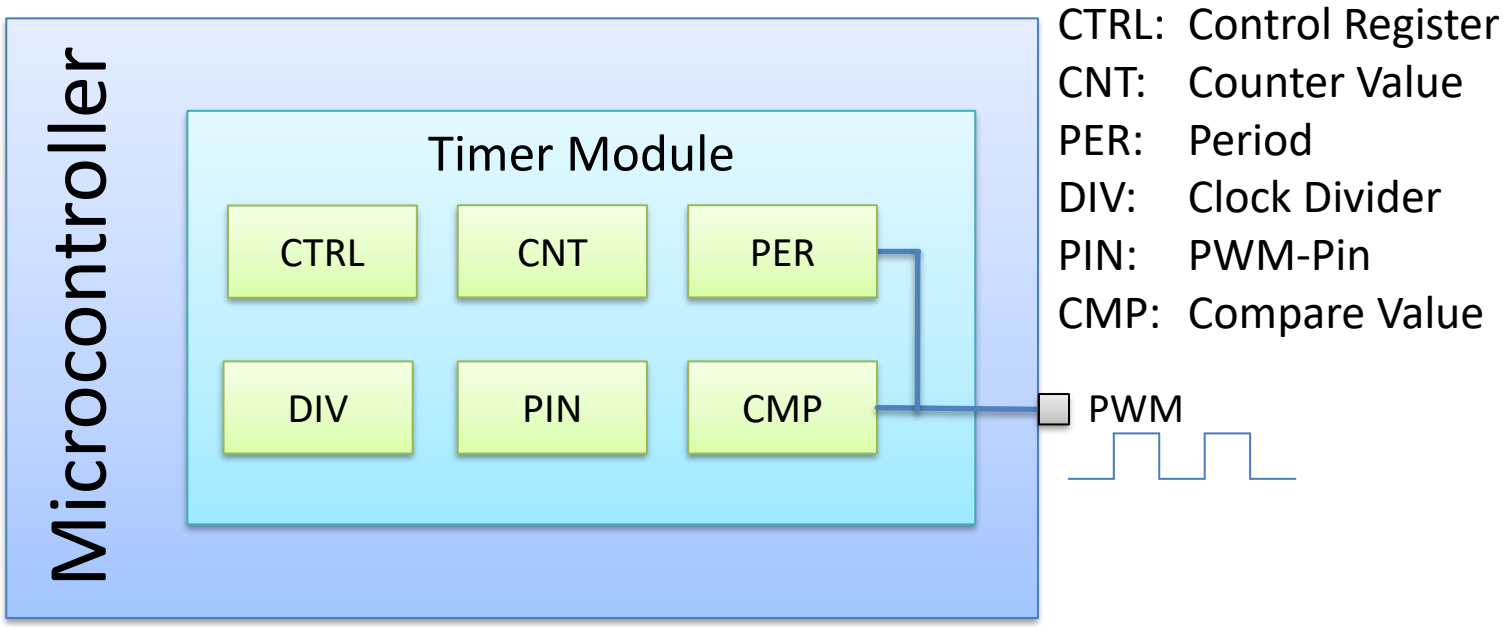
PWM (3)

- Architecture



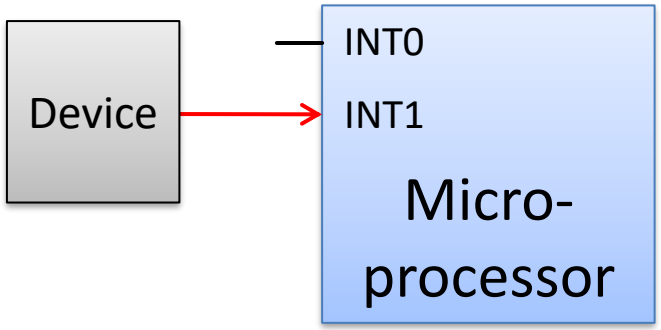
PWM (4)

- PWM Module



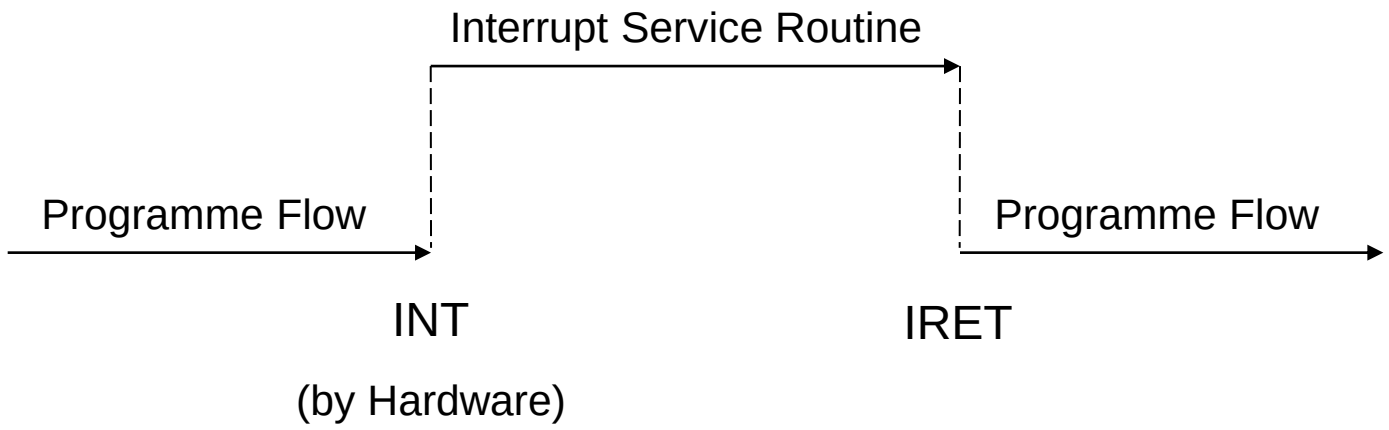
Interrupts (1)

- Generated by hardware
- Indicated by an interrupt line
- Occur unpredictably
- Tell that something happened
- Examples
  - Port interrupt
  - Timer interrupt
  - Data ready interrupt



Interrupts (2)

- Interrupt Processing
  - Execution of code is interrupted
  - Interrupt service route (ISR) is executed
  - Execution of original code is resumed.



Interrupts (3)

- Interrupt Controller
  - Handles Interrupts
    - Queueing
      - Which one is served first?
    - Nesting
      - Based on priority levels
      - Which interrupt interrupts other interrupts?
    - Forwarding
      - Notifies the CPU of an interrupt request (IRQ)