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

Timers and Interrupts

\[
f_{\text{CLK}} = \frac{f_{\text{SYS}}}{n}
\]

- \( f_{\text{SYS}} \): System Frequency
- \( f_{\text{CLK}} \): Clock Frequency
- \( n \): Prescaler Value
Timers (2)

• Operating mode

CNT: counter value, PER: period = TOP, CLK: clock ticks
Timers (3)

• Operating mode (continued)

— Example: TOP = 5

\[ T = (\text{TOP}+1) \cdot t_{\text{CLK}} \]

\( T \): timer period, \( t_{\text{CLK}} \): clock period
Timers (4)

• Timer Module

CTRL: Control Register
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 intermediate 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

CTRL: Control Register
CNT: Counter Value
PER: Period
CMP: Compare Value
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)

[Diagram of Interrupt Controller and Microprocessor with IRQ Vector]