Timing and Interrupts on the PIC

Today:

- Timer configuration
- Interrupt handling

Administrivia

- Due today:
  - HW 1 in class
  - Milestone 2 by 17:00
- Next Milestone: PWM control
  - October 24th
  - Possibilities include: LEDs or micro servo motors

Sensor of the Day...

Sensor of the Day...
Thermistor

Resistance varies as a function of temperature:

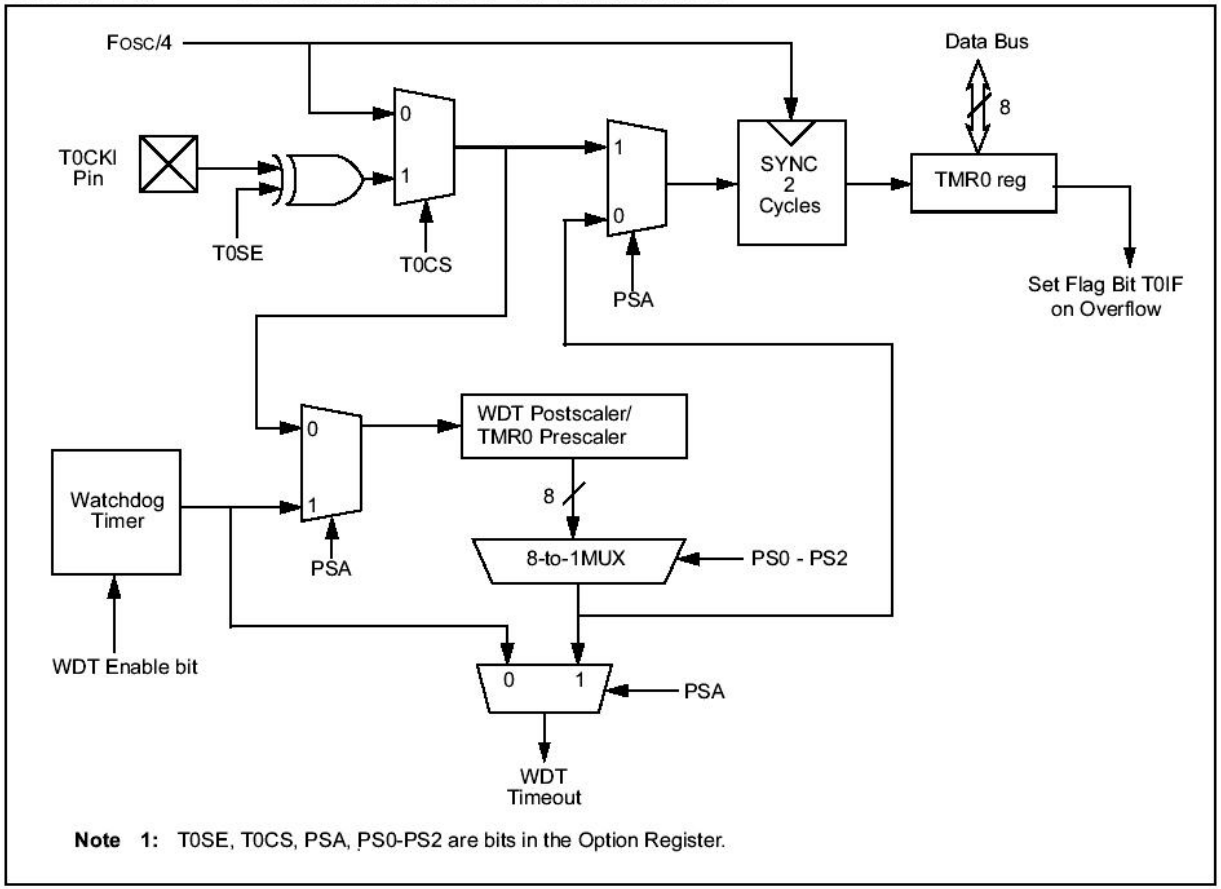
27.28 KΩ	0^{o} C
10.00 KΩ	25^{o} C

Timers

- Timer 0: 8-bit counter
  - Internally or externally driven
- Timer 1: 16-bit counter
  - Timing capture (e.g., of a PWM signal)
  - Timing comparator
- Timer 2: 8-bit counter
  - Flexible timing
  - Baud rate generation
  - Pulse-width modulation (PWM) generation

Timer 0

FIGURE 6-1: BLOCK DIAGRAM OF THE TIMER0/WDT



- TMR0 is an 8-bit counter
  - Can generate interrupt on overflow
- Clock source: instruction clock or external (T0CKI pin)
- TMR0 prescaler allows clock frequency to be divided by 2^i (where $1 <= i <= 8$)

Timer 0 Example

- Suppose:
  - TOCS = 0
  - PSA = 1
  - Clock frequency is 20MHz
- What is interrupt frequency?
- What is the interrupt period?

Timer 0 Example

- Suppose:
  - TOCS = 0
  - PSA = 1
  - Clock frequency is 20MHz
- What is interrupt frequency?
  - 20000000 / (4 * 256) = 19531.25 Hz = 19.531 KHz
- What is the period?
  - 1/freq = 0.0512 ms = 51.2 μsec

Timer 0 Example II

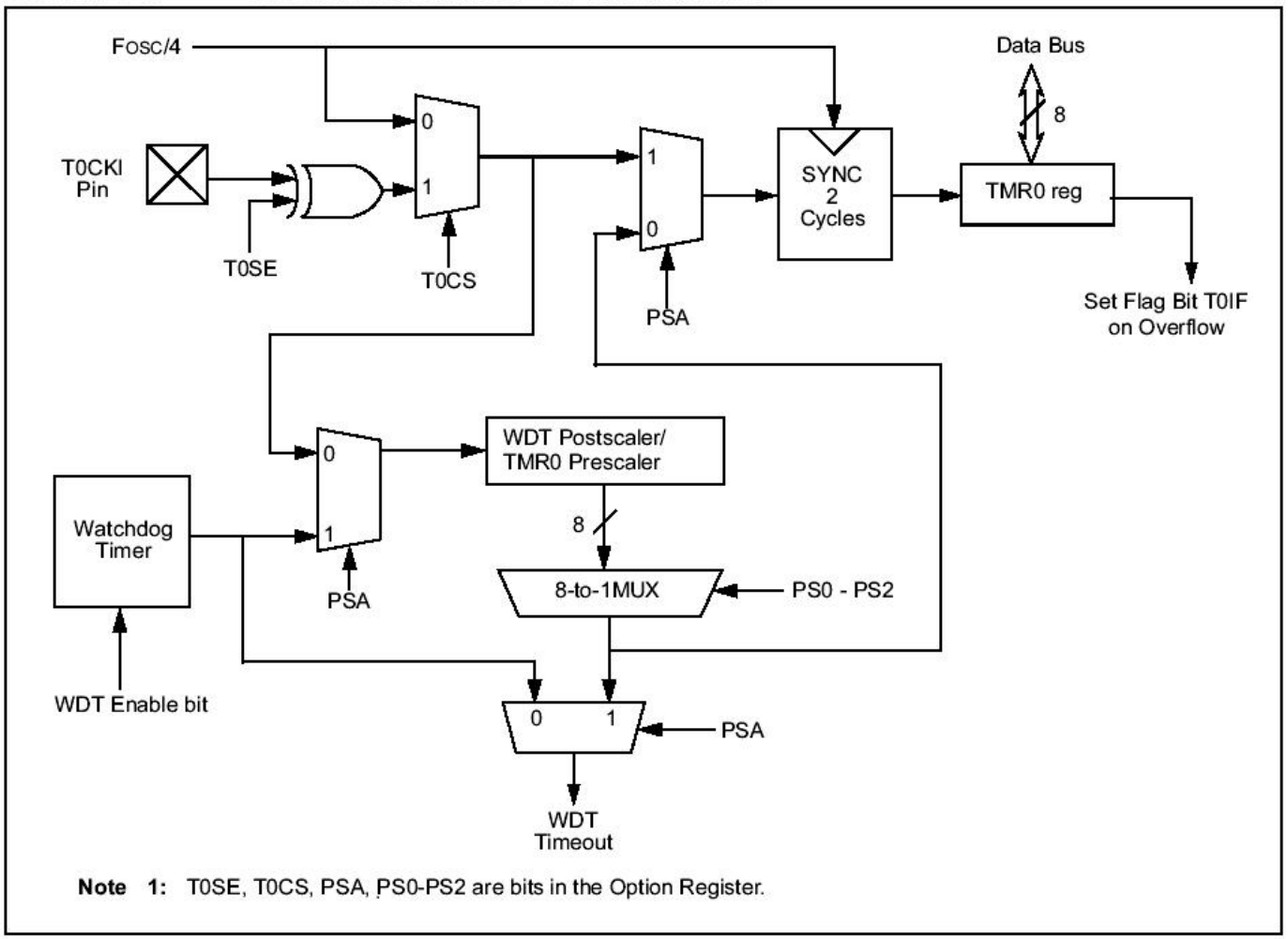
- Suppose:
  - TOCS = 0
  - PSA = 0
  - PS[2..0] = 0b100
  - Clock frequency is 20MHz
- What is interrupt frequency?
- What is the interrupt period?

Timer 0 Example II

- Suppose:
  - TOCS = 0
  - PSA = 1
  - Clock frequency is 20MHz
- What is interrupt frequency?
  - 20000000 / (4 * 256 * 32) = 610.35 Hz
- What is the period?
  - 1/freq = 1.638 ms

Timer 0

FIGURE 6-1: BLOCK DIAGRAM OF THE TIMER0/WDT



Note 1: T0SE, T0CS, PSA, PS0-PS2 are bits in the Option Register.

How do we count at a higher frequency than 19.31 kHz?

Timer 0

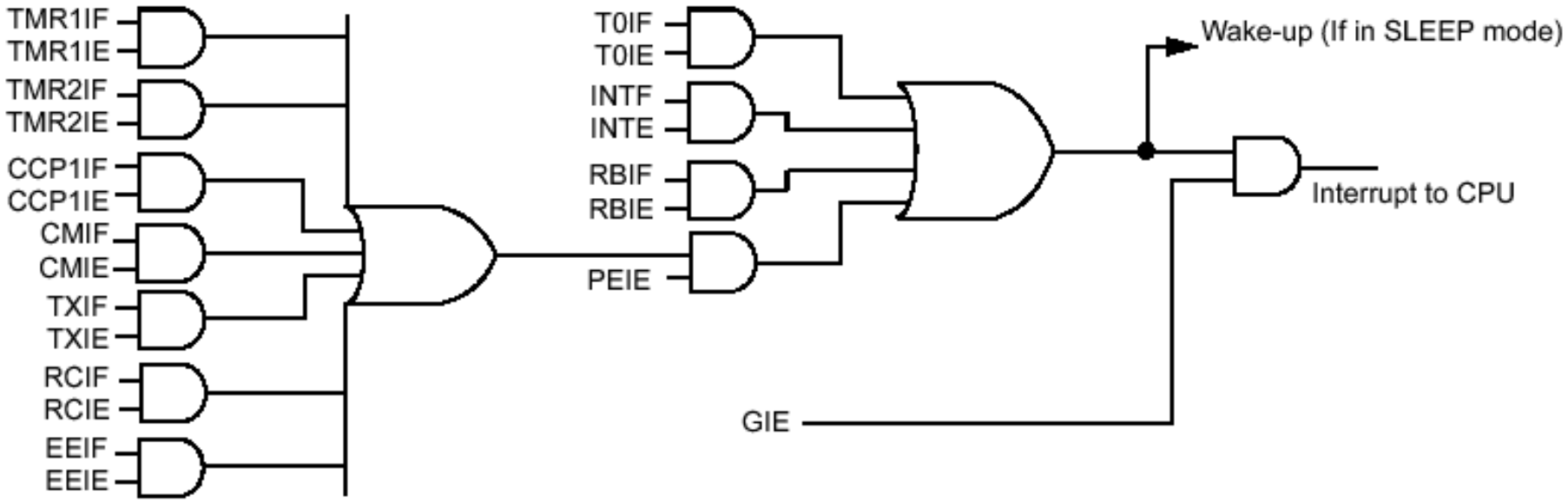
How do we count at a higher frequency than 19.31 KHz?

- Interrupt service routine must set TMR0 so that the timeout happens in fewer cycles
- Note that when this write happens, counting will be inhibited for 2 cycles

Prescaler Note

- The prescaler hardware for Timer 0 is shared with the watch-dog-timer
- But only one circuit may use it at any one time

Interrupt Processing System



- Global interrupt control (GIE)
- Individual interrupt enable lines

Interrupt Service Routines and CCSC

The CCSC library provides a default interrupt handling routine:

- Determines which interrupt has been raised
- Calls the specific interrupt service routine that you have defined
- Handles saving and restoring the working register and the zero bit

Specifying a Function as an Interrupt Handler

Use the corresponding preprocessor directive. For example:

```
#int_timer0
void my_interrupt_handler() {
     :
     :
}
```

Enabling the Interrupt

- Don't forget to configure the timer before enabling it...

```
main() {
      :
   setup_timer_0(RTCC_INTERNAL | RTCC_DIV_256);
   ENABLE_INTERRUPTS(INT_TIMER0);
   ENABLE_INTERRUPTS(GLOBAL);
      :
}
```

- RTCC_INTERNAL: use the internal clock
- RTCC_DIV_256: divide clock by 256

Global Interrupt Service Routine

- The CCSC-provided GISR may be replaced with your own:

```
#int_global
void my_global_interrupt_handler() {
     :
     :
}
```

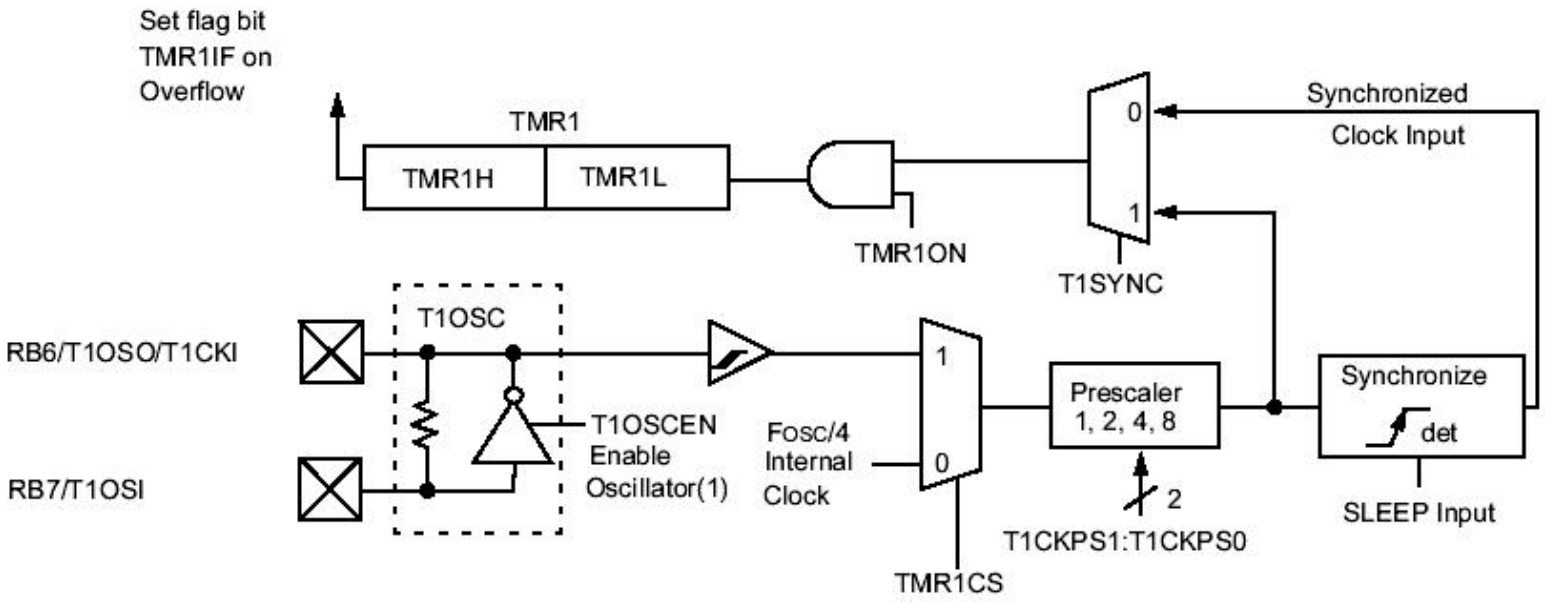
- You are responsible for saving registers

External Input to Timer 0

- Input from pin T0CKI
- mode = RTCC_EXT_L_TO_H: count on pin transitions from low to high
- mode = RTCC_EXT_H_TO_L: count on pin transitions from high to low

Timer 1

FIGURE 7-1: TIMER1 BLOCK DIAGRAM



Note 1: When the T1OSCEN bit is cleared, the inverter and feedback resistor are turned off. This eliminates power drain.

- TMR1: 16-bit timer
  - Generate an interrupt on overflow
- Clock source: instruction clock or RB7
- Prescaler: division of clock by 1, 2, 4, or 8 cycles

Timer 1 Notes

- Operation during sleep:
  - Can set up an RC-circuit across RB6/RB7
  - This oscillator can run up to 200 Khz
- Reading/Writing from TMR1/TMR0:
  - Must properly handle overflow between read or write operations

Timer 1 Example

- Suppose:
  - TMR1SC = 0
  - Prescaler = 2
- What is the interrupt frequency?
- What is the period?

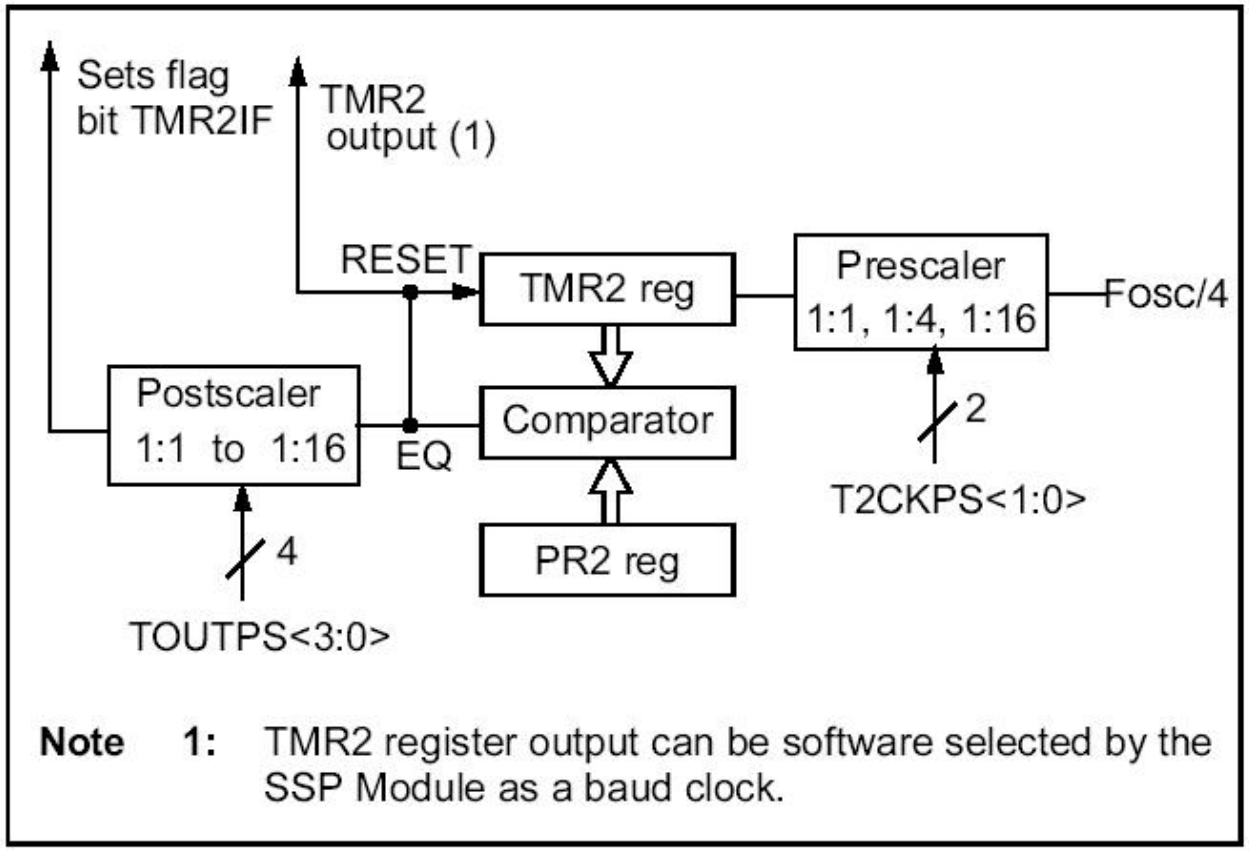
Timer 1 Example

- Suppose:
  - TMR1SC = 0
  - Prescaler = 2
- What is the interrupt frequency?
  - 20000000 / (4 * 2 * 256 * 256) = 38.15 Hz
- What is the period?
  - 1/freq = 26.2 ms

Timer 2

More flexible than other two

FIGURE 8-1: TIMER2 BLOCK DIAGRAM



- Input: instruction clock
- Prescaler: 1, 4, 16
- Reset at any time (determined by PR2)
- Postscaler: 1-16

Timer 2 Example

- Suppose:
  - Prescaler = 4
  - PR2 = 150
  - Postscaler = 7
- What is the interrupt frequency?
- What is the period?

Timer 2 Example

- Suppose:
  - Prescaler = 4
  - PR2 = 150
  - Postscaler = 7
- What is the interrupt frequency?
  - 20000000 / (4 * 4 * 150 * 7) = 1190.48 Hz
- What is the period?
  - 1/freq = .84 ms

Timer 2 Example II

How could we configure Timer 2 so that we have an interrupt period of 1 ms?

Timer 2 Example II

How could we configure Timer 2 so that we have an interrupt period of 1 ms?

- Prescaler = 4
- PR2 = 125
- Postscaler = 10

Next Time

- Handling critical sections
- PWM control
- Reading: DwPICM Chapters 4,5,6 (if you haven't already)
- Lab:
  - Wednesday, Friday, Monday
  - Exercise 3: RC circuits