

Lab 6 Notes

Tuesday, April 3, 2018 1:08 PM

Very important for Lab 7!

Interrupts

- 2 kinds: High & Low Priority

Low Priority Interrupt

- ROM Location = $0x0018$
- Can put any kind of event to be handled here
- Ex) If a switch or button is set high

High Priority Interrupt

- ROM Location = $0x0008$
- Used for more important events
- Can also interrupt a Low Priority Interrupt
- Ex) When clock is on and running, High Priority will reset the clock no matter what the state is.

Specific Pins are usually meant for Interrupts

- Pin 33 \rightarrow RB0/INT0
- Pin 34 \rightarrow RB1/INT1
- Pin 35 \rightarrow RB2/INT2
- Look at PIC18F452 Pinout in M Drive for more detail.

Interrupts in Code

- Lecture 11, slides 45-46 shows how to set up interrupt functions
- Note the keywords
- High Priority has to have "interrupt" in the name
- Low Priority has to have "interrupt low-priority" in the name

Timers

- PIC18F452 has 4 timers: Timer 0-3, but for this lab, we only care about Timer 0
- We need a Timer in this lab to handle the actual clock (MPLAB will provide the frequency for updating clock)
- Timer 0 can be set to 8 or 16-bit timer
- We want 16-bit so that we can load the proper preload values to give us as close as 1 sec delay

Calculating the X second delay & how to load calculation

- Lecture 12 slides 10-13

ex) Calculating a 1.2 second Delay

$d = 1.2$ (Delay time in seconds) (Time when you want Timer 0 registers to set the overflow flag) \leftarrow

$$F_{in} = 2.5 \text{ MHz}$$

$$X = d \times F_{in} = 1.2 \text{ s} \times 2.5 \text{ MHz} = 3,000,000 \text{ cycles/tick occurs in 1.2 sec}$$

16 bit Timer max is $2^{16} = 65,536$, so we need to scale 3,000,000 down under 65,536

- Use Prescalers, Timer 0 has many (1:2, 1:4, ... 1:256)

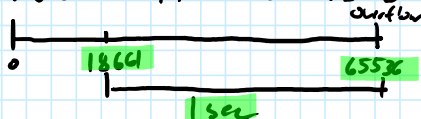
$$\begin{array}{lcl} \text{- } 1:2 & 3,000,000 / 2 = 1,500,000 & \leq 65,536 \quad \times \\ \text{- } 1:32 & 3,000,000 / 32 = 93,750 & \leq 65,536 \quad \times \\ \text{- } 1:64 & 3,000,000 / 64 = 46,875 & \leq 65,536 \quad \checkmark \end{array} \quad \begin{array}{l} \\ \\ \text{!!! Hurry!!!} \\ \text{Prescaler 1:64 got us under 65,536} \end{array}$$

You can say 46,875 ticks takes 1 sec

- Take the max value 65,536 & subtract by calculation

$$65,536 - 46,875 = 18,661$$

- Now are timer starts counting at 18,661 ticks/cycle, not zero



- To load this calculated value, we use 2 new SFR's TMR0H & TMR0L

- Similar to ADRESL & ADRESH

- Now we don't need to bit-shift or combine

- Take 18,661 & convert to hex

$$18,661 = 48E5$$

* Set load High First!

$$\text{- TMR0H} = 0x48$$

- Take 18.661 & Convert to Hex
18.661 = 48E5
- * Set Load High First!
- TMROH = 0x48
- TMROL = 0xE5

SFR's

We used ADCON, SSPSTAT, SSPCON... and now TOCON, INTCON1/2/3, RCON

TOCON: Timer 0 Configuration (8-bit) *Lecture 12, Slide 17*

- **TMRODN**
- Similar to ADCON bits. GO from ADC, keep it off until ready
- **TO8bit**
- We are using 16-bit timer
- **TOCS**
- Use internal instruction CLK0
- **TOSE**
- Doesn't matter
- **PSA**
- We are using a Prescaler
- **TOPS2 - TOPS0**
- Select Prescaler used to calculate preload values. In example, 1:64 prescaler used.

INTCON: Interrupt Configuration *Lecture 11, Slide 30*

- **GIE**
- Disable for now, start later
- **PEIE**
- Disable for now, start later
- **TMROIE**
- We need to enable overflow so the timer 0 registers can set overflow flag at every 1 sec
- **INT0IE**
- We need to enable external interrupt for interrupt 0, where the clear/reset button will be handled
- **RBIE**
- Disable RB port change interrupt, not using it
- **TMROIF**
- Initialize flag to zero, Needs to be cleared when flag is raised later on.
- **INT0IF**
- Initialize flag to zero, Needs to be cleared when flag is raised later on.
- **RBIF**
- Again, Not using Port B Change Interrupts

INTCON2: Interrupt Configuration 2 *Lecture 11 Slide 31*

- **RBP0**
- Disable all Port B Pullups
- **INTEDG0 - 62**
- Interrupt on Rising Edge (low-to-high)
- **TMROIP**
- Timer 0 is low priority for this lab
- **RBIP**
- Don't Care

INTCON3: Interrupt Configuration 3 *Lecture 11 Slide 32*

- **INT2IP**
- Interrupt 2 not used, don't care
- **INT1IP**
- Interrupt 1 is low priority
- **INT2IF**
- Disable, Interrupt 2 not used
- **INT1IF**
- Enable external interrupt for interrupt 1, the play/pause switch
- **INT2IF**
- Don't care, not used
- **INT1IF**
- Initialize flag to zero, needs to be cleared when flag is raised later on

RCON: Reset Control *Lecture 11, Slide 39*

- **IDEN**
- We want interrupts to have priority levels

Check List

- ✓ TRIS settings
- ✓ Timer 0 settings
- ✓ INTCONX settings
- ✓ RCON settings
- ✓ Load preload values
- ✓ Turn On all high/low priority interrupts (GIE & PEIE), and then turn on Timer 0
- If you clear/reset, you must reload preload values (TMROH/L)
- If Timer 0 overflows, clear flag & reload preload values (TMROH/L) } Remember! Load High First!

24 Hour Clock

00:00:00 → 23:59:59 → 24:00:00 → 00:00:00

LED's when clearing/reset

Start off as → ● ● ●
Every clear/reset → ● ● ● → ● ● ● → ● ● ● → ● ● ●