EE 2004
Week 10 Homework
Solution

1. Find the locations of the flag bit, enable bit and priority bit for the following interrupt sources (i.e., Which SFR are they located? Which bit?). Please refer to the Appendix for SFRs involved in interrupts.

    a. Timer 0
    b. Timer 1
    c. Timer 2

	Flag bit (Register)	*Enable bit (Register)*	*Priority Bit (Register)*
Timer0	*TMR0IF (INTCON)*	*TMR0IE (INTCON)*	*TMR0IE (INTCON2)*
Timer1	*TMR1IF (PIR1)*	*TMR1IE (PIE1)*	*TMR1IP (IPR1)*
Timer2	*TMR2IF (PIR1)*	*TMR2IE (PIE1)*	*TMR2IP (IPR1)*

2. Show the instructions needed to enable the Timer0 interrupt in the following situations:
    a. Interrupt priority is disabled.
    b. Interrupt priority is enabled, and Timer0 interrupt is a high-priority interrupt.
    c. Interrupt priority is enabled, and Timer0 interrupt is a low-priority interrupt.

```
a. bcf  INTCON, GIE
   bsf  INTCON, TMR0IE
b. bsf  INTCON, GIEH
   bsf  INTCON, TMR0IE
c. bsf  INTCON, GIEH
   bsf  INTCON, GIEL
   bsf  INTCON, TMR0IE
```

3. Where the PC will be redirected if a high-priority interrupt is triggered? If we want to put the entire interrupt service routine (ISR) in the interrupt vector for a high priority interrupt source, what is the maximum size of this ISR in bytes? What would we do if the size of the ISR is larger?

    0x000008.

    Because the 0x000018 is assigned to low-priority interrupts. If the entire ISR were fitted into this location, it has to be stored within location 0x000008-17. Therefore, the maximum size is 16 bytes.

    Use a goto or bra instruction to redirect the PC to the subroutine written to handle the interrupt.

4. Why do we put a goto instruction at program memory address 0?

    Because 0x000008 is assigned to high-priority interrupts, no part of the main program should be written on beyond 0x000006. Therefore, at restart, we should redirect the PC to a separate location where the main program is implemented.

5. With a single instruction, disable all the interrupts.

   All interrupts would be disabled if the global interrupt enable bit is disabled.

```
bcf INTCON, GIE
```

6. Which bits are required to be enabled if we want to enable Timer1 interrupt as a low-priority interrupt?

```
bsf   RCON, IPEN;   enable priority interrupt
bsf   PIE1, TMR1IE; enable TMR1 overflow interrupt
bcf   IPR1, TMR1IP; set TMR1 interrupt to low priority
bsf   INTCON, GIEH; enable high priority interrupt
bsf   INTCON, GIEL; enable low priority interrupt
```

7. Explain what happens if Interrupt B is activated while the PIC18 is serving Interrupt A (i.e., while the the ISR associated with Interrupt A is being executed) under the following five situations:
   a. Interrupt priority is disabled.
   b. Interrupt priority is enabled. Interrupt A&B are both high-priority interrupts.
   c. Interrupt priority is enabled. Interrupt A&B are both low-priority interrupts.
   d. Interrupt priority is enabled. Interrupt A is a high-priority interrupt, whereas B is a low-priority interrupt.
   e. Interrupt priority is enabled. Interrupt A is a low-priority interrupt, whereas B is a high-priority interrupt.

   a. When Interrupt A is being served, the GIE bit is disabled. Therefore when serving Interrupt A, Interrupt B would not be served. (See pp. 16 of Chap 6 for details)

   (For b-e, see pp. 17 of Chap 6 for details)
   b. When Interrupt A is being served, the GIEH bit is disabled, which disables all interrupts. Therefore when serving Interrupt A, Interrupt B would not be served.
   c. When Interrupt A is being served, the GIEL bit is disabled, which disables all low-priority interrupts. Therefore when serving Interrupt A, Interrupt B would not be served.
   d. When Interrupt A is being served, the GIEH bit is disabled, which disables all interrupts. Therefore when serving Interrupt A, Interrupt B would not be served.
   e. When Interrupt A is being served, the GIEL bit is disabled, which disables all low-priority interrupts. However, GIEH still stays at 1. Therefore as a high-priority interrupt, Interrupt B will be served immediately even if the ISR associated with Interrupt A is currently being executed.

8. Explain why we cannot use `return` instead of `retfie` as the last instruction of an ISR.

The global interrupt enable (GIE) flag has been disabled when an interrupt is evoked to lock out further interrupt requests.

After an ISR has been completed, `retfie` *re-enables GIE, so that other interrupts could be served.* `return` *would not re-enable GIE.*

Appendix: Interrupt Registers

INTCON

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-x
GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF(1)
bit 7							bit 0

bit 7 **GIE/GIEH:** Global Interrupt Enable bit

When IPEN = 0:
1 = Enables all unmasked interrupts
0 = Disables all interrupts

When IPEN = 1:
1 = Enables all high-priority interrupts
0 = Disables all interrupts

bit 6 **PEIE/GIEL:** Peripheral Interrupt Enable bit

When IPEN = 0:
1 = Enables all unmasked peripheral interrupts
0 = Disables all peripheral interrupts

When IPEN = 1:
1 = Enables all low-priority peripheral interrupts
0 = Disables all low-priority peripheral interrupts

bit 5 **TMR0IE:** TMR0 Overflow Interrupt Enable bit
1 = Enables the TMR0 overflow interrupt
0 = Disables the TMR0 overflow interrupt

bit 4 **INT0IE:** INT0 External Interrupt Enable bit
1 = Enables the INT0 external interrupt
0 = Disables the INT0 external interrupt

bit 3 **RBIE:** RB Port Change Interrupt Enable bit
1 = Enables the RB port change interrupt
0 = Disables the RB port change interrupt

bit 2 **TMR0IF:** TMR0 Overflow Interrupt Flag bit
1 = TMR0 register has overflowed (must be cleared in software)
0 = TMR0 register did not overflow

bit 1 **INT0IF:** INT0 External Interrupt Flag bit
1 = The INT0 external interrupt occurred (must be cleared in software)
0 = The INT0 external interrupt did not occur

bit 0 **RBIF:** RB Port Change Interrupt Flag bit(1)
1 = At least one of the RB<7:4> pins changed state (must be cleared in software)
0 = None of the RB<7:4> pins have changed state

INTCON2

R/W-1	R/W-1	R/W-1	R/W-1	U-0	R/W-1	U-0	R/W-1
$\overline{\text{RBPU}}$	INTEDG0	INTEDG1	INTEDG2	—	TMR0IP	—	RBIP
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **$\overline{\text{RBPU}}$:** PORTB Pull-up Enable bit
1 = All PORTB pull-ups are disabled
0 = PORTB pull-ups are enabled by individual port latch values

bit 6 **INTEDG0:** External Interrupt 0 Edge Select bit
1 = Interrupt on rising edge
0 = Interrupt on falling edge

bit 5 **INTEDG1:** External Interrupt 1 Edge Select bit
1 = Interrupt on rising edge
0 = Interrupt on falling edge

bit 4 **INTEDG2:** External Interrupt 2 Edge Select bit
1 = Interrupt on rising edge
0 = Interrupt on falling edge

bit 3 **Unimplemented:** Read as '0'

bit 2 **TMR0IP:** TMR0 Overflow Interrupt Priority bit
1 = High priority
0 = Low priority

bit 1 **Unimplemented:** Read as '0'

bit 0 **RBIP:** RB Port Change Interrupt Priority bit
1 = High priority
0 = Low priority

RCON

R/W-0	R/W-1(1)	U-0	R/W-1	R-1	R-1	R/W-0(1)	R/W-0
IPEN	SBOREN	—	$\overline{RI}$	$\overline{TO}$	$\overline{PD}$	$\overline{POR}$	$\overline{BOR}$
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **IPEN:** Interrupt Priority Enable bit
1 = Enable priority levels on interrupts
0 = Disable priority levels on interrupts (PIC16CXXX Compatibility mode)

bit 6 **SBOREN:** Software BOR Enable bit(1)
For details of bit operation, see Register 4-1.

bit 5 **Unimplemented:** Read as '0'

bit 4 **$\overline{RI}$:** RESET Instruction Flag bit
For details of bit operation, see Register 4-1.

bit 3 **$\overline{TO}$:** Watchdog Timer Time-out Flag bit
For details of bit operation, see Register 4-1.

bit 2 **$\overline{PD}$:** Power-Down Detection Flag bit
For details of bit operation, see Register 4-1.

bit 1 **$\overline{POR}$:** Power-on Reset Status bit(1)
For details of bit operation, see Register 4-1.

bit 0 **$\overline{BOR}$:** Brown-out Reset Status bit
For details of bit operation, see Register 4-1.

PIR1

R/W-0	R/W-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
PSPIF(1)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF
bit 7							bit 0

bit 7 **PSPIF:** Parallel Slave Port Read/Write Interrupt Flag bit(1)
1 = A read or a write operation has taken place (must be cleared in software)
0 = No read or write has occurred

bit 6 **ADIF:** A/D Converter Interrupt Flag bit
1 = An A/D conversion completed (must be cleared in software)
0 = The A/D conversion is not complete

bit 5 **RCIF:** EUSART Receive Interrupt Flag bit
1 = The EUSART receive buffer, RCREG, is full (cleared when RCREG is read)
0 = The EUSART receive buffer is empty

bit 4 **TXIF:** EUSART Transmit Interrupt Flag bit
1 = The EUSART transmit buffer, TXREG, is empty (cleared when TXREG is written)
0 = The EUSART transmit buffer is full

bit 3 **SSPIF:** Master Synchronous Serial Port Interrupt Flag bit
1 = The transmission/reception is complete (must be cleared in software)
0 = Waiting to transmit/receive

bit 2 **CCP1IF:** CCP1 Interrupt Flag bit
Capture mode:
1 = A TMR1 register capture occurred (must be cleared in software)
0 = No TMR1 register capture occurred
Compare mode:
1 = A TMR1 register compare match occurred (must be cleared in software)
0 = No TMR1 register compare match occurred
PWM mode:
Unused in this mode.

bit 1 **TMR2IF:** TMR2 to PR2 Match Interrupt Flag bit
1 = TMR2 to PR2 match occurred (must be cleared in software)
0 = No TMR2 to PR2 match occurred

bit 0 **TMR1IF:** TMR1 Overflow Interrupt Flag bit
1 = TMR1 register overflowed (must be cleared in software)
0 = TMR1 register did not overflow

PIE1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PSPIE(1)	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **PSPIE:** Parallel Slave Port Read/Write Interrupt Enable bit(1)
1 = Enables the PSP read/write interrupt
0 = Disables the PSP read/write interrupt

bit 6 **ADIE:** A/D Converter Interrupt Enable bit
1 = Enables the A/D interrupt
0 = Disables the A/D interrupt

bit 5 **RCIE:** EUSART Receive Interrupt Enable bit
1 = Enables the EUSART receive interrupt
0 = Disables the EUSART receive interrupt

bit 4 **TXIE:** EUSART Transmit Interrupt Enable bit
1 = Enables the EUSART transmit interrupt
0 = Disables the EUSART transmit interrupt

bit 3 **SSPIE:** Master Synchronous Serial Port Interrupt Enable bit
1 = Enables the MSSP interrupt
0 = Disables the MSSP interrupt

bit 2 **CCP1IE:** CCP1 Interrupt Enable bit
1 = Enables the CCP1 interrupt
0 = Disables the CCP1 interrupt

bit 1 **TMR2IE:** TMR2 to PR2 Match Interrupt Enable bit
1 = Enables the TMR2 to PR2 match interrupt
0 = Disables the TMR2 to PR2 match interrupt

bit 0 **TMR1IE:** TMR1 Overflow Interrupt Enable bit
1 = Enables the TMR1 overflow interrupt
0 = Disables the TMR1 overflow interrupt

IPR1

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
PSPIP(1)	ADIP	RCIP	TXIP	SSPIP	CCP1IP	TMR2IP	TMR1IP
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **PSPIP:** Parallel Slave Port Read/Write Interrupt Priority bit(1)
1 = High priority
0 = Low priority

bit 6 **ADIP:** A/D Converter Interrupt Priority bit
1 = High priority
0 = Low priority

bit 5 **RCIP:** EUSART Receive Interrupt Priority bit
1 = High priority
0 = Low priority

bit 4 **TXIP:** EUSART Transmit Interrupt Priority bit
1 = High priority
0 = Low priority

bit 3 **SSPIP:** Master Synchronous Serial Port Interrupt Priority bit
1 = High priority
0 = Low priority

bit 2 **CCP1IP:** CCP1 Interrupt Priority bit
1 = High priority
0 = Low priority

bit 1 **TMR2IP:** TMR2 to PR2 Match Interrupt Priority bit
1 = High priority
0 = Low priority

bit 0 **TMR1IP:** TMR1 Overflow Interrupt Priority bit
1 = High priority
0 = Low priority