

**ECE 425 Introductions to Microprocessors
Laboratory Work 10**

Objective:

- 1) Interrupts.
- 2) Sleep Command.
- 3) WatchDog Timer.

Preparation:

Option register and interrupt control register are used to program interrupt resources and their register bits are shown in the figure below.

OPTION REGISTER

RBPU	INTEDG	TOCS	TOSE	PSA	PS2	PS1	PS0
------	--------	------	------	-----	-----	-----	-----

INTCON REGISTER

GIE	EEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF
-----	------	------	------	------	------	------	------

Bit-4 of the STATUS register (\overline{TO}) has value '1' on reset or when WDT is cleared. It has value 0 when WDT overflows. It takes 18msec for WDT to overflow (without prescalar). When WDT overflows processor is resetted.

1) In the program below WDT is enabled. Set your oscillator frequency to 4KHz in MPLAB and animate the following program. Comment on the results.

```
LIST P=16F84A
INCLUDE "P16f84A.INC"

__config _CP_OFF&_WDT_ON&_XT_OSC; WDT is enabled.

start
    bsf    STATUS, RP0;
    bsf    OPTION_REG, PSA;
    bcf    OPTION_REG, PS2;
    bcf    OPTION_REG, PS1;
    bcf    OPTION_REG, PS0;
    bcf    STATUS, RP0;

    nop;
    nop;
    nop;
    nop;
loop
    nop;
    nop;
    nop;
    nop;
goto loop;

end
```

2) In the program below WDT is enabled. Set your oscillator frequency to 4KHz in MPLAB and animate the following program. Comment on the results.

```
LIST P=16F84A
INCLUDE "P16f84A.INC"

__config __CP_OFF&_WDT_ON&_XT_OSC; WDT is enabled.

start

        bsf    STATUS, RP0;
        bsf    OPTION_REG, PSA;
        bcf    OPTION_REG, PS2;
        bcf    OPTION_REG, PS1;
        bsf    OPTION_REG, PS0;
        bcf    STATUS, RP0;

        nop;
        nop;
        nop;
        nop;
loop
        nop;
        nop;
        nop;
        nop;
goto loop;

end
```

3) In the program below WDT is enabled. Set your oscillator frequency to 4KHz in MPLAB and animate the following program. Comment on the results.

```
LIST P=16F84A
INCLUDE "P16f84A.INC"

__config __CP_OFF&_WDT_ON&_XT_OSC; WDT is enabled.

start

        bsf    STATUS, RP0;
        bsf    OPTION_REG, PSA;
        bcf    OPTION_REG, PS2;
        bcf    OPTION_REG, PS1;
        bcf    OPTION_REG, PS0;
        bcf    STATUS, RP0;

        nop;
        nop;
        nop;
        nop;
loop
        nop;
        sleep;
        nop;
        nop;
        nop;
goto loop;

end
```

4) In the program below RB0/INT is enabled. Animate the program using stimulus, and send interrupt signals from RB0/INT and comment on the results.

```
LIST P=16F84A
INCLUDE "P16f84A.INC"

__config __CP_OFF&__WDT_OFF&__XT_OSC

    org 0x00;
    goto start
    org 0x04;
    goto ISR_INTE;

start
    bsf    STATUS, RP0;
    bsf    TRISB,0;
    bsf    OPTION_REG, INTEDG;
    bcf    STATUS, RP0;

    bsf    INTCON, GIE;
    bsf    INTCON, INTE;
    bcf    INTCON, INTF;

loop
    nop;
    nop;
    sleep;
    nop;
    nop;
    nop;

goto loop;

ISR_INTE
    bcf    INTCON, INTF;
    nop;
    nop;
    nop;
    retfie;

end
```

Laboratory Work:

- 1) Trace program segments in preparation 1-4 and comment on the results. What happens STATUS<bit-4> and STATUS<bit-5>.
- 2) Write a program that contains interrupt subroutine for RB4-RB7 interrupts. In your program your processor sleeps and when an interrupt is received from RB4-RB7 it wakes up and toggles RB0 and goes to sleep mode again.
- 3) Write the following program. Your processor wakes up every 72msecs and scans PORTA for any input. If an input is available, it activates the alarm device at RB0.

During your LAB work show every step that you complete to the LAB assistant. Get a copy of assembly files you write during the LAB hour via a flash disk for future reference.