ENGR 270 LAB #6 – Sensor and A/D Convertor

Objective
Utilize EDbot Micro and A/D convertor to sense obstacals and follow objects.

Related Principles
- Computer Organization and Design
- Microprocessors
- Hardware and Software Interface
- Digital Design
- Assembly language

Equipment
- Windows-based PC with MPLAB Simulation Solutions Software
- USB hard disk or other removable drives
- Microchip PICKit programmer
- EDbot Micro V11 Platform

Preparation/Background
EDbot Micro includes Infrared (IR) transmitters and Receivers, which can be used to detect object in front of the EDbot. The transimmter broadcasts an infrared When you give the infared transmitter power, it sends out infared waves directly out from the transmitter. As these rays reflect off objects, they return back towards the EDbot's IR receivers. When the receiver receives enough infrared waves, it acts like a short, causing RA0 to go to zero.

The following sample code uses RGB LED to indicate how close the front of EDbot Micro is to the obstacle. The colors change from green to blue to red to white as the front of EDbot Micro gets closer to an obstacle. Note that IR sensor is more effective in detecting surfaces with lighter colors.

```
;-------------------------------------------------------------------------------
; File: Lab6Exp1.asm
; Desc: A to D convertor (ADC) and IR sensor usage example.
;       RGB LED color change as the IR sensors get closer to obstacle.
;
; Last Update: August, 2019
; Auth: Class
;-------------------------------------------------------------------------------
   list    p=18F1220           ;processor type
   radix   hex                 ;default radix for data
   ; Disable Watchdog timer, Low V. Prog, and RA6 as I/O
   config  WDT=OFF,LVP=OFF,OSC=INTIO2

   #include    p18F1220.inc                    ;header file

   #define     dCount          0x80
   #define     dCountInner 0x81
   #define     IRdata          0x82    ; last IR input read

   org             0x000                           ; Executes after rest
StartL:
   ; initialize all I/O ports
   CLRF    IRdata          ; Clear IR reading
   CLRF    PORTA           ; Initialize PORTA
   CLRF    PORTB           ; Initialize PORTB
   MOVLW   0x7E
   MOVWF   ADCON1      ; Configure PortA<1:7> Digital and PortA<0> Analog
   MOVLW   0x35
   MOVWF   TRISA       ; Set Port A direction Per EDbot Micro Spec.
   MOVLW   0xC3
   MOVWF   TRISB       ; Set Port A direction Per EDbot Micro Spec.
   MOVLW   0x60
   IORWF   OSCCON      ; Set internal System Clock to 4 Mhz
```

```
   ; Wait until INT0 Button is pressed (include SW Debounce)
   Call    Int0Press

MainL:                                          ; Main loop
   CALL    IRread
   MOVLW   0xC0
   CPFSLT  IRdata
   BRA             Mgreen
   MOVLW   0x80
   CPFSLT  IRdata
   BRA             Mblue
   MOVLW   0x40
   CPFSLT  IRdata
   BRA             Mred
   BSF             PORTA,3  ;Green
   BSF             PORTB,5  ;Blue
   BSF             PORTB,2  ;Red
   BRA             MainL
Mgreen:
   BSF             PORTA,3  ;Green
   BCF             PORTB,5  ;Blue
   BCF             PORTB,2  ;Red
   BRA             MainL
Mblue:
   BCF             PORTA,3  ;Green
   BSF             PORTB,5  ;Blue
   BCF             PORTB,2  ;Red
   BRA             MainL
Mred:
   BCF             PORTA,3  ;Green
   BCF             PORTB,5  ;Blue
   BSF             PORTB,2  ;Red
   BRA             MainL

; Returns the IR reading in IRdata
IRread:
   BSF             PORTA,1  ; Turn on IR Transmitter
   BCF             ADCON2,7                    ; Left Adjusted reading
   MOVLW   0x03
   MOVWF   ADCON0       ; Turn on ADC
WaitNdone:
   BTFSC   ADCON0,1
   BRA          WaitNdone   ; wait until AD conversion has completed
   MOVFF     ADRESH,IRdata
   BCF             PORTA,1  ; Turn off IR Transmitter
   RETURN

; Delay function waits for (Wreg/10) seconds before returning
Delay:
   MOVWF   dCount
DelayLoop:
   CALL    DelayOnce
   DECF    dCount
   BNZ             DelayLoop
   RETURN
DelayOnce:
   CLRF    dCountInner      ; Internal delay loop
DelayOnceLoop:
   NOP
   INCF    dCountInner
   BNZ             DelayOnceLoop
   RETURN  ; Delay Function

; Wait until INT0 Button is pressed (include SW Debounce)
Int0Press:
   MOVLW   .1
```

```
    CALL    Delay
    MOVF    PORTB,0
    ANDLW   0x01
    BZ              Int0Press ; wait for button to be released
Int0PressZ:
    MOVLW   .1
    CALL    Delay
    MOVF    PORTB,0
    ANDLW   0x01
    BNZ             Int0PressZ ; wait for buton to be pressed
    RETURN  ; Int0Press Function

    end             ; end program
```

Experiment

Write EDbot Micro Assembly code to enable the EDbot Micro to follow a white Target (for example a sheet of white paper) without hitting the Target. Your code is expected the handel the following cases:

- EDbot circles in place while waiting to detect the Target
- EDbot approaches the Target as close as possible without hitting the target
- EDbot moves back if target is moved too close

This experiment requires that you review your high level design (flow chart or pseudo code) and demonstrate your system to the instructor upon completion. Include the approval signature in your report.

Team Members: • • • •	LAB6 Demo Instructor Approval Signature & Date:

Report Requirements

All reports must be computer printed (formulas and diagrams may be hand drawn) and at minimum include:

For each experiment:

a) Clear problem statement; specify items given and to be found.
b) Specific responses to each question asked in the experiment.
c) Documentation of resulting high level design, disassembled code, system diagram, schematics and any other supporting material.

For the report as a whole

a) Cover sheet with your name, course, lab title, date of completion and your teammates' name.
b) Lessons learned from this lab.
c) A new experiment and expected results which provide additional opportunity to practice the concepts in this lab.