ENGR 270 LAB #3 – EDbot Introduction

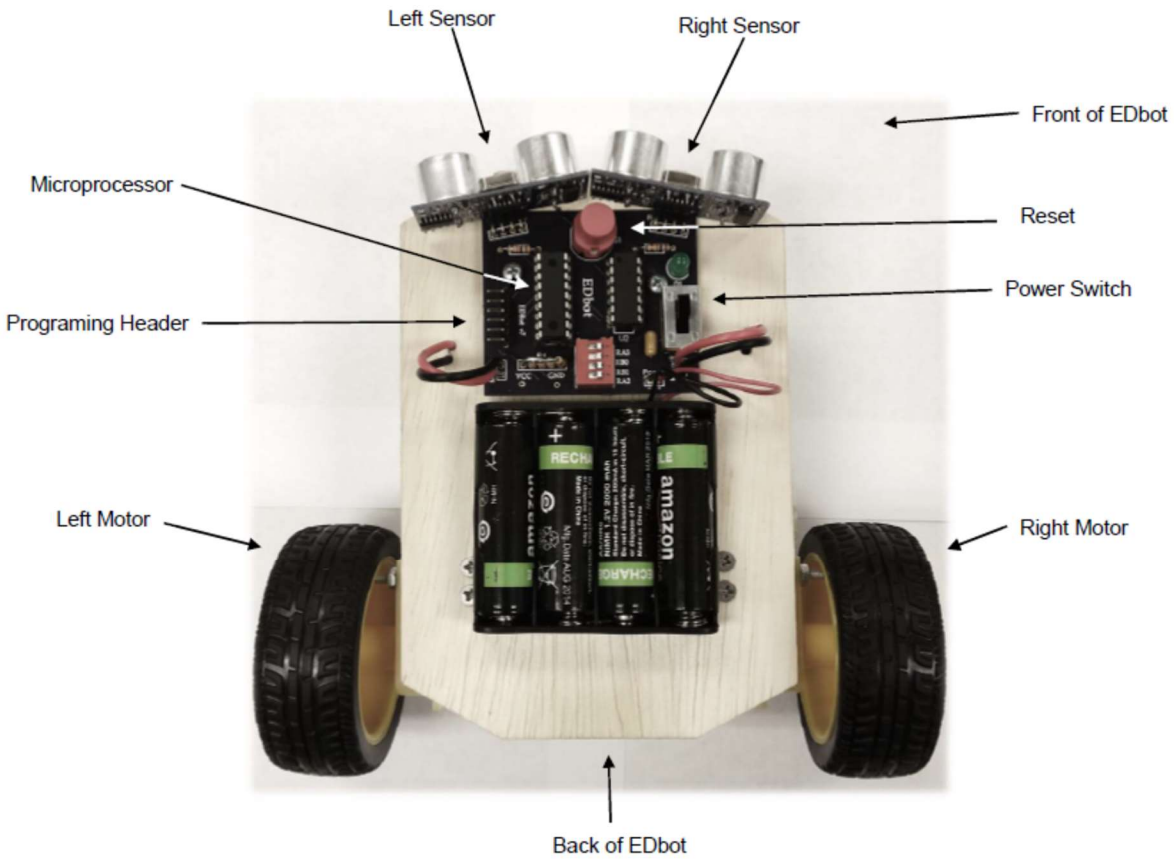
Objective
Introduce the EDbot platform and use of broader range of assembly instructions and constructs..

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

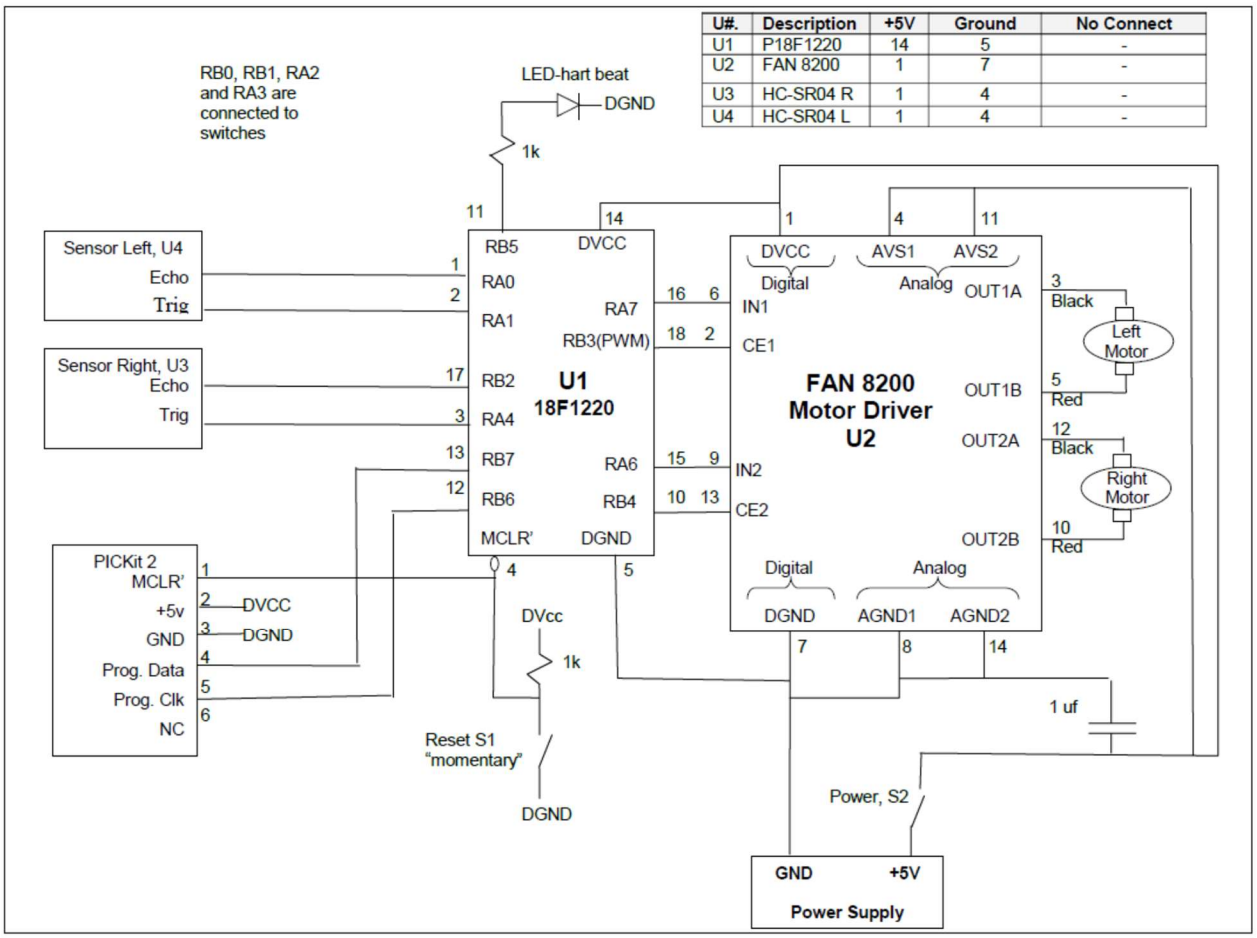
Equipment
- Windows-based PC with MPLAB Simulation Solutions Software
- USB hard disk or other removable drives
- Microchip PICKit programmer
- EDbot V7.0 Platform

Supplies
- None

Preparation/Background
EDbot was designed and implement by past students based on the learning from this course. The design is similar to the work done in labs 1 and 2. The following diagram outlines the physical design and labels major components of EDbot:



EDbot uses PIC18F1220 as the microcontroller with two independent DC motors and two independent distance sensors (HC-SR04) which make EDbot a highly flexible robotic platform. EDbot V7.0 schematic follows:



EDbot has specific assignment for all the PICmicro I/O Pins as follows:

Registers <bit #>	Pin # - Name -Type	Function
Port A <0>	1 – RA0 – Input	Echo Left Sensor
Port A <1>	2 – RA1 – Output	Trigger Left sensor
Port B <2>	17 – RB2– Input	Echo Right Sensor
Port A <4>	3 – RA4 – Output	Trigger Right sensor
Port A <7>	16 – RA7 – Output	Left Motor Director
Port B <3>	18 – RB3 – Output	Left Motor Enable (PWM capable)
Port A <6>	15 – RA6 – Output	Right Motor Director
Port B <4>	10– RB4 – Output	Right Motor Enable
PortB <0,1>	8,9 – RB 0,1 – Input	DIP Switch #3,2 (INT0 and INT1)
Port A <2,3>	6,7 – RA 2,3 – Inputt	DIP Switch #1,4
Port A <5>	4 – MCLR – Output	Reset – Red push button
Port B <5>	11 - RB5 - Output	D_1 LED

Experiment #1

Create a new MPLAB project using the code provided on the next page. Build the project and program EDbot. Write a summary of EDbot operation based on your review of the code, schematics and observation of EDbot executing the code.

Notes:

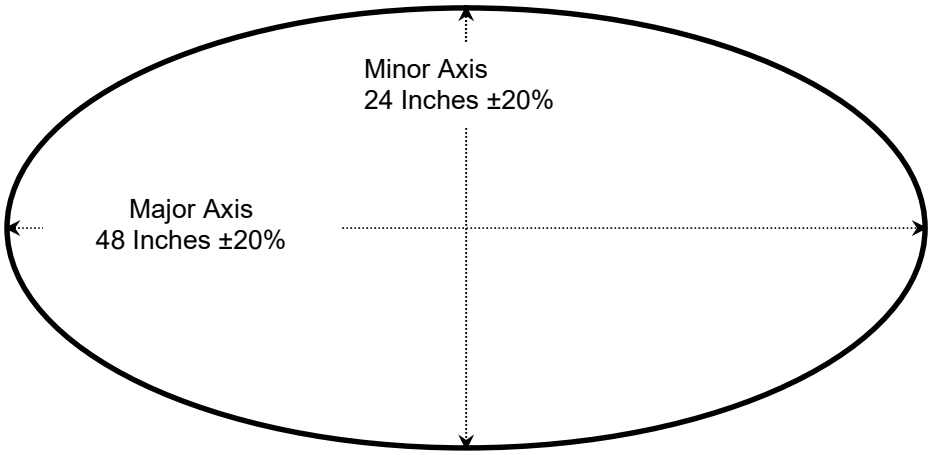
1) During the Programming and execution EDbot wheel will rotate. Be careful not to drop it!

2) Microchip header file supplied with MPLAB® IDE contains the definition for all the SFR register addresses and bit names in addition to commonly used constant values listed in the course text appendix. You can use SFR register names and bit names by adding the following statement in your code to include the header file;

```
# include p18f1220.inc
```

Experiment #2

Modify the code in experiment 1 in order for EDbot to drive an ellipse (per specifications shown below) pattern that takes between 5 to 15 seconds to complete.



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.

```
;-------------------------------------------------------------------------------
; FILE: EDbotTest.asm
; DESC: Design to test EDbot basic functionality
; LAST UPDATE: 6/15/2016
; AUTH: Class
; DEVICE: PICmicro (PIC18F1220)
;-------------------------------------------------------------------------------
   list        p=18F1220           ; processor type
   radix       hex                 ; default radix for data
   config      WDT=OFF, LVP=OFF, OSC = INTIO2      ; Disable Watchdog timer,  Low V. Prog, and RA6 as a clock

#include       p18F1220.inc        ; This header file includes address and bit definitions for all SFRs

#define        dCount              0x80
#define        dCountInner         0x81

   org      0x000          ; Set the program origin (start) to absolute 0x000

   ; Initialize all I/O ports
   CLRF       PORTA               ; Initialize PORTA
   CLRF       PORTB               ; Initialize PORTB
   MOVLW      0x7F                ; Set all A\D Converter Pins as
   MOVWF      ADCON1              ; digital I/O pins
   MOVLW      0x0D                ; Value used to initialize data direction
   MOVWF      TRISA               ; Set Port A direction
   MOVLW      0xC7                ; Value used to initialize data direction
   MOVWF      TRISB               ; Set Port B direction
   MOVLW      0x00                ; clear Wreg

   ; Toggle Portb,5, direction, and delay.
   ; start by going forward for first delay cycle
Main:
   BSF        PORTB,4             ;Enable Right motor
   BSF        PORTA,6             ;Forward Right
   BSF        PORTB,3             ;Enable Left Motor
   BCF        PORTA,7             ;Backward Left
   MOVLW      .1
   CALL       Delay
   BCF        PORTA,6             ;Backward Right
   BSF        PORTA,7             ;Forward Left
   MOVLW      .1
   CALL       Delay
   BCF        PORTB,4 ;Disable Right
   BCF        PORTB,3 ;Disable Left Motor
   MOVFF      PORTA, 0x82 ; prime for first loop cycle.

Loop:   ; Toggle LED
   BTG         PORTB,5
   MOVLW      .5
   CALL       Delay
   MOVF        PORTA, 0            ; W = PORTA
   XORWF      0x82, 0             ; W = W XOR LASTIN
   BZ         Loop                ; Loop if zero
   BRA        Main                 ;Restart when Dip switch 1 and 4 is changed

; Delay function - uses the Wreg value as the number of 1/10 of seconds delay period
Delay:
   MOVWF      dCount
DelayLoop:
   CALL       DelayOnce
   DECF       dCount
   BNZ        DelayLoop
   RETURN
DelayOnce:
              CLRF dCountInner     ;Internal delay loop
DelayOnceLoop:
    NOP
   INCF       dCountInner
   BNZ        DelayOnceLoop
   RETURN

   end        ; code end
```

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.