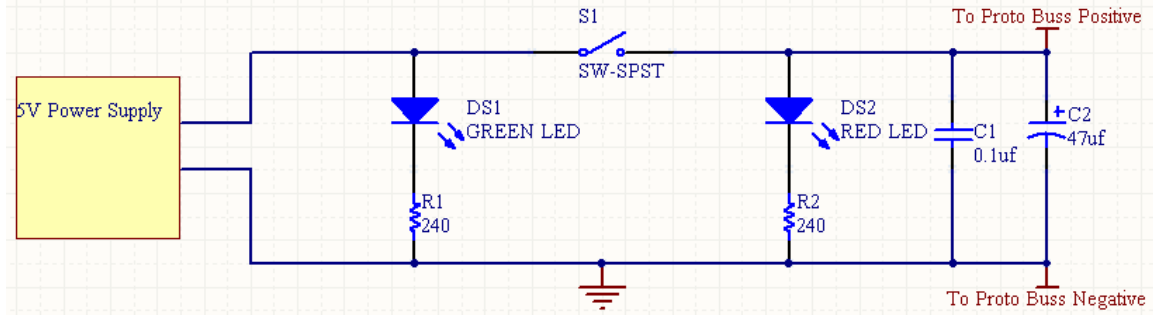# PROTO BOARD SETUP

**Here is a schematic of the circuit we will build.**

Cut off the connector from the end of your 5V supply and strip the insulation off for one half inch. Tin the striped ends with solder.

Plug in your supply and measure the ends with your multimeter. Mark the positive end. Place the positive end into the red binding post with a red jumper wire. Place the negative end into the black binding post with a red jumper wire as shown below.
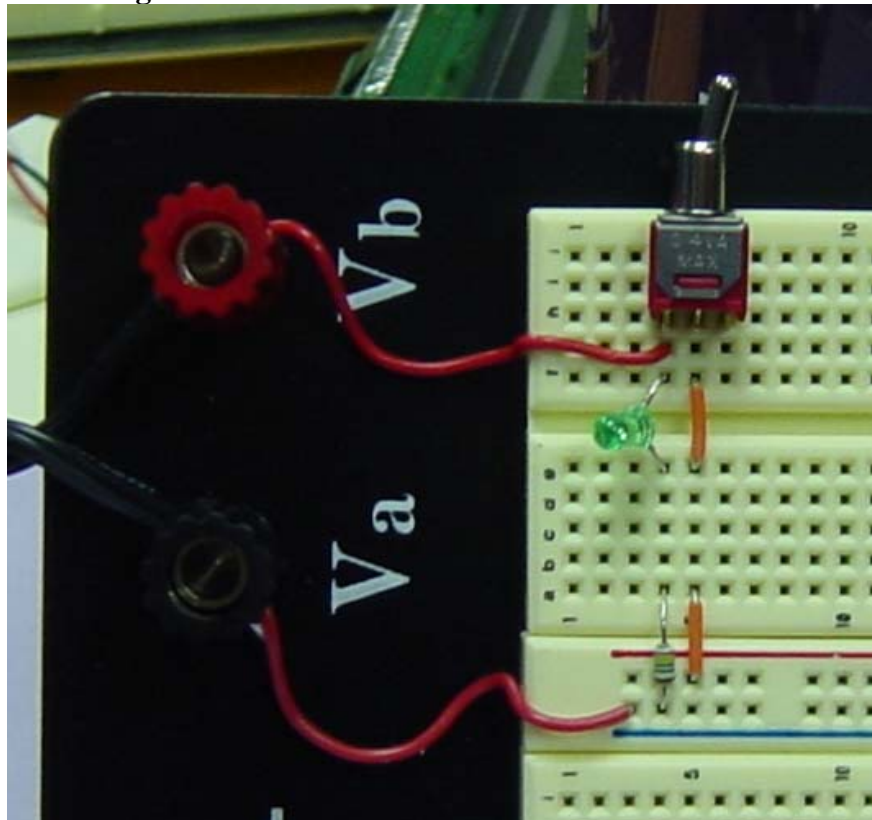

Next take your switch and cut off one of the silver mounting posts as shown below.
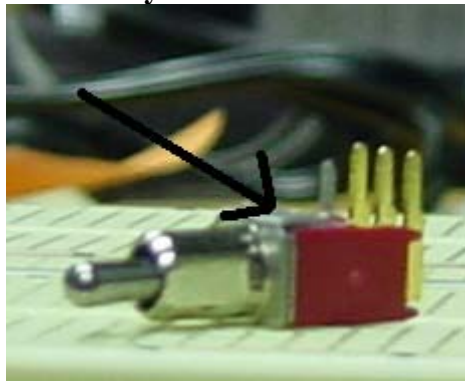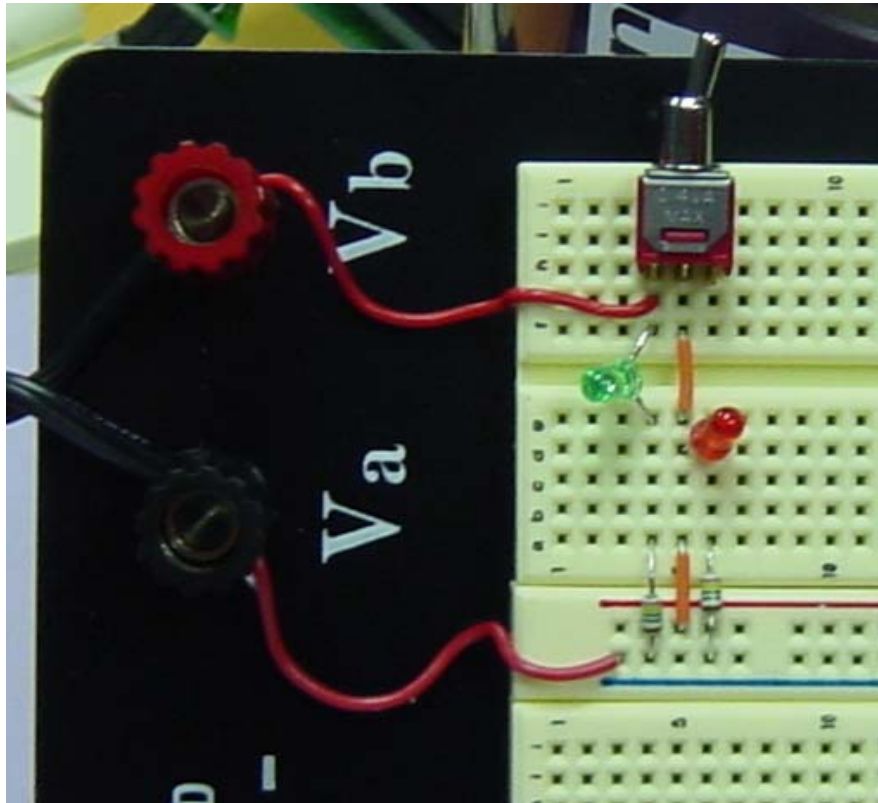
Next we will start the wiring of the proto board. Install the two red power wires and switch as shown. Then install the two orange jumper wires. See below.
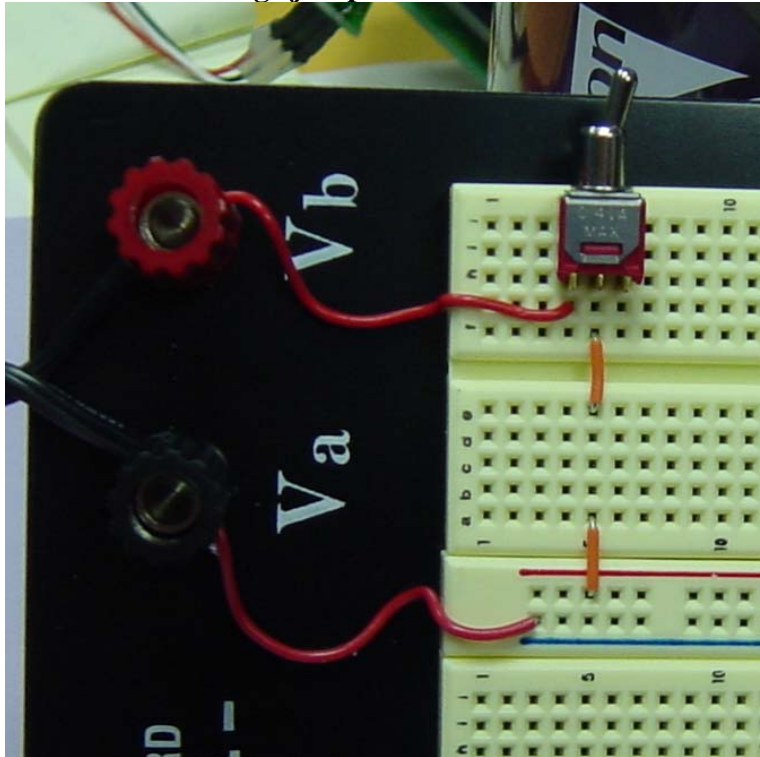
Next we can install the green LED and the 240ohm ballast resistor as shown below.

Now we can install the red LED with its ballast resistor. See below.

We can now install the two capacitors on the power buss. A 0.1uf and a 47uf. Pay attention to the polarity on the 47uf capacitor. The long lead is positive and the negative side is marked with a stripe on the side of the body.
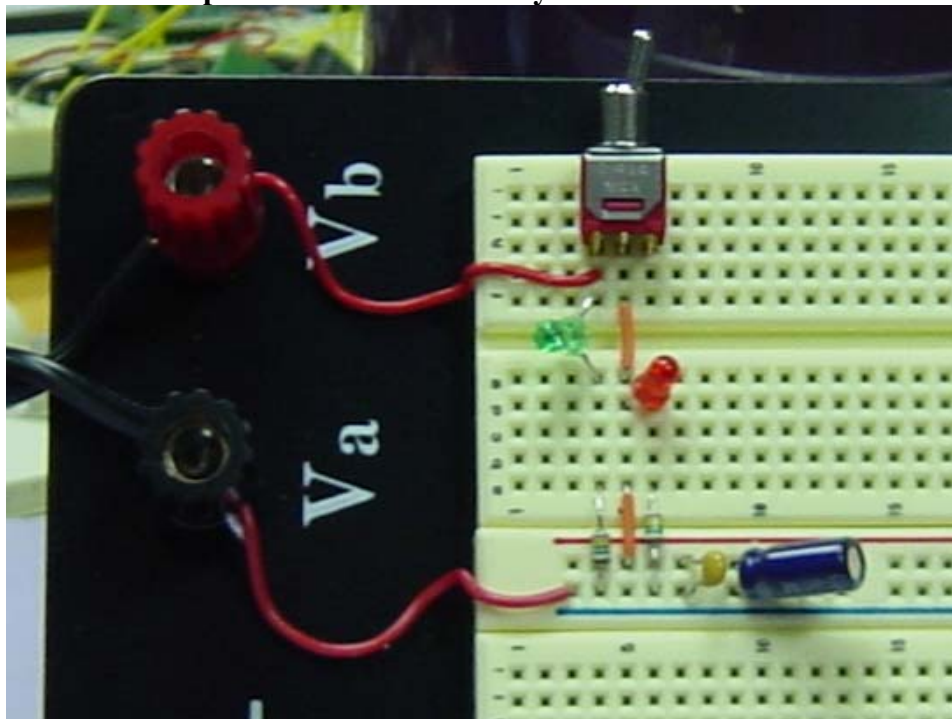
The last step is to secure your switch on to the board with super glue. This will keep the switch from popping off of the board during operation.

Junior/Senior Design PIC Microprocessor Demo 1

Here is the schematic for our PIC microprocessor demo.
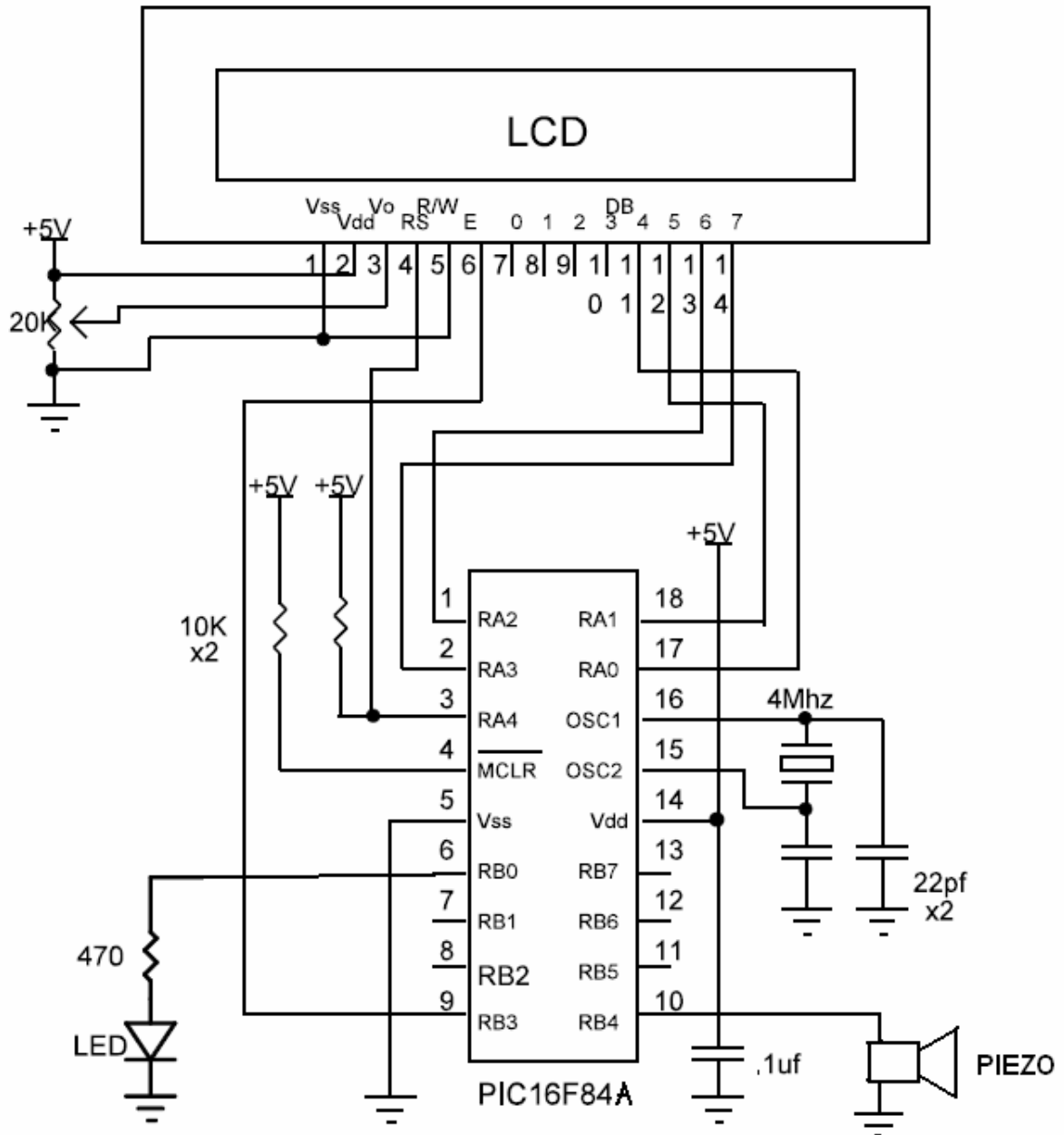
This is the code used for the demo setup.

```
' PicBasic program to demonstrate operation of an LCD in 4-bit mode
' It also flashes an LED and beeps a Piezo element
' LCD should be connected as follows:
'        LCD      PIC
'        DB4      PortA.0
'        DB5      PortA.1
'        DB6      PortA.2
'        DB7      PortA.3
'        RS       PortA.4 (add 4.7K pullup resistor to 5 volts)
'        E        PortB.3
'        RW       Ground
'        Vdd      5 volts
'        Vss      Ground
'        Vo       10K-20K potentiometer
'        DB0-3    No connect

        cnt VAR BYTE
        TRISB.0 = 0 ' Set PORTB, pin 0 to an output
        PAUSE 500        ' Wait 500mS for LCD to startup

loop:
        LCDOUT $fe, 1    ' Clear LCD screen
        PAUSE 500        ' Wait .5 second
        LCDOUT $fe,$80   ' Move cursor to the begening of the first line
        LCDOUT "Junior Design"  ' DisplayJunior Design
        PAUSE 500        ' Wait .5 second

        LCDOUT $fe,$c0   ' Move cursor to the begening of the second line
        LCDOUT "PIC Lecture"  'Display PIC Lecture
        PAUSE 500        ' Wait .5 second
        GOSUB Alert
        GOTO loop        ' Do it forever

Alert:
            FOR cnt = 1 TO 4
            FREQOUT portb.4,75,2000 'send 2KHz tone on Portb.4
            TOGGLE portb.0   'Toggle port B.0 output state or Blink LED
            PAUSE 75                    ' Pause 75mS
            NEXT cnt
            RETURN
 END
```

Junior/Senior Design Homework

Add two pushbutton switches to your LDC proto board. Connect one switch so that the input is normally high and the second switch so that the input is normally low. Now modify the current code so that when the first switch is pressed the LCD will display "Switch one pressed". Also add to the code to display on the LCD "Switch two pressed" when the second switch is pressed. The program should resume the demo loop when the switches are no longer activated. After you succeed with this, make the first switch display on the first line and the second switch display on the second line.

Junior/Senior Design PIC Microprocessor Demo 2

Here is the schematic for our PIC microprocessor A/D demo.

This is the code used for the second demo setup.

```
' PicBasic program to demonstrate operation of an LCD in 4-bit mode
' using the A/D input on portb.0 as display data
' LCD should be connected as follows:
'      LCD    PIC
'      DB4    PortA.0
'      DB5    PortA.1
'      DB6    PortA.2
'      DB7    PortA.3
'      RS     PortA.4 (add 10K pull up resistor to 5 volts)
'      E      PortB.3
'      RW     Ground
'      Vdd    5 volts
'      Vss    Ground
'      Vo     10K-20K potentiometer
'      DB0-3  No connect
```

```
Define  ADC_BITS       10      ' Set number of bits in result
Define  ADC_CLOCK      3      ' Set clock source (3=rc)
Define  ADC_SAMPLEUS   50        ' Set sampling time in uS




        OSCCON = %01100010 ' Set up internal oscillator
ADCON0 = %00010011' Set up the A/D channel and references
        ADCON1 = %01101111      ' Set up the I/O pins as digital or analog
        ADCON2 = %10111110      ' Set up A/D result format and acquisition time



volts VAR WORD        ' Variable to store A/D result in
dec0  VAR BYTE        ' Variable to store ones digit
ten0  VAR BYTE        ' Variable to store tenths digit

TRISB.0 = 1 ' Set PORTB, pin 0 to an input

Pause 500       ' Wait 500mS for LCD to startup
LCDOut $fe, 1   ' Clear LCD screen
Pause 1              ' Wait 1 millisecond
LCDOut $fe,$80  ' Move cursor to the beginning of the first line
LCDOut "A/D Demo using"  ' Display
LCDOut $fe,$c0  ' Move cursor to the beginning of the second line
LCDOut "PIC18F1320"  'Display PIC Lecture
Pause 2000      ' Wait 2 seconds

loop:
    ADCIN 4,volts       ' Read in the A/D value
    dec0 = volts / 204         ' Calculate the ones digit
    ten0 = volts // 204 ' Calculate tenths digit
    ten0 = ten0 / 21   ' Finish calculatong tenths
    LCDOut $fe, 1   ' Clear LCD screen
    LCDOut $fe,$80  ' Move cursor to the beginning of the first line
    LCDOut DEC1(dec0),".",DEC1(ten0),"Volts"," ",dec4 volts,"DEC"
    LCDOut $fe,$c0  ' Move cursor to the beginning of the second line
    'lcdout rep 255\(volts/51)
    LCDOut BIN10 volts,"Bin ",HEX3 volts,"Hex"  'Display PIC Lecture
    Pause 100       ' Wait 0.1 second

    GoTo loop       ' Do it forever

End
```