



**Birzeit University**  
**Faculty of Engineering & Technology**  
**Department of Electrical & Computer Engineering**  
**ENEE4101**

**“PCB and MicroC Report”**

**Student :Mohamad Bornat #1130842**

**Instructor:Dr.Jamal Siam**

**Date: 4-3-2017**

# *Abstract*

**In these two experiments, Proteus 7 Professional and MicroC PRO software were used to act as a power supply for the layout and to develop codes for the PIC controllers respectively. the PCB technology and the Mircochip PIC Controllers were introduced to us in this experiment .**

# Contents

<b>Acronyms and Abbreviations</b> .....	<b>ii</b>
<b>List of Figures</b> .....	<b>iii</b>
<b>Chapter 1 Introduction</b> .....	<b>1</b>
<b>Chapter 2 Procedure</b> .....	<b>2</b>
<b>2.1 Methods and Materials</b> .....	<b>2</b>
<b>2.1.1 PCB Experiment Procedure</b> .....	<b>2</b>
<b>2.1.2 PCB Experiment Results</b> .....	<b>4</b>
<b>2.1.2.1 Procedure of MicroC Experiment</b> .....	<b>6</b>
<b>2.1.2.2 Results of MicroC Experiment</b> .....	<b>8</b>
<b>Chapter 3 Discussion</b> .....	<b>10</b>
<b>Chapter 4</b> .....	<b>11</b>
<b>Conclusion</b> .....	<b>11</b>
<b>References</b> .....	<b>12</b>
<b>Appendices</b> .....	<b>13</b>
<b>Appendix A</b> .....	<b>13</b>

# Acronyms and Abbreviations

PCB	Printed Circuit Board
CNC	Computer Numerical Control
PIC	Programmable Interface Controller

# List of Figures

Fig. 1.1: PCB Construction .....	1
Fig. 2.1: Voltage Regulator Circuit .....	2
Fig. 2.2: PCB Layout Parts.....	3
Fig. 2.3: Routed Board .....	3
Fig. 2.4: Routed Board Result.....	4
Fig. 2.5 : 3D design of the circuit .....	5
Fig. 2.6: PIC and LCD Circuit Design.....	6
Fig. 2.7: A/D module.....	7
Fig. 2.8: Name and Number Output.....	8
Fig. 2.9: result for a low resistance .....	8
Fig. 2.10: result for a other high resistance .....	9

# Chapter 1

## Introduction

---

The main goal of these two experiments was to get more knowledge about Printed Circuit Boards (PCBs) technologies and Microchip PIC controllers. The purpose of the first experiment was to construct a layout for a given circuit to use it later as a practical reference to build the circuit using CNC machines. On the other hand, the purpose of the second experiment was to write a C language code and apply it to a PIC Controller and proteus was used to simulate the code.

PCBs are the most common method of assembling modern electronic circuits. Comprised of a sandwich of one or more insulating layers and one or more copper layers which contain the signal traces and the powers and grounds, the design of the layout of printed circuit boards can be as demanding as the design of the electrical circuit [1].

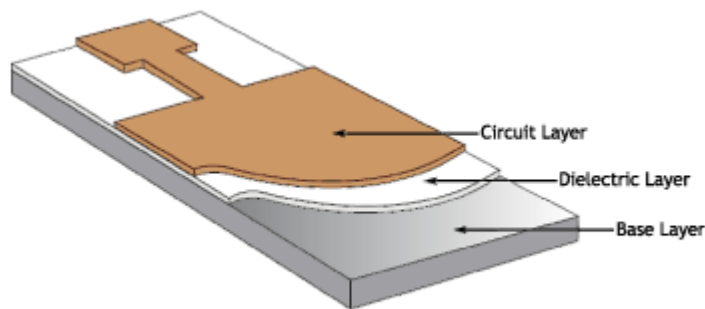


Fig. 1.1: PCB Construction.

After the layout is designed, it can be implemented on the board by several ways. Which goes into several stages .First, removing the unwanted copper cladding such that the designed layout (tracks, pads, etc.) is implemented by the copper then drilling the board to place the components. Finally, solder the desired connections [2].

Microcontrollers are different from microprocessors on many ways, the most important different is its functionality, to use a microprocessor the components must be added to it. On the other hand, the microcontroller does not need all the other components, it is all built in one, so it saves time cost and space [3].

# Chapter 2

## Procedure

---

### 2.1 Methods and Materials

In these two experimnts the following tools were used :

- Proteus 7 Professional program.
- CNC Machine.
- Micro Pro program.

#### 2.1.1 PCB Experiment Procedure

Proteus professional software was started to create a schematic page and a PCP layout page. In order to create a new project, these steps were followed:

From *Start Menu >> Programs >> Proteus 7 Professional >> Isis 7 Professional*, the Isis was started to start the project.

Then the voltage regulator circuit was drawn using proteus as follows:

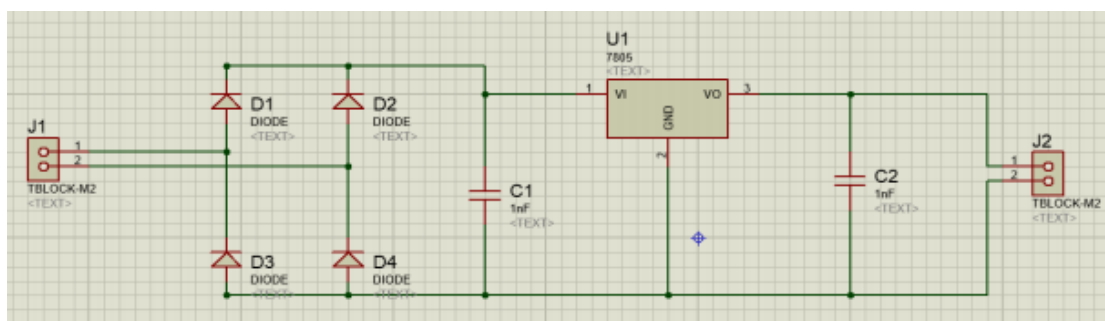


Fig. 2.1: Voltage Regulator Circuit.

After that the needed parts were placed into a PCB layout page as follows:

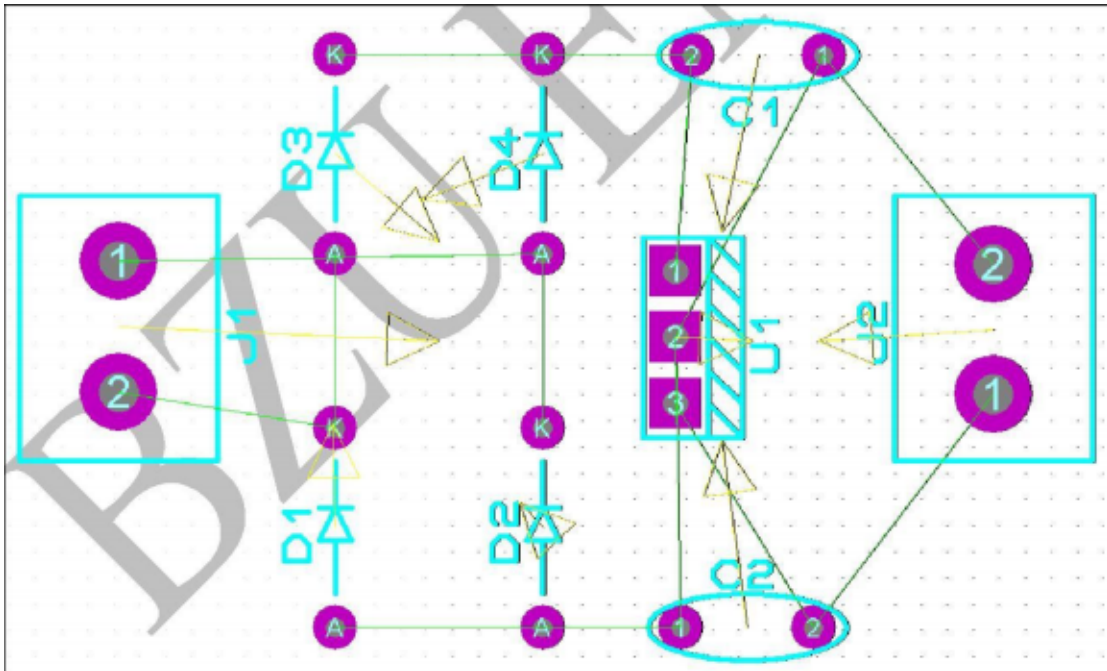


Fig. 2.2: PCB Layout Parts

Routing layers were determined by going to Technology >> Design Rule Manager >> Net Classes.

And then the board was routed by going to Tools >> Auto Router >> Begin Routing

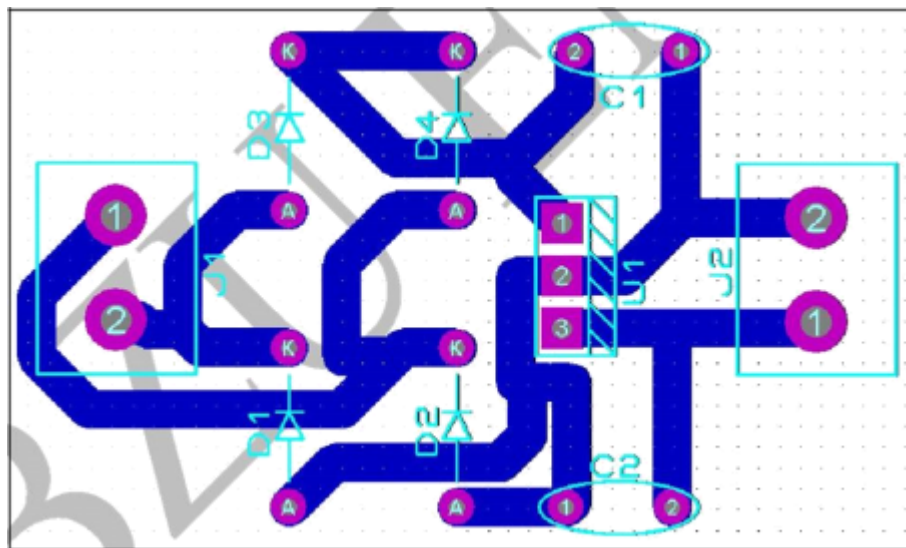


Fig. 2.3: Routed Board



Finally, the files needed for the PCB manufacturing process were generated. The output files are called Gerber files which are the main output of the PCB layout software since the Gerber format is a standard format that contains a map of the tracks, the pins and other specifications of the board.

### 2.1.2 PCB Experiment Results

The board was routed as it is shown in the next figure :

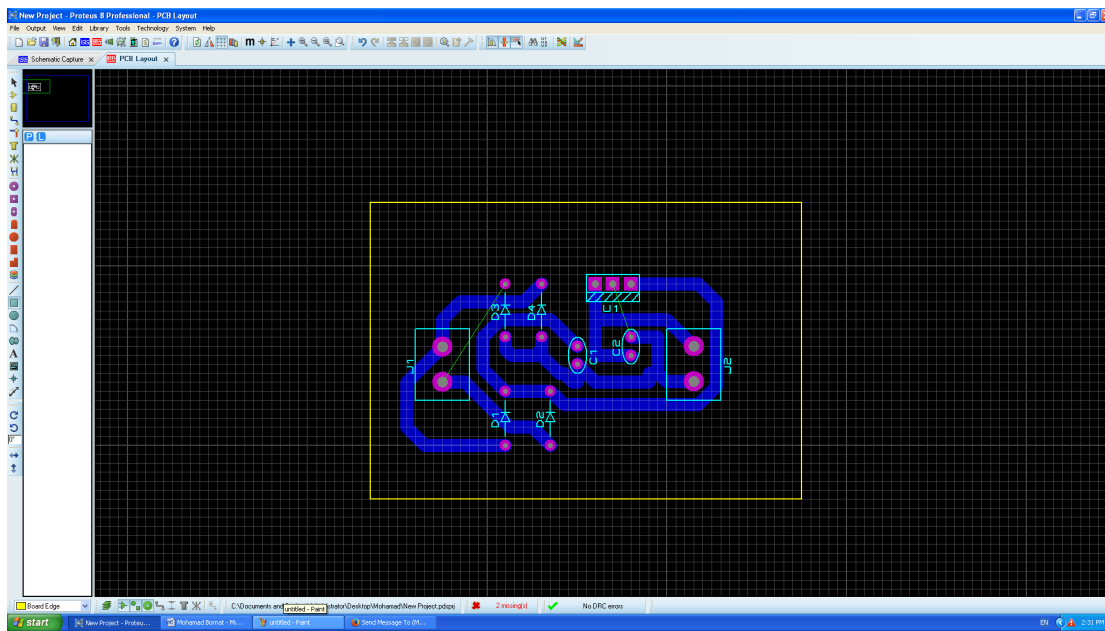


Fig. 2.4: Routed Board Result

A 3D design of the circuit was created as shown in the next figure:

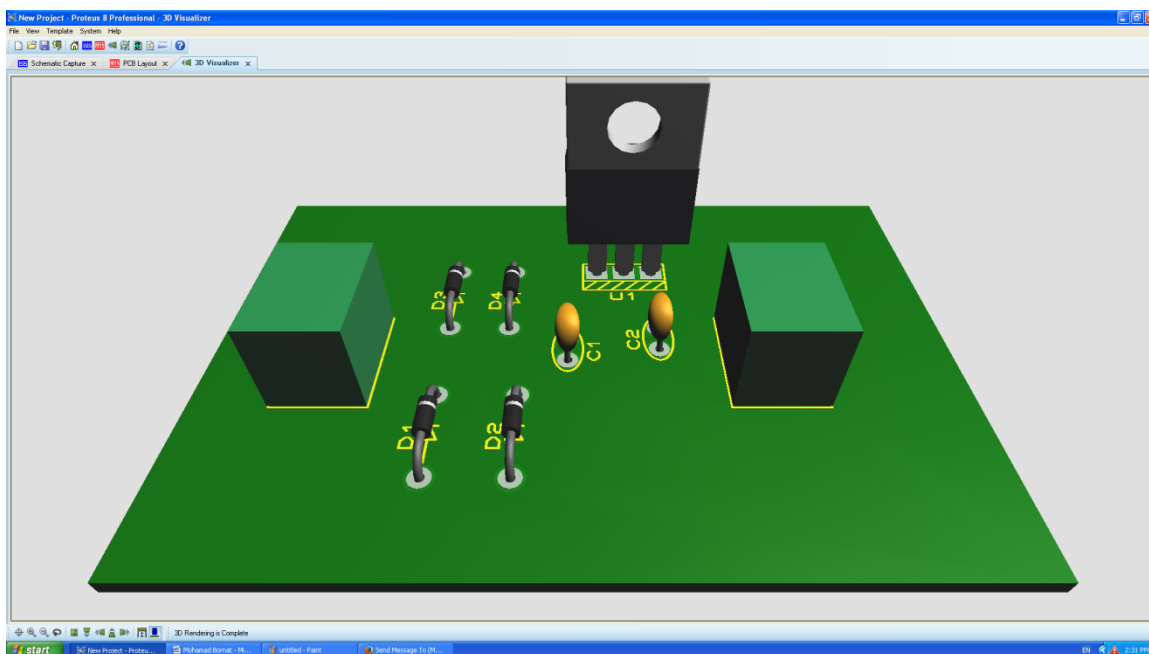


Fig. 2.5 : 3D design of the circuit

### 2.1.2.1 Procedure of MicroC Experiment

In this part a code was created to control a 16X2 LCD using a 16F84A PIC with 4MHZ clock frequency.

After creating a new project, the code (A.1) was typed.

After that, the code was built from build menu, the hex file was created.

Then Protues was used to draw the circuit as shown in the next figure, then we double click on the PIC in order to choose the created hex file. Then the clock frequency was changed to 4MHz.

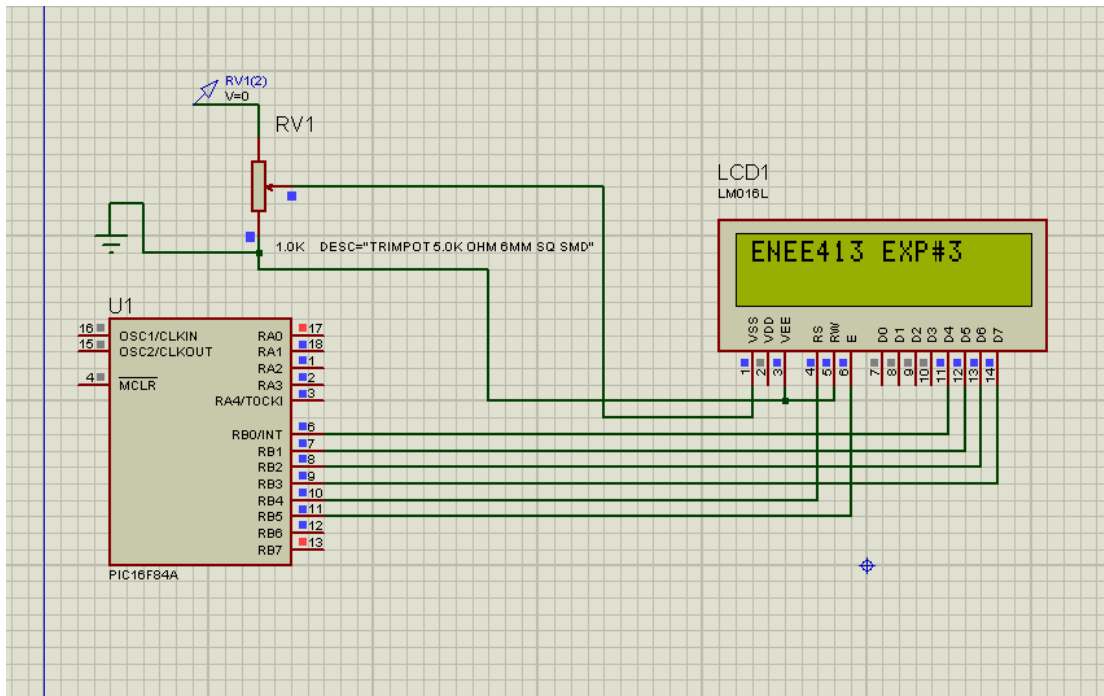


Fig. 2.6: PIC and LCD Circuit Design

As a classwork we were asked to print our names and numbers each in one line, and so the next code (A.2) was used.

In the part of using A/D Module , 16F877A PIC microcontroller was used ,the PIC was supposed to read the analog input, and convert it into digital and outputs the digital value in binary representation on port B. after a new MicroC project was created, the (A.3) code was written and built.

Then Protues was used to draw the circuit as shown in next figure and the hex file was installed to the controller.

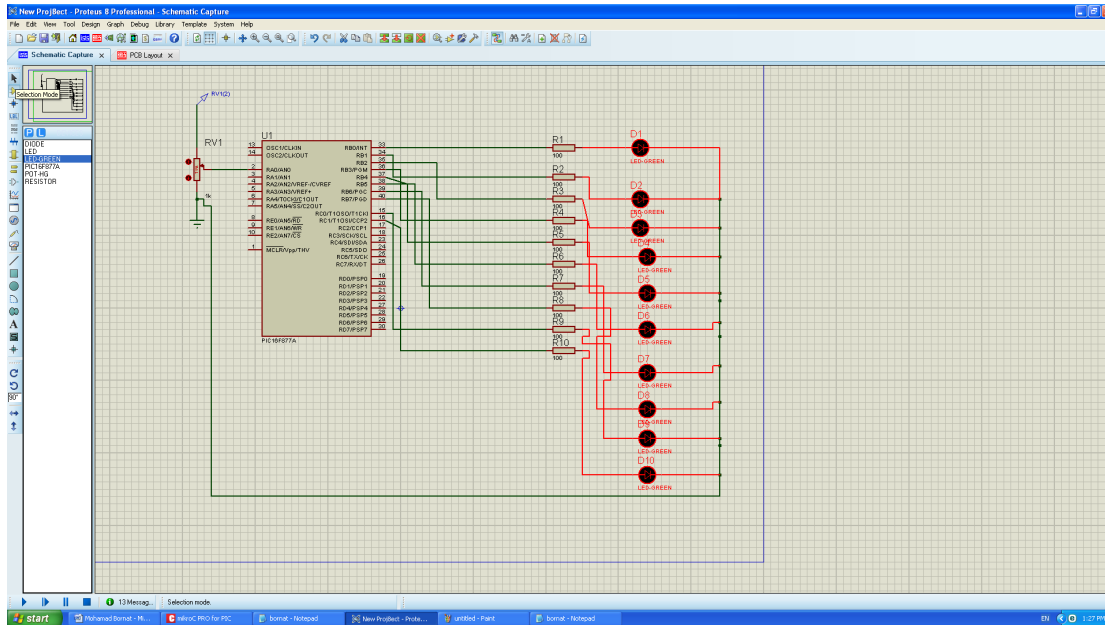


Fig. 2.7: A/D module

### 2.1.2.2 Results of MicroC Experiment

The result of the class exercise that we were asked to do is shown in the next figure :

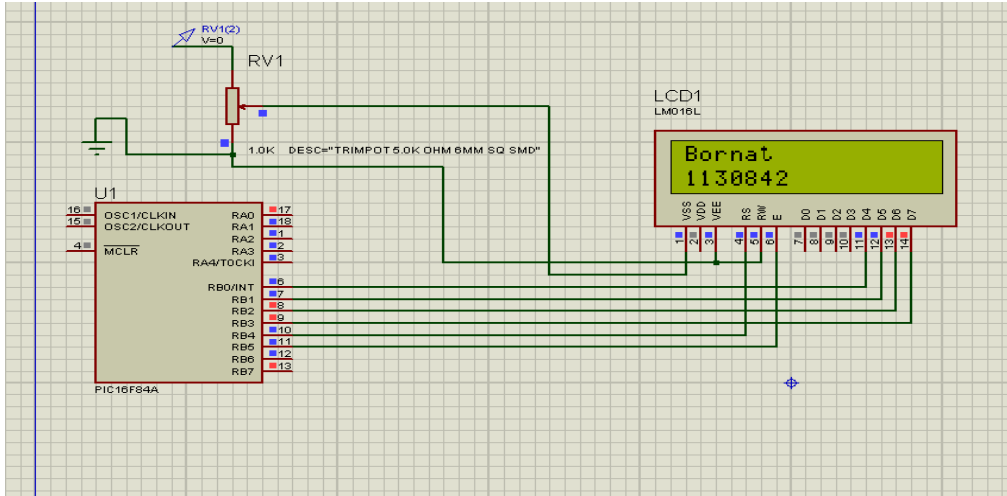


Fig. 2.8: Name and Number Output

Using the A/D module the results are shown in the next figures, these figures are for different values of the potentiometer resistance:

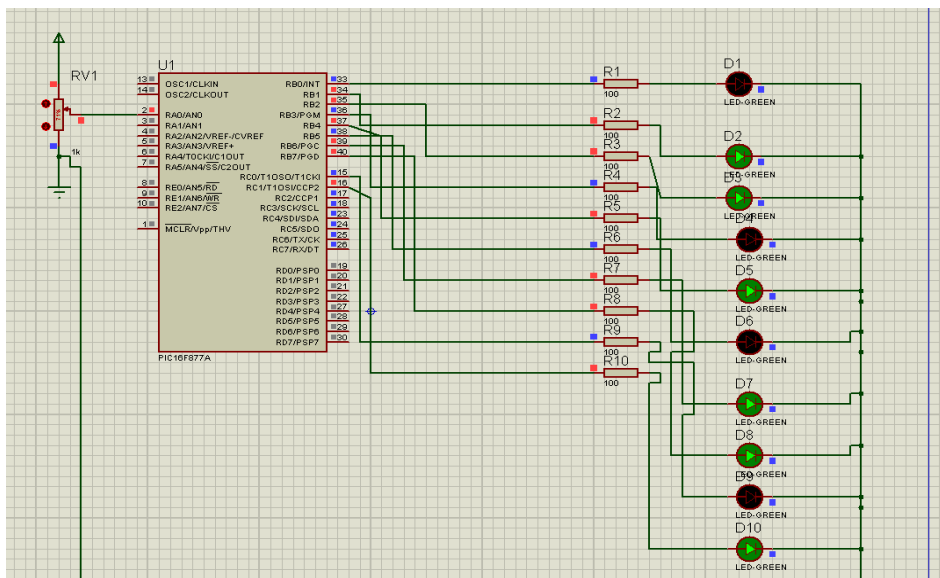


Fig. 2.9: result for a low resistance

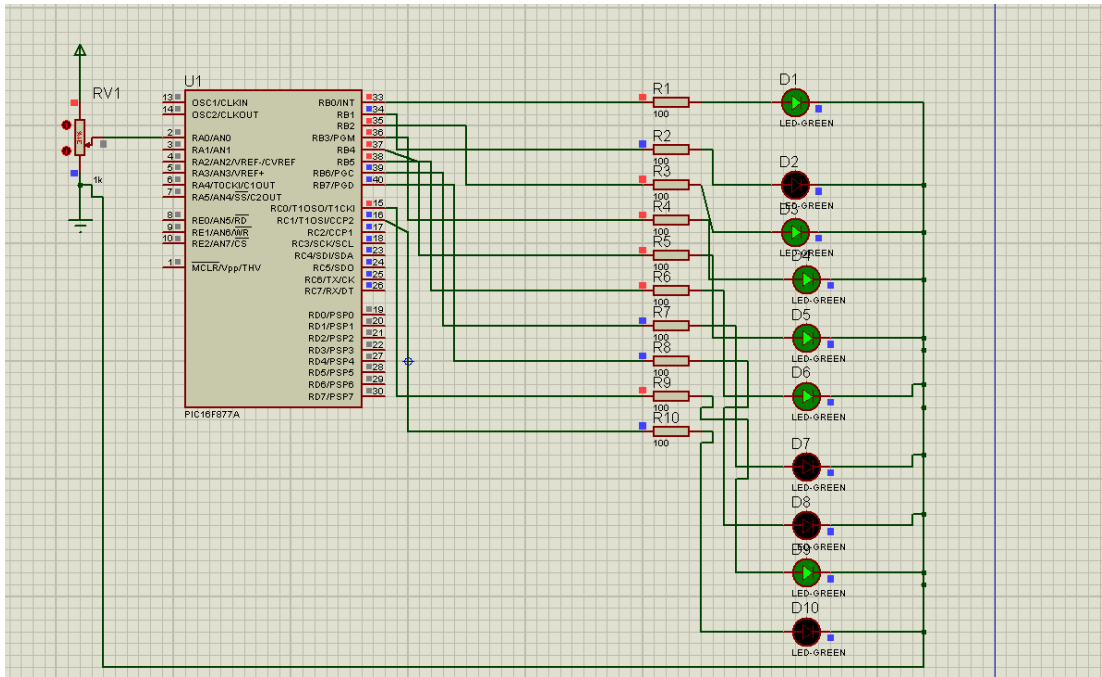


Fig. 2.10: result for a other high resistance

# Chapter 3

## Discussion

---

The obtained results were close enough to the desired, this shows us how much these softwares are important, the MircroC program is very helpful to test the code before installing it on a real PIC controller and the Proteus software is useful in designing and simulating the circuits, and these two programs can work with each other and complete each other.

It was helpful and an easy way to design the PCBs , I faced a problem with routing the board because it gave me that some connections are missing, I reduced the missing connections as much as I could but the time wasn't my friend in finishing the layout without missing lines.

We saw that the most important thing in the design that we should make the space of the design small as much as possible to avoid missing lines and to avoid expensive cost when building the layout using CNC machines.

One of the beautiful features of the software that we can make a 3D image of the design to make sure that's the components ordered well and the design does not waste a lot of the copper.

I also faced one other problem which is the output of the first part was flashing rapidly so we couldn't see the output wee, in order to get rid of this problem a delay command was used which makes the output lasts on the LCD as much time as I want.

# Chapter 4

## Conclusion

---

In this experiments, two different programs were used, Proteus and MicroC, the MicroC program was used to write a C language code and create a hex file in order to use it as a reference for the microcontroller that was built using Proteus, this shows how much these two programs are related to each other.

Proteus was also used to design PSBs to implement it later using CNC machines.



# References

---

[1] "Signal Integrity Issues and Printed Circuit Board Design", Douglas Brooks.

[2] Lab Simulation Manual

[3] "PIC Microcontroller for Beginners", Nebojsa Matic

# Appendices

---

## Appendix A

```
// Lcd pinout settings
sbit LCD_RS at RB4_bit;
sbit LCD_EN at RB5_bit;
sbit LCD_D7 at RB3_bit;
sbit LCD_D6 at RB2_bit;
sbit LCD_D5 at RB1_bit;
sbit LCD_D4 at RB0_bit;
// Pin direction
sbit LCD_RS_Direction at TRISB4_bit;
sbit LCD_EN_Direction at TRISB5_bit;
sbit LCD_D7_Direction at TRISB3_bit;
sbit LCD_D6_Direction at TRISB2_bit;
sbit LCD_D5_Direction at TRISB1_bit;
sbit LCD_D4_Direction at TRISB0_bit;
void main()
{
TRISA = 0x00; // set all pins of port A as output
TRISB = 0x00; // set all pins of port B as output
while(1)
{ lcd_init(); // initialize the lcd
lcd_out(1,1,"ENEE413 EXP#3"); // print this message
Lcd_Cmd(_LCD_CURSOR_OFF);
}}
```

(A.1)

```

while(1)
{ lcd_init(); // initialize the lcd
  lcd_out(1,1,"Bornat"); // print this message
  lcd_out(2,1,"1130842");
  Lcd_Cmd(_LCD_CURSOR_OFF);
  delay_ms(800);
}
}

```

(A.2)

```

unsigned int a;
void main() {
  TRISA = 0xFF; // PORTA is input
  TRISC = 0; // PORTC is output
  TRISB = 0; // PORTB is output
  do {
    a = ADC_Read(0); // Get 10-bit results of AD conversion
    PORTB = a; // Send lower 8 bits to PORTB
    PORTC = a >> 8; // Send 2 most significant bits to RC1, RC0
  } while(1);
}

```

(A.3)