

LAB File Record
B.Tech. VI Sem

Subject : Embedded System Design LAB (CSP315)



Department of Computer Science & Engineering
Malaviya National Institute of Technology Jaipur
Session: 2022 - 2023

Submitted By :
Name : Mohak
ID : 2020UCP1025
Batch :A1

Program List

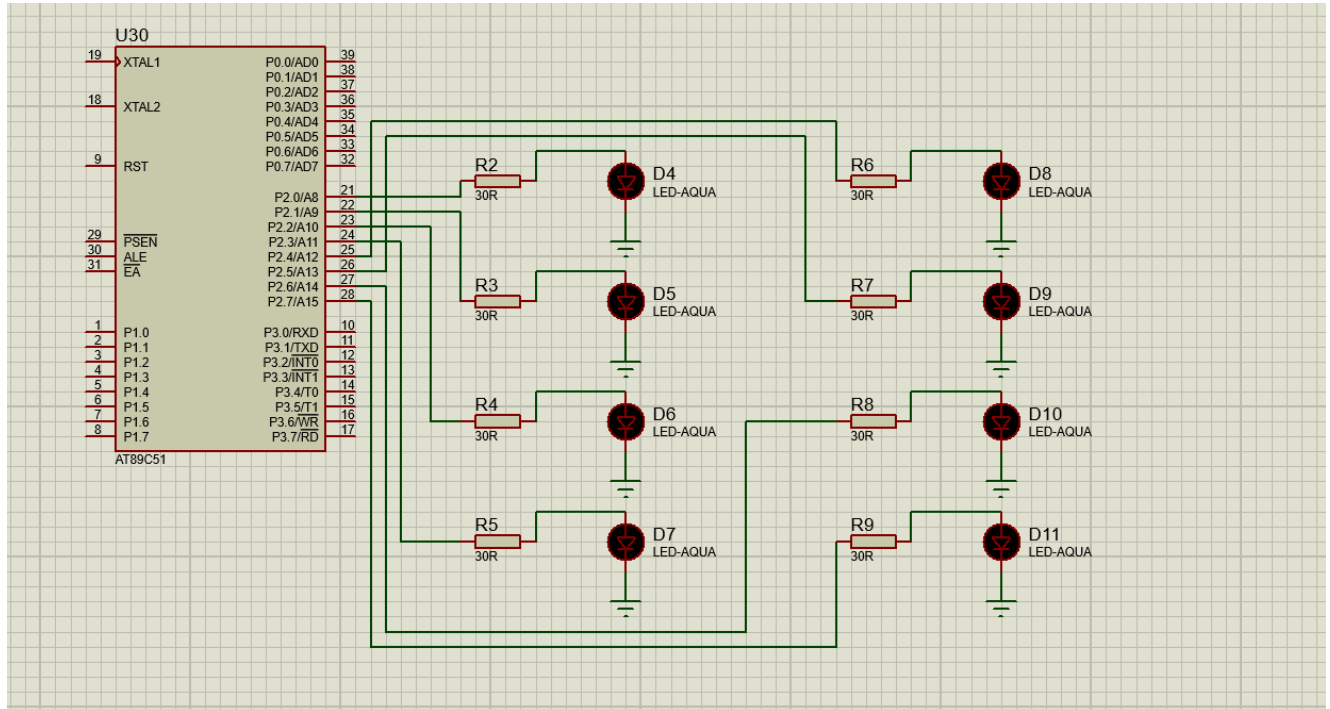
Sr. No.	Program Objective	Page No.
1	Working of LED a. Wave pattern b. Circular rotation Pattern	
2	Display a message on LCD.	
3	Display the pressed key of hex keypad on LCD.	
4	Implement a simple calculator.	
5	Display the pressed key of the hex keypad on 4 digit 7-Segment Display.	
6	Implement an increasing counter and show on 4 digits 7-Segment Display.	
7	Set the counter using a hex keypad and display on 7 Segment.	
8	Write a program to show the movement of Elevator.	
9	Write a program for Quiz-answer setup using Relay Buzzer and LCD.	
10	Display date and time on LCD using RTC section.	

LAB 1

Program Objective: Blinking of LED in circular rotation Pattern

Component used: 8051 microprocessor, LEDs, Resistors

Connection Diagram:



Program Code:

```
// write a program to blink the led in rotation
```

```
#include <reg51.h>
```

```
#define LED P2
```

```
void delay(int);
```

```
void main() {
```

```
    while (1) {
```

```
        int i = 1, j;
```

```
        for (j = 1; j <= 7; j++) {
```

```
            int K = i, r = 9 - j;
```

```
            while (r--) {
```

```
                LED = K;
```

```
                delay(7);
```

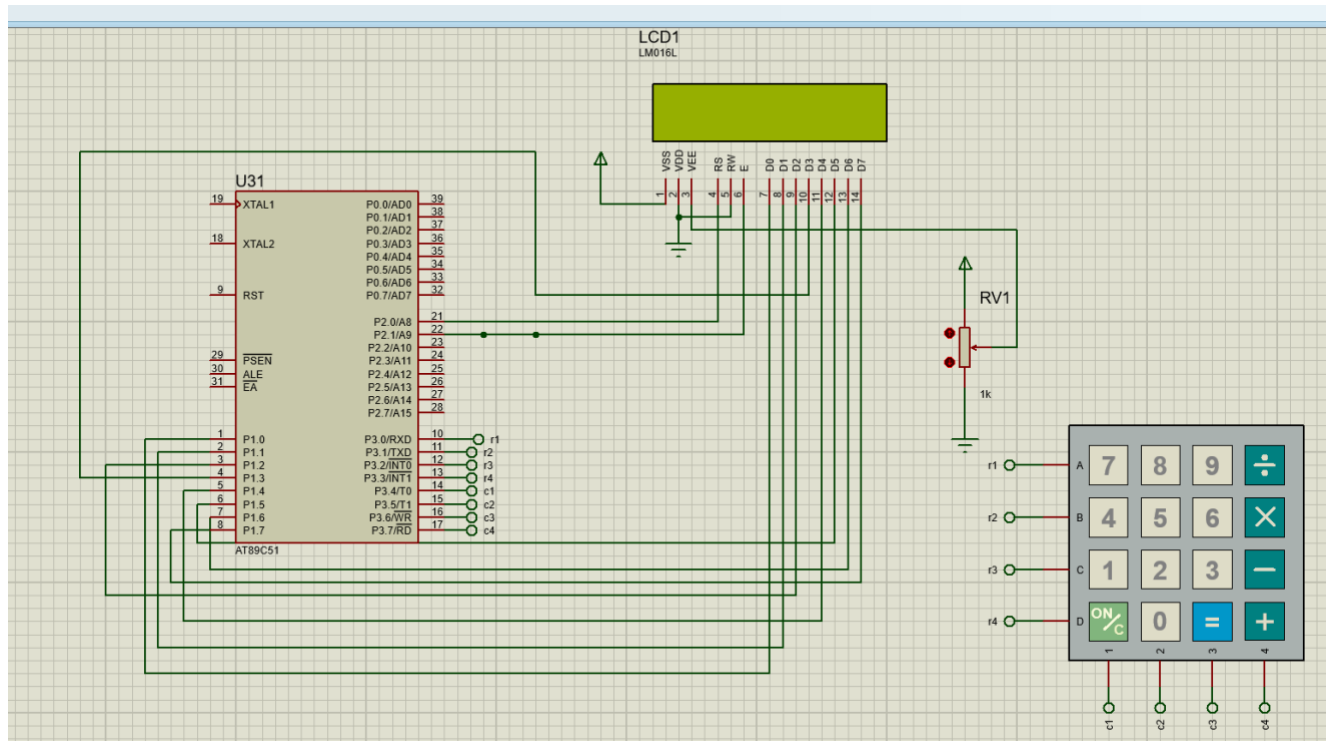
```
                LED = 0;
```

```
        delay(2);
        K <<= 1;
    }
    i = i * 2 + 1;
}
}
}
void delay(int x) {
    int i, j;
    for (i = 0; i < x; i++) {
        for (j = 0; j < 10000; j++) {
        }
    }
}
```

Program Objective: Display a message on LCD.

Component used: 8051 microprocessor, LEDs, Resistors

Connection Diagram:



Program Code:

```
#include <reg51.h>
sbit rs = P2 ^ 0;
sbit e = P2 ^ 1;
#define l P1
void delay(int);
void w_data(char);
void w_command(int);
void lcd_init();

void main() {
    char name[12] = "Mohak Sahil";
    char ids[24] = "1025 1105";
    while (1) {
        int i = 0;
        lcd_init();
        for (i = 0; i < 12; i++) {
```

```

        w_data(name[i]);
    }
    w_command(0xC0);
    for (i = 0; i < 24; i++) {
        w_data(ids[i]);
    }
    delay(10);
}

void lcd_init() {
    w_command(0x38); //
    w_command(0x0C); // line change
    w_command(0x01); // clear
    w_command(0x80); //
}

void w_command(int x) {
    rs = 0;
    P1 = x;
    e = 1;
    delay(5);
    e = 0;
    delay(5);
}

void w_data(char x) {
    rs = 1;
    P1 = x;
    e = 1;
    delay(2);
    e = 0;
    delay(2);
}

void delay(int x) {
    int i, j;
    for (i = 0; i < x; i++) {
        for (j = 0; j < 500; j++) {
        }
    }
}

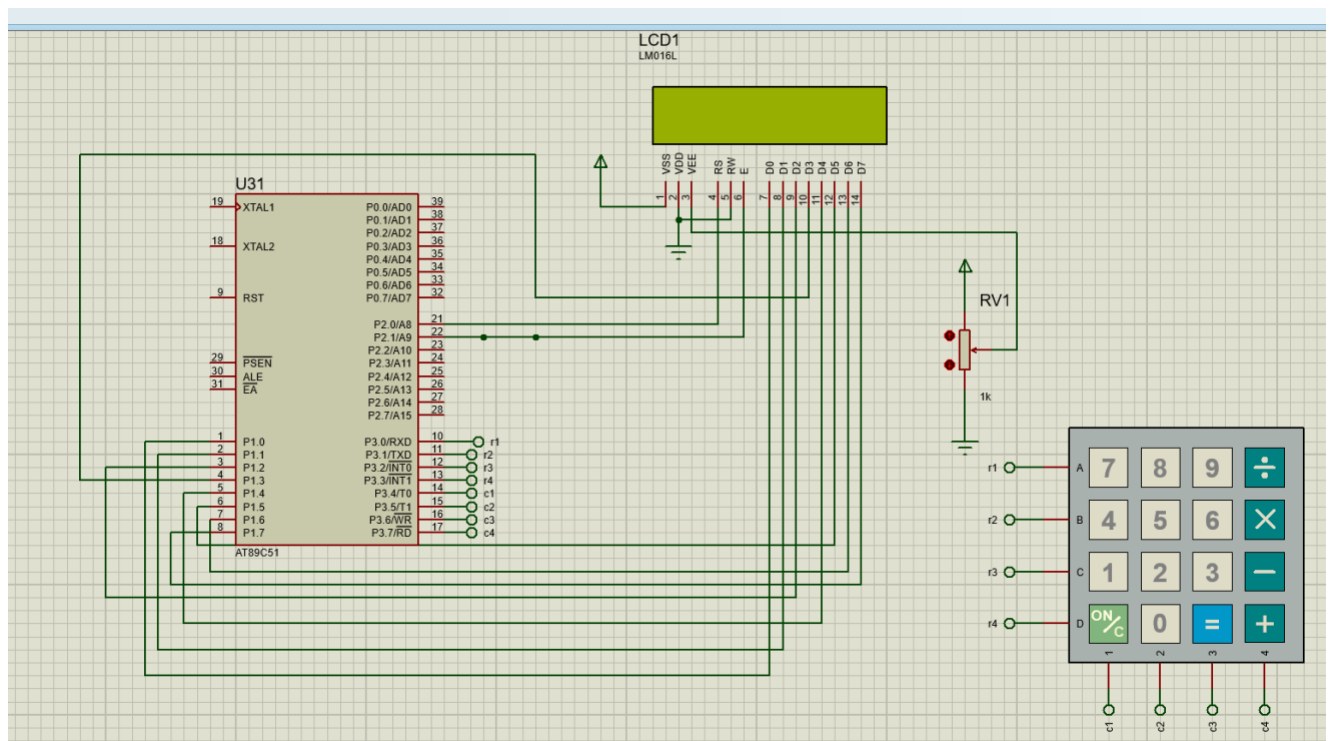
```

LAB 3

Program Objective: Display the pressed key of hex keypad on LCD.

Component used: 8051 microprocessor, LEDs, Resistors

Connection Diagram:



Program Code:

```
#include <reg51.h>
sbit rs = P2 ^ 0;
sbit e = P2 ^ 1;
#define l P1
void delay(int);
void w_data(char);
void w_command(int);
void lcd_init();

void main() {
    char name[12] = "Mohak Sahil";
    char ids[24] = "1025 1105";
    while (1) {
        int i = 0;
```

```

    lcd_init();
    for (i = 0; i < 12; i++) {
        w_data(name[i]);
    }
    w_command(0xC0);
    for (i = 0; i < 24; i++) {
        w_data(ids[i]);
    }
    delay(10);
}
}

void lcd_init() {
    w_command(0x38); //
    w_command(0x0C); // line change
    w_command(0x01); // clear
    w_command(0x80); //
}

void w_command(int x) {
    rs = 0;
    P1 = x;
    e = 1;
    delay(5);
    e = 0;
    delay(5);
}

void w_data(char x) {
    rs = 1;
    P1 = x;
    e = 1;
    delay(2);
    e = 0;
    delay(2);
}

void delay(int x) {
    int i, j;
    for (i = 0; i < x; i++) {
        for (j = 0; j < 500; j++) {
        }
    }
}

```

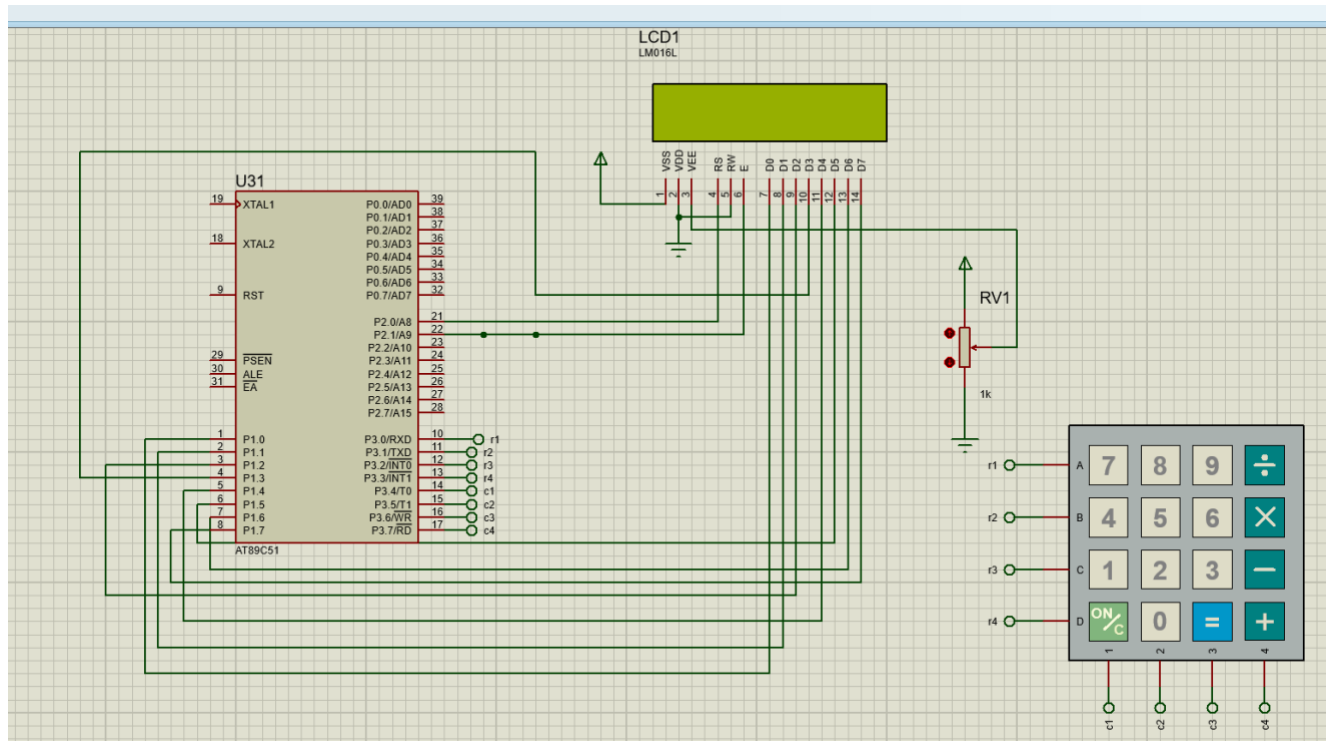

}

LAB 4

Program Objective: Implement a simple calculator

Component used: 8051 microprocessor, LEDs, Resistors

Connection Diagram:



Program Code:

```
#include <reg51.h>
#include
#define display_port P1 // Data pins connected to port 2 on microcontroller
sbit rs = P2 ^ 0; // RS pin connected to pin 2 of port 3
// sbit rw = P2^1; // RW pin connected to pin 3 of port 3
sbit e = P2 ^ 1; // E pin connected to pin 4 of port 3
sbit C4 = P3 ^ 7; // Connecting keypad to Port 1
sbit C3 = P3 ^ 6;
sbit C2 = P3 ^ 5;
sbit C1 = P3 ^ 4;
sbit R4 = P3 ^ 3;
sbit R3 = P3 ^ 2;
```

```

sbit R2 = P3 ^ 1;
sbit R1 = P3 ^ 0;
int val = 0, val1, val2, val3, reverse, n, remainder, i;
void msdelay(unsigned int time) // Function for creating delay in milliseconds.
{
    unsigned i, j;
    for (i = 0; i < time; i++)
        for (j = 0; j < 1275; j++)
            ;
}

void lcd_cmd(
    unsigned char command) // Function to send command instruction to LCD
{
    display_port = command;
    rs = 0;
    e = 1;
    msdelay(1);
    e = 0;
}

void lcd_data(unsigned char disp_data) // Function to send display data to LCD
{
    display_port = disp_data;
    rs = 1;
    e = 1;
    msdelay(1);
    e = 0;
}

void lcd_init() // Function to prepare the LCD and get it ready
{
    lcd_cmd(0x38); // for using 2 lines and 5X7 matrix of LCD
    msdelay(10);
    lcd_cmd(0x0F); // turn display ON, cursor blinking
    msdelay(10);
    lcd_cmd(0x01); // clear screen
    msdelay(10);
    lcd_cmd(0x80); // bring cursor to position 1 of line 1
    msdelay(10);
}
void first() {

```

```

char a[13] = "FIRST NUMBER";
int i;
for (i = 0; i < 13; i++) {
    lcd_data(a[i]);
}
lcd_cmd(0xC0);
}

void second() {
    char a[15] = "SECOND NUMBER";
    int i;
    for (i = 0; i < 15; i++) {
        lcd_data(a[i]);
    }
    lcd_cmd(0xC0);
}

void row_finder1() // Function for finding the row for column 1
{
    R1 = R2 = R3 = R4 = 1;
    C1 = C2 = C3 = C4 = 0;
    if (R1 == 0) {
        val = val * 10 + 7;
        lcd_data('7');
    }
    if (R2 == 0) {
        val = val * 10 + 4;
        lcd_data('4');
    }
    if (R3 == 0) {
        val = val * 10 + 1;
        lcd_data('1');
    }
    if (R4 == 0) {
        lcd_init();
        first();
        val = 0;
    }
}

void row_finder2() // Function for finding the row for column 2
{
    R1 = R2 = R3 = R4 = 1;
    C1 = C2 = C3 = C4 = 0;

```

```

if (R1 == 0) {
    val = val * 10 + 8;
    lcd_data('8');
}
if (R2 == 0) {
    val = val * 10 + 5;
    lcd_data('5');
}
if (R3 == 0) {
    val = val * 10 + 2;
    lcd_data('2');
}
if (R4 == 0) {
    val = val * 10 + 0;
    lcd_data('0');
}
}
void row_finder3() // Function for finding the row for column 3
{
    R1 = R2 = R3 = R4 = 1;
    C1 = C2 = C3 = C4 = 0;
    if (R1 == 0) {
        val = val * 10 + 9;
        lcd_data('9');
    }
    if (R2 == 0) {
        val = val * 10 + 6;
        lcd_data('6');
    }
    if (R3 == 0) {
        val = val * 10 + 3;
        lcd_data('3');
    }
    if (R4 == 0) {
        val2 = val;
        if (val3 == 4)
            val = val1 / val2;
        else if (val3 == 3)
            val = val1 * val2;
        else if (val3 == 2)
            val = val1 - val2;
        else

```

```

    val = val1 + val2;
    lcd_data('=');
    lcd_init();
    if (val < 0)
        n = 0 - val;
    else
        n = val;

    reverse = 0;
    while (n != 0) {
        remainder = n % 10;
        reverse = reverse * 10 + remainder;
        n /= 10;
    }
    if (val3 == 4) {
        lcd_data('D');
        lcd_data('I');
        lcd_data('V');
        lcd_data('I');
        lcd_data('D');
        lcd_data('E');
        lcd_data(':');
    } else if (val3 == 3) {
        lcd_data('M');
        lcd_data('U');
        lcd_data('L');
        lcd_data('T');
        lcd_data('I');
        lcd_data('P');
        lcd_data('L');
        lcd_data('Y');
        lcd_data(':');
    } else if (val3 == 2) {
        lcd_data('S');
        lcd_data('U');
        lcd_data('B');
        lcd_data('T');
        lcd_data('R');
        lcd_data('A');
        lcd_data('C');
        lcd_data('T');
        lcd_data(':');
    }

```

```

    } else {
        lcd_data('A');
        lcd_data('D');
        lcd_data('D');
        lcd_data(':');
    }
    lcd_cmd(0xC0);
    if (val < 0) lcd_data('-');
    while (reverse != 0) {
        lcd_data('0' + reverse % 10);
        reverse /= 10;
    }
    n = val;
    while (n % 10 == 0) {
        lcd_data('0');
        n /= 10;
    }
}
}

void row_finder4() // Function for finding the row for column 4

{
    R1 = R2 = R3 = R4 = 1;
    C1 = C2 = C3 = C4 = 0;
    val1 = val;
    val = 0;
    if (R1 == 0) {
        val3 = 4;
        lcd_data('/');
    }
    if (R2 == 0) {
        val3 = 3;
        lcd_data('*');
    }
    if (R3 == 0) {
        val3 = 2;
        lcd_data('-');
    }
    if (R4 == 0) {
        val3 = 1;
        lcd_data('+');
    }
}

```

```

    lcd_init();
    second();
}

void main()

{
    lcd_init();
    first();
    while (1) {
        msdelay(30);
        C1 = C2 = C3 = C4 = 1;
        R1 = R2 = R3 = R4 = 0;
        if (C1 == 0)
            row_finder1();
        else if (C2 == 0)
            row_finder2();
        else if (C3 == 0)
            row_finder3();
        else if (C4 == 0)
            row_finder4();
    }
}

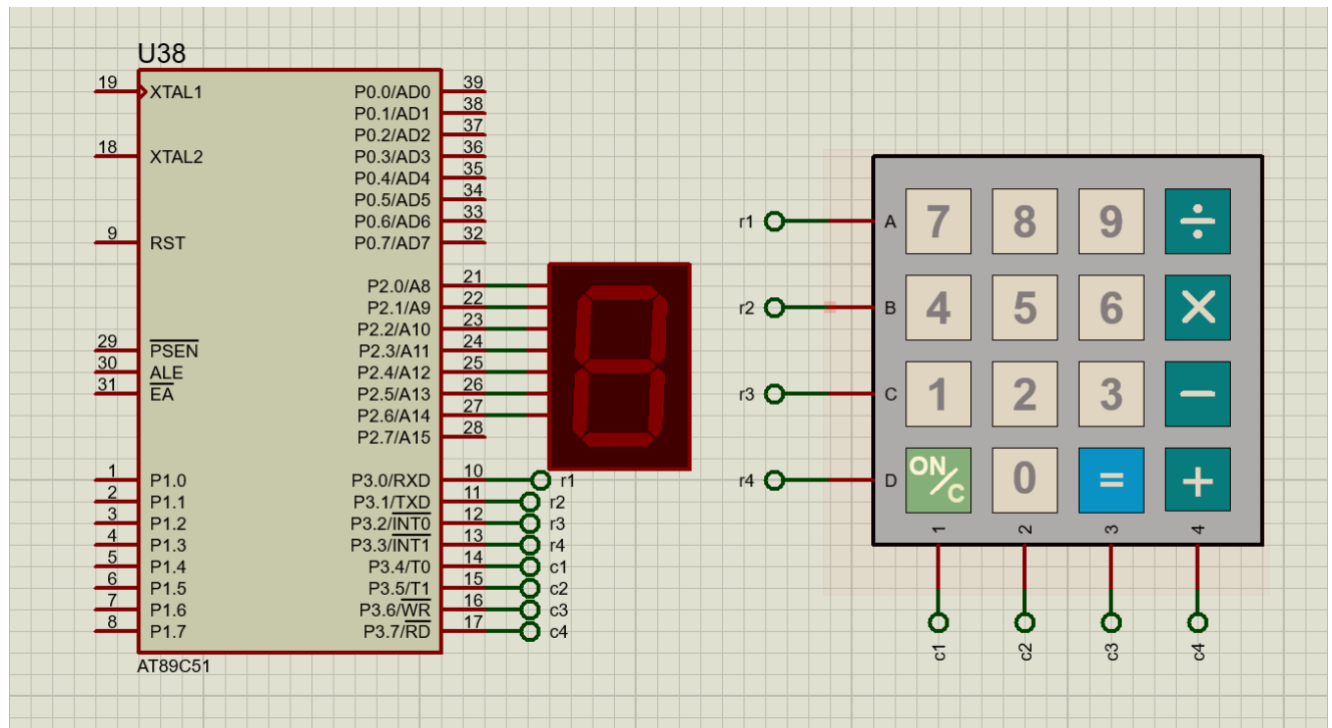
```

LAB 5

Program Objective: Display the pressed key of the hex keypad on 4 digit 7-Segment Display.

Component used: 8051 microprocessor, LEDs, Resistors

Connection Diagram:



Program Code:

```
#include <reg51.h>
```

```
sbit row1 = P3^0;
sbit row2 = P3^1;
sbit row3 = P3^2;
sbit row4 = P3^3;
```

```
sbit col1 = P3^4;
sbit col2 = P3^5;
sbit col3 = P3^6;
sbit col4 = P3^7;
```

```
void delay(int time) {
    int i, j;
    for(i = 0; i < time; i++) {
        for(j = 0; j < 100; j++);
    }
}
```

```
void main() {
    int num = 0;
    while(1) {
```



```

row1 = 0; row2 = 1; row3 = 1; row4 = 1;
if(col1 == 0) {
    num = 1;
    while(col1 == 0);
}
if(col2 == 0) {
    num = 2;
    while(col2 == 0);
}
if(col3 == 0) {
    num = 3;
    while(col3 == 0);
}
if(col4 == 0) {
    num = 10;
    while(col4 == 0);
}
row1 = 1; row2 = 0; row3 = 1; row4 = 1;
if(col1 == 0) {
    num = 4;
    while(col1 == 0);
}
if(col2 == 0) {
    num = 5;
    while(col2 == 0);
}
if(col3 == 0) {
    num = 6;
    while(col3 == 0);
}
if(col4 == 0) {
    num = 11;
    while(col4 == 0);
}
row1 = 1; row2 = 1; row3 = 0; row4 = 1;
if(col1 == 0) {
    num = 7;
    while(col1 == 0);
}
if(col2 == 0) {
    num = 8;
    while(col2 == 0);
}

```

```

    }
    if(col3 == 0) {
        num = 9;
        while(col3 == 0);
    }
    if(col4 == 0) {
        num = 12;
        while(col4 == 0);
    }
    row1 = 1; row2 = 1; row3 = 1; row4 = 0;
    if(col1 == 0) {
        num = 0;
        while(col1 == 0);
    }
    if(col2 == 0) {
        num = 15;
        while(col2 == 0);
    }
    if(col3 == 0) {
        num = 14;
        while(col3 == 0);
    }
    if(col4 == 0) {
        num = 13;
        while(col4 == 0);
    }
}

```

P2 = 0xFF; // Turn off all segments of the 7-segment display

delay(50); // Delay for a short period of time

```

switch(num) {
    case 0: P2 = 0xC0; break;
    case 1: P2 = 0xF9; break;
    case 2: P2 = 0xA4; break;
    case 3: P2 = 0xB0; break;
    case 4: P2 = 0x99; break;
    case 5: P2 = 0x92; break;
    case 6: P2 = 0x82; break;
    case 7: P2 = 0xF8; break;
    case 8: P2 = 0x80; break;
    case 9: P2 = 0x90; break;
    case 10: P2 = 0xF7; break;
    case 11: P2 = 0xBF; break;
}

```

```
        case 12: P2 = 0xDF; break;
        case 13: P2 = 0xE7; break;
        case 14: P2 = 0xDB; break;
        case 15: P2 = 0xEE; break;
        default: P2 = 0xFF; break;
    }
    delay(500); // Delay to display the number for some time
}
}
```

LAB 6

Program Objective: Implement an increasing counter and show on 4 digits 7-Segment Display.

Component used: 8051 microprocessor, LEDs, Resistors


```

sbit r2 = P3 ^ 2;
sbit r3 = P3 ^ 3;
sbit c0 = P3 ^ 4;
sbit c1 = P3 ^ 5;
sbit c2 = P3 ^ 6;
sbit c3 = P3 ^ 7;

```

```

void delay(int x){
int i,j;
for(i=0;i<x;i++){
    for(j =0;j<10000;j++);}
}

```

```

int main(){
while(1){

```

```

    int i,j;
        A=0; s=0;C=0;D = 0; E = 0 ; F= 0; G = 0 ;
        a=0; b=0;c=0;e = 0; e = 0 ; f= 0; g = 0 ;
        for(i=0;i<10;i++){
            switch(i){
            case 0:{a=1; b=1;c=1;d = 1; e = 1 ; f= 1; g = 0 ; break;}
            case 1:{a=0; b=1;c=1;d = 0; e = 0 ; f= 0; g = 0 ; break;}
            case 2:{a=1; b=1;c=0;d = 1; e = 1 ; f= 0; g = 1 ; break;}
            case 3:{a=1; b=1;c=1;d = 1; e = 0 ; f= 0; g = 1 ; break;}
            case 4:{a=0; b=1;c=1;d = 0; e = 0 ; f= 1; g = 1 ; break;}
            case 5:{a=1; b=0;c=1;d = 1; e = 0 ; f= 1; g = 1 ; break;}
            case 6:{a=1; b=0;c=1;d = 1; e = 1 ; f= 1; g = 1 ; break;}
            case 7:{a=1; b=1;c=1;d = 0; e = 0 ; f= 0; g = 0 ; break;}
            case 8:{a=1; b=1;c=1;d = 1; e = 1 ; f= 1; g = 1 ; break;}
            case 9:{a=1; b=1;c=1;d = 1; e = 0 ; f= 1; g = 1 ; break;}
            default:{a=0; b=0;c=0;d = 0; e = 0 ; f= 0; g = 1 ; break;}
            }
        for(j=0;j<10;j++){
            switch(j){
            case 0:{A=1; s=1;C=1;D = 1; E = 1 ; F= 1; G = 0 ; break;}
            case 1:{A=0; s=1;C=1;D = 0; E = 0 ; F= 0; G = 0 ; break;}
            case 2:{A=1; s=1;C=0;D = 1; E = 1 ; F= 0; G = 1 ; break;}
            case 3:{A=1; s=1;C=1;D = 1; E = 0 ; F= 0; G = 1 ; break;}
            case 4:{A=0; s=1;C=1;D = 0; E = 0 ; F= 1; G = 1 ; break;}
            case 5:{A=1; s=0;C=1;D = 1; E = 0 ; F= 1; G = 1 ; break;}
            case 6:{A=1; s=0;C=1;D = 1; E = 1 ; F= 1; G = 1 ; break;}

```

```

    case 7:{A=1; s=1;C=1;D = 0; E = 0 ; F= 0; G = 0 ; break;}
    case 8:{A=1; s=1;C=1;D = 1; E = 1 ; F= 1; G = 1 ; break;}
    case 9:{A=1; s=1;C=1;D = 1; E = 0 ; F= 1; G = 1 ; break;}
    default:{A=0; s=0;C=0;D = 0; E = 0 ; F= 0; G = 1 ; break;}
  }
  delay(2);    }
  delay(2);
}
}
return 0 ;
}

```

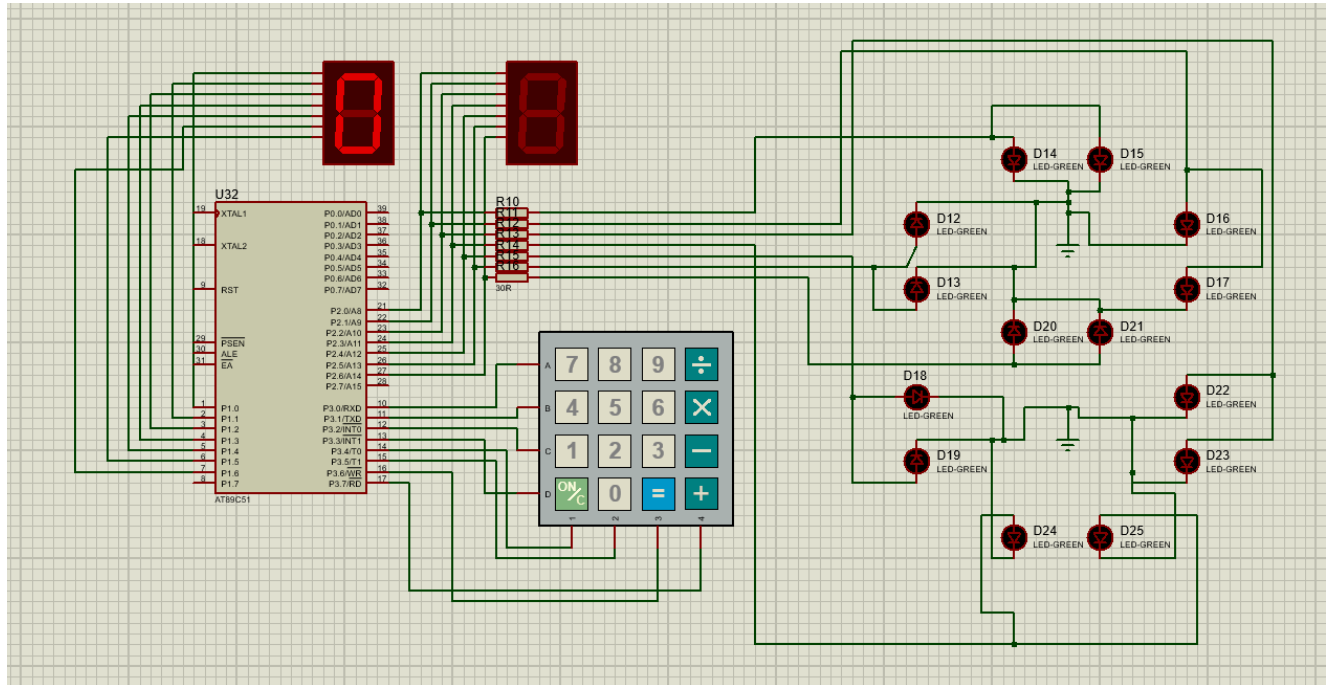
LAB 7

Program Objective: Set the counter using a hex keypad and

display on 7Segment.

Component used: 8051 microprocessor, LEDs, Resistors

Connection Diagram:



Program Code:

```
#include<reg51.h>
```

```
#include<math.h>
```

```
#define display_port P1 //Data pins connected to port 2 on microcontroller
```

```
sbit rs = P2 ^ 0; //RS pin connected to pin 2 of port 3
```

```
//sbit rw = P2^1; // RW pin connected to pin 3 of port 3
```

```
sbit e = P2 ^ 1; //E pin connected to pin 4 of port 3
```

```
sbit h = P3 ^ 0;
```

```
sbit i = P3 ^ 1;
```

```
sbit j = P3 ^ 2;
```

```
sbit k = P3 ^ 3;
```

```
sbit l = P3 ^ 4;
```

```
int a = 0, b = 0, c = 0, d = 0, xr = 0;
```

```
void delay(int x) {
```

```
    int i, j;
```

```

    for (i = 0; i < x; i++) {
        for (j = 0; j < 10000; j++);
    }
}

void msdelay(unsigned int time) // Function for creating delay in milliseconds.
{
    unsigned i, j;
    for (i = 0; i < time; i++)
        for (j = 0; j < 1275; j++);
}

void lcd_cmd(unsigned char command) //Function to send command instruction
to LCD
{
    display_port = command;
    rs = 0;
    e = 1;
    msdelay(1);
    e = 0;
}

void lcd_data(unsigned char disp_data) //Function to send display data to LCD
{
    display_port = disp_data;
    rs = 1;
    e = 1;
    msdelay(1);
    e = 0;
}

void lcd_init() //Function to prepare the LCD and get it ready
{
    lcd_cmd(0x38); // for using 2 lines and 5X7 matrix of LCD
    msdelay(10);
    lcd_cmd(0x0F); // turn display ON, cursor blinking
    msdelay(10);
    lcd_cmd(0x01); //clear screen
    msdelay(10);
    lcd_cmd(0x80); // bring cursor to position 1 of line 1
    msdelay(10);
}

void first() {
    char ar[16] = "A B C D X ";

```



```

char br[16] = "0 0 0 0 0 ";
int i;
for (i = 0; i < 15; i++) {
    lcd_data(ar[i]);
}
lcd_cmd(0xC0);
for (i = 0; i < 15; i++) {
    lcd_data(br[i]);
}
}
void updateLED(int n) {
    char br[16] = "0 0 0 0 0 ";
    int mz,i;
    lcd_cmd(0xC0);
    if (n == 1) {
        a = (a + 1) % 100;
    } else if (n == 2) {
        b = (b + 1) % 100;
    } else if (n == 3) {
        c = (c + 1) % 100;
    } else if (n == 4) {
        d = (d + 1) % 100;
    } else {
        xr = (xr + 1) % 100;
    }

    mz = a;
    br[1] = mz % 10 + '0';
    mz /= 10;
    br[0] = mz % 10 + '0';
    br[2] = ' ';
    mz = b;
    br[3] = mz % 10 + '0';
    mz /= 10;
    br[2] = mz % 10 + '0';
    br[4] = ' ';
    mz = c;
    br[6] = mz % 10 + '0';
    mz /= 10;
    br[5] = mz % 10 + '0';
    br[7] = ' ';
    mz = d;

```

```

    br[9] = mz % 10 + '0';
    mz /= 10;
    br[8] = mz % 10 + '0';
    br[10] = ' ';
    mz = xr;
    br[12] = mz % 10 + '0';
    mz /= 10;
    br[11] = mz % 10 + '0';
    br[13] = ' ';
    for (i = 0; i < 15; i++) {
        lcd_data(br[i]);
    }
}
int main() {
    int currBuzz = -1;
    lcd_init();
    first();
    h = 0;
    i = 0;
    j = 0;
    k = 0;
    l = 0;
    while (1) {
        if (h == 1 || i == 1 || j == 1 || k == 1) {
            if (h == 1) {
                if (currBuzz == -1) {
                    currBuzz = 1;
                    l = 1;
                    updateLED(1);
                    l = 0;
                    currBuzz = -1;
                }
                h = 0;
            } else if (i == 1) {
                if (currBuzz == -1) {
                    currBuzz = 2;
                    l = 1;
                    updateLED(2);
                    l = 0;
                    currBuzz = -1;
                }
                i = 0;
            }
        }
    }
}

```

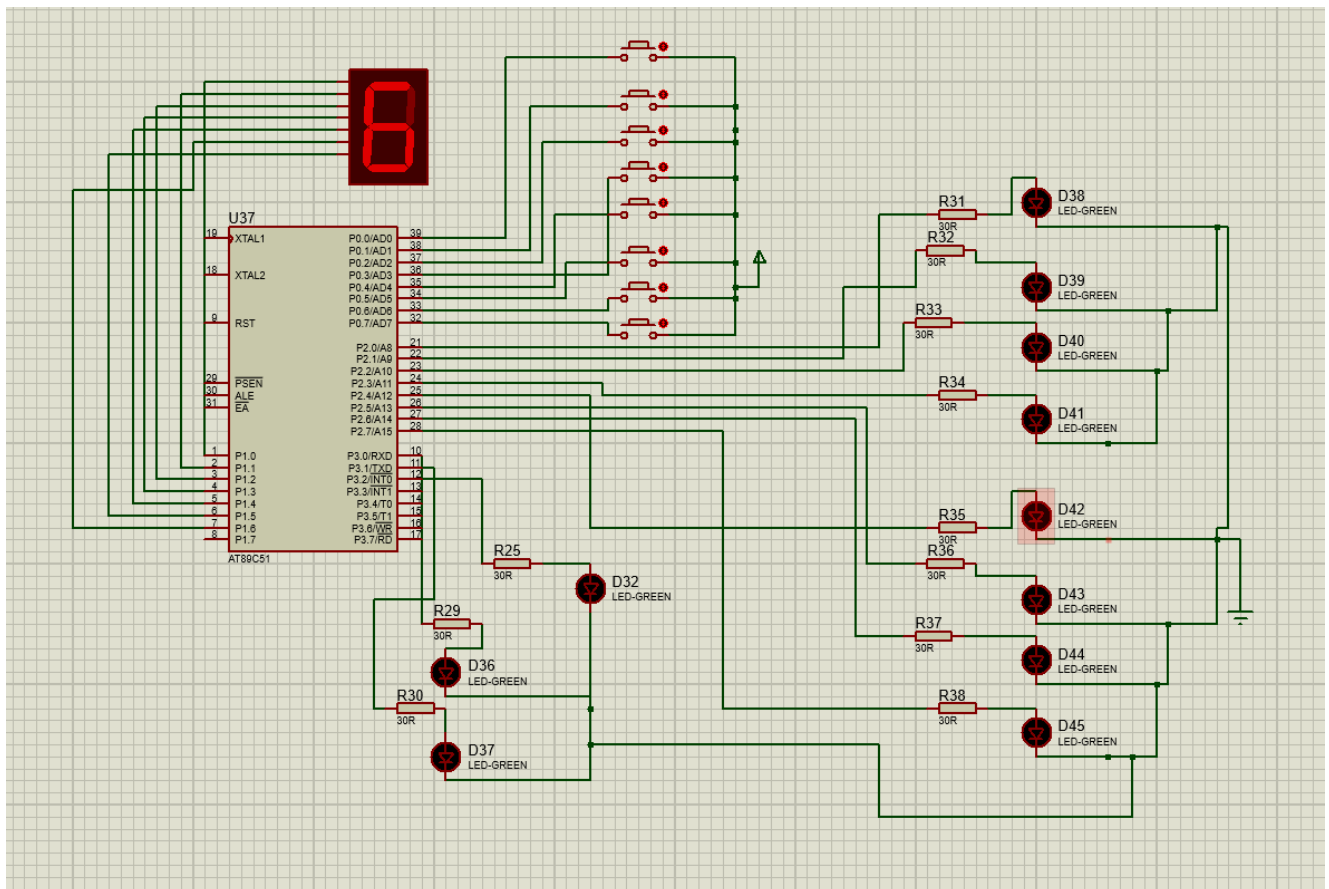
```
} else if (j == 1) {  
    if (currBuzz == -1) {  
        currBuzz = 3;  
        l = 1;  
        updateLED(3);  
        l = 0;  
        currBuzz = -1;  
    }  
    j = 0;  
} else if (k == 1) {  
    if (currBuzz == -1) {  
        currBuzz = 4;  
        l = 1;  
        updateLED(4);  
        l = 0;  
        currBuzz = -1;  
    }  
    k = 0;  
}  
delay(5);  
} else {  
    updateLED(5);  
}  
}  
return 0;  
}
```

LAB 8

Program Objective: Write a program to show the movement of Elevator.

Component used: 8051 microprocessor, LEDs, Resistors

Connection Diagram:



Program Code:

```
#include<reg51.h>
#include<math.h>
```

```
sbit h = P0 ^ 0;
sbit i = P0 ^ 1;
sbit j = P0 ^ 2;
sbit k = P0 ^ 3;
sbit l = P0 ^ 4;
sbit m = P0 ^ 5;
sbit n = P0 ^ 6;
sbit o = P0 ^ 7;
sbit p = P2 ^ 0;
sbit q = P2 ^ 1;
sbit r = P2 ^ 2;
sbit s = P2 ^ 3;
sbit t = P2 ^ 4;
sbit u = P2 ^ 5;
sbit v = P2 ^ 6;
sbit w = P2 ^ 7;
sbit U = P3 ^ 0;
sbit D = P3 ^ 1;
sbit E = P3 ^ 2;
sbit a = P1 ^ 0;
sbit b = P1 ^ 1;
sbit c = P1 ^ 2;
sbit d = P1 ^ 3;
sbit e = P1 ^ 4;
sbit f = P1 ^ 5;
sbit g = P1 ^ 6;
int arr[8] = {
    -1,
    -1,
    -1,
    -1,
    -1,
    -1,
    -1,
    -1
};
int z;
int currFloor = 0;
```

```

int lr = 0;
void delay(int x) {
    int i, j;
    for (i = 0; i < x; i++) {
        for (j = 0; j < 10000; j++);
    }
}
void blinkSevenSeg(int i) {
    a = 0;
    b = 0;
    c = 0;
    e = 0;
    e = 0;
    f = 0;
    g = 0;
    switch (i) {
    case 0: {
        a = 1;
        b = 1;
        c = 1;
        d = 1;
        e = 1;
        f = 1;
        g = 0;
        break;
    }
    case 1: {
        a = 0;
        b = 1;
        c = 1;
        d = 0;
        e = 0;
        f = 0;
        g = 0;
        break;
    }
    case 2: {
        a = 1;
        b = 1;
        c = 0;
        d = 1;
        e = 1;

```

```
f = 0;
g = 1;
break;
}
case 3: {
    a = 1;
    b = 1;
    c = 1;
    d = 1;
    e = 0;
    f = 0;
    g = 1;
    break;
}
case 4: {
    a = 0;
    b = 1;
    c = 1;
    d = 0;
    e = 0;
    f = 1;
    g = 1;
    break;
}
case 5: {
    a = 1;
    b = 0;
    c = 1;
    d = 1;
    e = 0;
    f = 1;
    g = 1;
    break;
}
case 6: {
    a = 1;
    b = 0;
    c = 1;
    d = 1;
    e = 1;
    f = 1;
    g = 1;
```

```
    break;
}
case 7: {
    a = 1;
    b = 1;
    c = 1;
    d = 0;
    e = 0;
    f = 0;
    g = 0;
    break;
}
case 8: {
    a = 1;
    b = 1;
    c = 1;
    d = 1;
    e = 1;
    f = 1;
    g = 1;
    break;
}
case 9: {
    a = 1;
    b = 1;
    c = 1;
    d = 1;
    e = 0;
    f = 1;
    g = 1;
    break;
}
default: {
    a = 0;
    b = 0;
    c = 0;
    d = 0;
    e = 0;
    f = 0;
    g = 1;
    break;
}
```



```

    }
    delay(10);
}
void goFrom(int start, int end) {
    int z;
    if (start < end) {
        U = 1;
        delay(5);
        for (z = start + 1; z <= end; z++) {
            blinkSevenSeg(z);
            delay(5);
        }
        U = 0;
    } else {
        D = 1;
        delay(5);
        for (z = start - 1; z >= end; z--) {
            blinkSevenSeg(z);
            delay(5);
        }
        D = 0;
    }
}
void on(int aa) {
    switch (aa) {
        case 0:
            w = 1;
            break;
        case 1:
            v = 1;
            break;
        case 2:
            u = 1;
            break;
        case 3:
            t = 1;
            break;
        case 4:
            s = 1;
            break;
        case 5:
            r = 1;

```

```

        break;
    case 6:
        q = 1;
        break;
    case 7:
        p = 1;
        break;
    }
}
void off(int aa) {
    switch (aa) {
    case 0:
        w = 0;
        break;
    case 1:
        v = 0;
        break;
    case 2:
        u = 0;
        break;
    case 3:
        t = 0;
        break;
    case 4:
        s = 0;
        break;
    case 5:
        r = 0;
        break;
    case 6:
        q = 0;
        break;
    case 7:
        p = 0;
        break;
    }
}
void execute() {
    int ii;
    for (ii = 0; ii < lr; ii++) {
        if (currFloor == arr[ii]) {
            blinkSevenSeg(currFloor);

```

```

    } else {
        goFrom(currFloor, arr[ii]);
        currFloor = arr[ii];
    }
    off(arr[ii]);
}
lr=0;
}
void update() {
    E = 1;
    z = 1000000000;
    while (z--) {
        if (h == 1) {
            on(7);
            arr[lr] = 7;
            lr++;
            h = 0;
        } else if (i == 1) {
            on(6);
            arr[lr] = 6;
            lr++;
            i = 0;
        } else if (j == 1) {
            on(5);
            arr[lr] = 5;
            lr++;
            j = 0;
        } else if (k == 1) {
            on(4);
            arr[lr] = 4;
            lr++;
            k = 0;
        } else if (l == 1) {
            on(3);
            arr[lr] = 3;
            lr++;
            l = 0;
        } else if (m == 1) {
            on(2);
            arr[lr] = 2;
            lr++;
            m = 0;
        }
    }
}

```

```

    } else if (n == 1) {
        on(1);
        arr[lr] = 1;
        lr++;
        n = 0;
    } else if (o == 1) {
        on(0);
        arr[lr] = 0;
        lr++;
        o = 0;
    }
}
E = 0;

}
int main() {

    blinkSevenSeg(currFloor);
    h = 0;
    i = 0;
    j = 0;
    k = 0;
    l = 0;
    m = 0;
    n = 0;
    o = 0;
    U = 0;
    D = 0, p = 0, q = 0, r = 0, s = 0, t = 0, u = 0, v = 0, w = 0, E = 0;
    while (1) {
        update();
        if(lr!=0){
            execute();
        }
    }
    return 0;
}

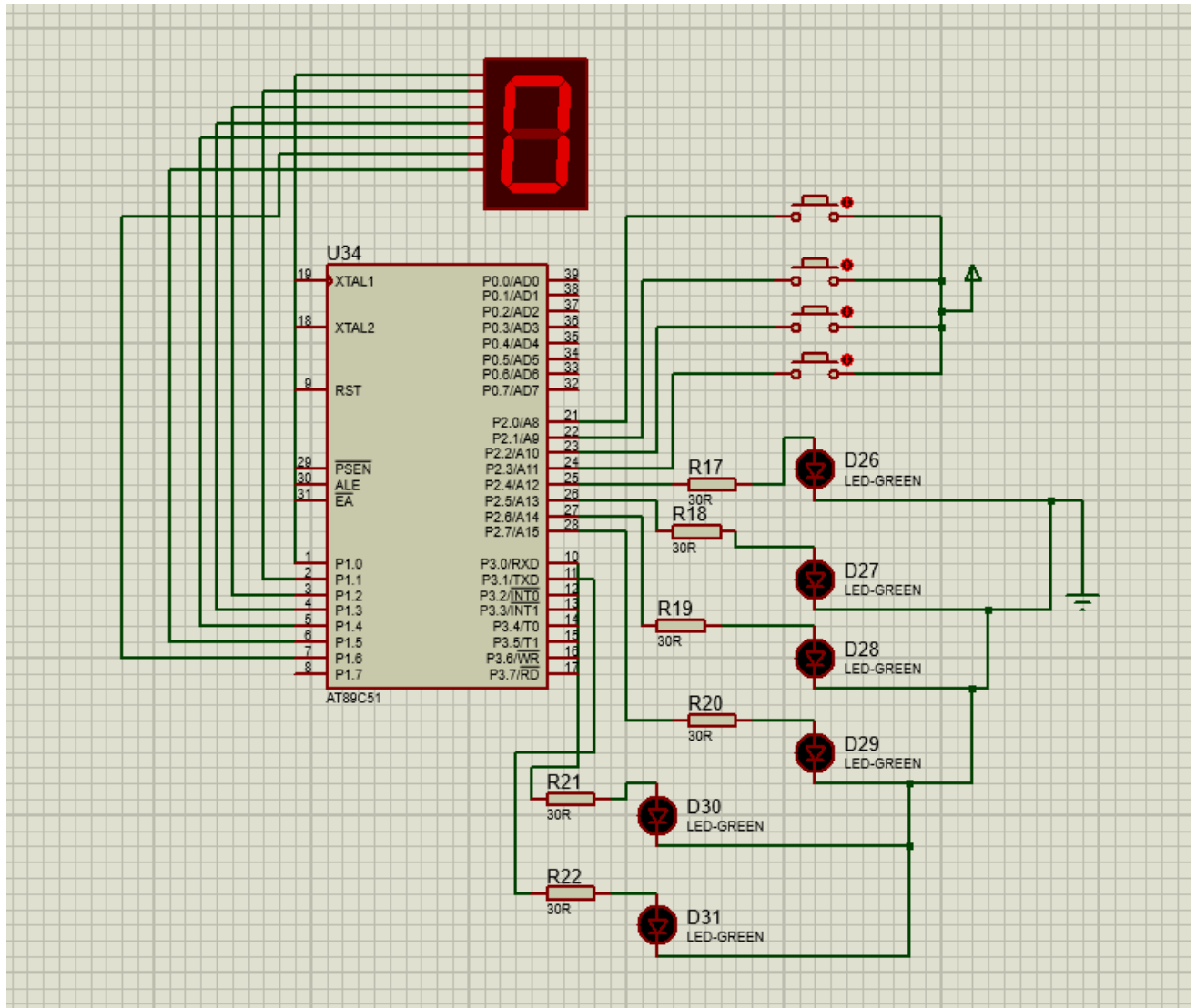
```

LAB 9

Program Objective: Write a program for Quiz-answer setup using Relay Buzzer and LCD.

Component used: 8051 microprocessor, LEDs, Resistors

Connection Diagram:



Program Code:

```
#include<reg51.h>
#include<math.h>
```

```
sbit a = P1 ^ 0;
sbit b = P1 ^ 1;
sbit c = P1 ^ 2;
sbit d = P1 ^ 3;
sbit e = P1 ^ 4;
sbit f = P1 ^ 5;
sbit g = P1 ^ 6;
sbit h = P2 ^ 0;
sbit i = P2 ^ 1;
sbit j = P2 ^ 2;
```

```

sbit k = P2 ^ 3;
sbit l = P2 ^ 4;
void delay(int x) {
    int i, j;
    for (i = 0; i < x; i++) {
        for (j = 0; j < 10000; j++);
    }
}
void blinkSevenSeg(int i) {
    a = 0;
    b = 0;
    c = 0;
    e = 0;
    e = 0;
    f = 0;
    g = 0;
    switch (i) {
    case 0: {
        a = 1;
        b = 1;
        c = 1;
        d = 1;
        e = 1;
        f = 1;
        g = 0;
        break;
    }
    case 1: {
        a = 0;
        b = 1;
        c = 1;
        d = 0;
        e = 0;
        f = 0;
        g = 0;
        break;
    }
    case 2: {
        a = 1;
        b = 1;
        c = 0;
        d = 1;

```

```
e = 1;
f = 0;
g = 1;
break;
}
case 3: {
    a = 1;
    b = 1;
    c = 1;
    d = 1;
    e = 0;
    f = 0;
    g = 1;
    break;
}
case 4: {
    a = 0;
    b = 1;
    c = 1;
    d = 0;
    e = 0;
    f = 1;
    g = 1;
    break;
}
case 5: {
    a = 1;
    b = 0;
    c = 1;
    d = 1;
    e = 0;
    f = 1;
    g = 1;
    break;
}
case 6: {
    a = 1;
    b = 0;
    c = 1;
    d = 1;
    e = 1;
    f = 1;
```



```
    g = 1;
    break;
}
case 7: {
    a = 1;
    b = 1;
    c = 1;
    d = 0;
    e = 0;
    f = 0;
    g = 0;
    break;
}
case 8: {
    a = 1;
    b = 1;
    c = 1;
    d = 1;
    e = 1;
    f = 1;
    g = 1;
    break;
}
case 9: {
    a = 1;
    b = 1;
    c = 1;
    d = 1;
    e = 0;
    f = 1;
    g = 1;
    break;
}
default: {
    a = 0;
    b = 0;
    c = 0;
    d = 0;
    e = 0;
    f = 0;
    g = 1;
    break;
}
```

```

    }
    }
    delay(20);
}
int main() {
    int currBuzz = -1;

    blinkSevenSeg(0);
    h = 0;
    i = 0;
    j = 0;
    k = 0;
    l = 0;
    while (1) {
    if(h==1||i==1||j==1||k==1){
        if (h == 1) {
            if (currBuzz == -1) {
                currBuzz = 1;
                l = 1;
                blinkSevenSeg(1);
                l = 0;
                currBuzz = -1;
            }
            h = 0;
        } else if (i == 1) {
            if (currBuzz == -1) {
                currBuzz = 2;
                l = 1;
                blinkSevenSeg(2);
                l = 0;
                currBuzz = -1;
            }
            i = 0;
        } else if (j == 1) {
            if (currBuzz == -1) {
                currBuzz = 3;
                l = 1;
                blinkSevenSeg(3);
                l = 0;
                currBuzz = -1;
            }
            j = 0;
        }
    }
}

```

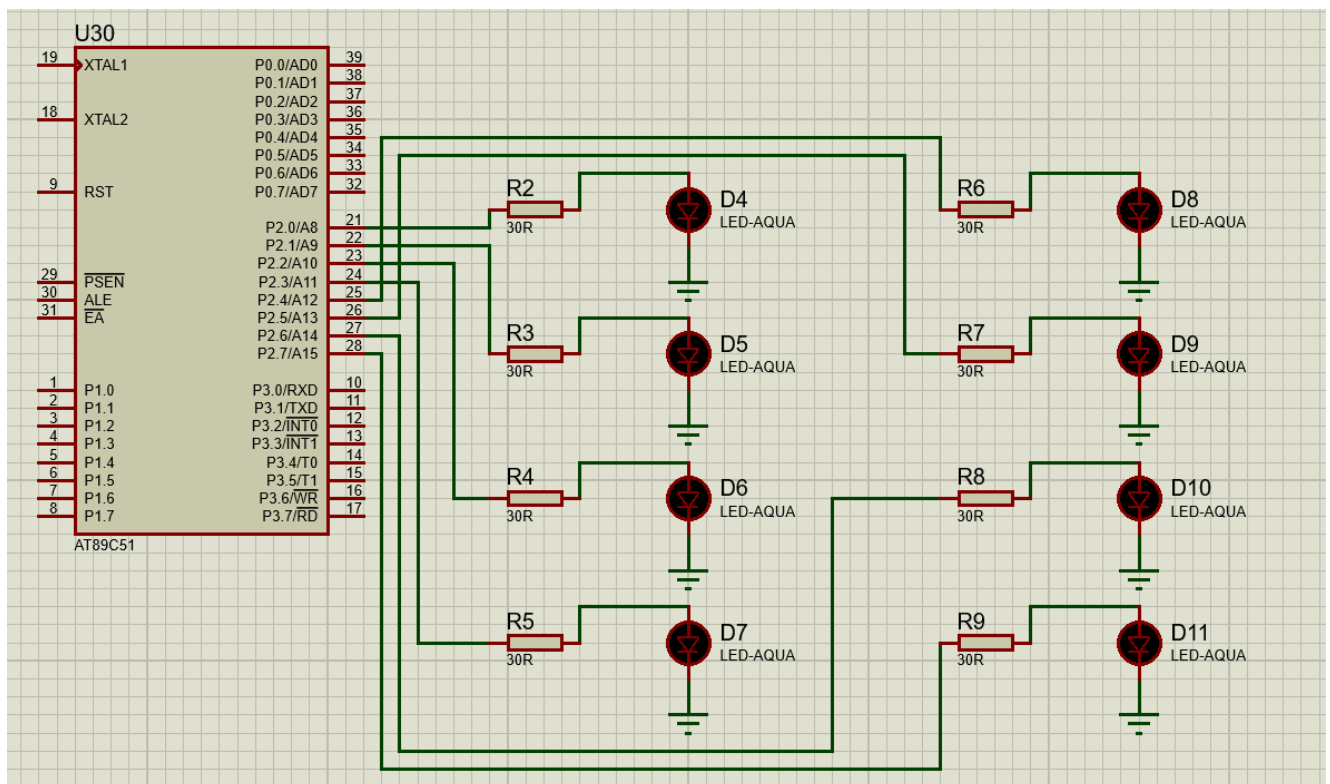
```
} else if (k == 1) {  
    if (currBuzz == -1) {  
        currBuzz = 4;  
        l = 1;  
        blinkSevenSeg(4);  
        l = 0;  
        currBuzz = -1;  
    }  
    k = 0;  
}  
    delay(5);  
}  
    else{  
        blinkSevenSeg(10);  
    }  
}  
return 0;  
}
```

LAB 10

Program Objective: Display date and time on LCD using RTC section.

Component used: 8051 microprocessor, LEDs, Resistors

Connection Diagram:



Program Code:

```
// write a program to blink the led in rotation
#include <reg51.h>
#define LED P2

void delay(int);
void main() {
    while (1) {
```

```

int i = 1, j;
for (j = 1; j <= 7; j++) {
    int K = i, r = 9 - j;
    while (r--) {
        LED = K;
        delay(7);
        LED = 0;
        delay(2);
        K <<= 1;
    }
    i = i * 2 + 1;
}
}
}
void delay(int x) {
    int i, j;
    for (i = 0; i < x; i++) {
        for (j = 0; j < 10000; j++) {
        }
    }
}
}

```

