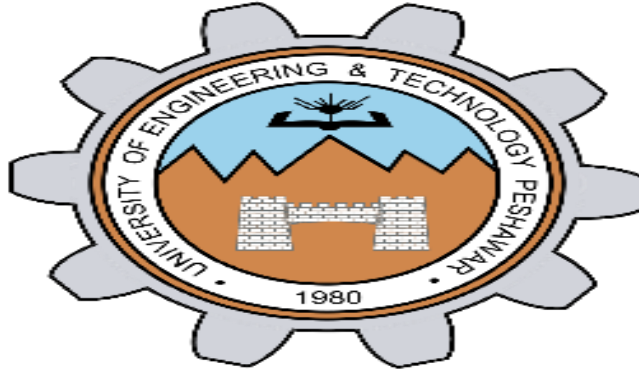**MICROPROCESSOR BASED SYSTEM DESIGN**

TASK 6

Spring 2022 CSE307

MBSD

Submitted by: **Zain ul Abideen**

Registration No. : **19pwcse1796**

Class Section: **A**

"On my honor, as student of University of Engineering and Technology, I have neither given nor received unauthorized assistance on this academic work."

Student Signature: _____

Submitted to:

Dr. Bilal Habib

Sunday, June 12, 2022

Task:

In this assignment, you are required to design a calculator that should be able to do the following operations, Addition, Subtraction, Multiplication and Division.

Code:

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


//Function declarations
void cct_init(void);
void delay(int); void
lcdinit(void); void
writecmd(int); void
writedata(char);
void Return(void); char
READ_SWITCHES(void); char
get_key(void);

//*******************
//Pin description
/*
P2 is data bus
P3.7 is RS
P3.6 is E
P1.0 to P1.3 are keypad row outputs
P1.4 to P1.6 are keypad column inputs
*/
//*******************
// Define Pins
//******************* sbit
RowA = P1^0;    //RowA sbit
RowB = P1^1;    //RowB sbit
RowC = P1^2;    //RowC
sbit RowD = P1^3;    //RowD

sbit C1  = P1^4;    //Column1 sbit
C2  = P1^5;    //Column2 sbit C3
= P1^6;    //Column3
sbit C4  = P1^7;        //Column4
```

```c
sbit E   = P3^6;    //E pin for LCD
sbit RS  = P3^7;    //RS pin for LCD

unsigned int key_count = 0, result_int; char
array[3], result_char;
// ************************************************************
// Main program
//
int main(void)
{
  char key;          // key char for keeping record of pressed key

  cct_init();        // Make input and output pins as required
lcdinit();           // Initilize LCD

  writecmd(0x95);
       writedata('T');                    //write
writedata('i');                    //write    writedata('m');
//write        writedata('e');                    //write
writedata(' ');                    //write    writedata('2');
//write        writedata('3');                    //write
       writedata(':');                    //write
       writedata('5');
writedata('9');
writedata(':');   writedata('2');
writedata('7');
  writecmd(0xd5);
       writedata('D');                    //write
writedata('a');                    //write    writedata('t');
//write        writedata('e');                    //write
writedata(' ');                    //write    writedata('3');
//write        writedata('1');                    //write
       writedata('/');                    //write
writedata('1');          writedata('2');
writedata('/');   writedata('2');
writedata('0');          writedata('2');
   writedata('1');                    //write

  writecmd(0x80);
  while(1)
  {
   key = get_key();     // Get pressed key
        if (key == 'C')
                {
```

```c
                writecmd(0x01);        // Clear screen
array[0]= array[1]= array[2]='n';
                result_int = 0; key_count = 0;
                }
        else if(key == '=')
                {
                writedata(key);        // Echo the key pressed to LCD
                        if(array[1] == 'x')
                                result_int = (array[0] -'0')  * (array[2] - '0');
        else if(array[1] == '/')
                                result_int = (array[0] -'0')  / (array[2] - '0');
        else if(array[1] == '+')
                                result_int = (array[0] -'0')  + (array[2] - '0');
        else if(array[1] == '-')
                                result_int = (array[0] -'0')  - (array[2] - '0');
  if(result_int > 9) //if the result is greater than 9 then we need to store each digit in an array
                        {
                           int result[2];        //Delare an array
                           result[0] = result_int/10; //Store the digit at ten's place at index 0
            result[1] = result_int%10; //Store the digit at unit's place at index 1
result_char = (char)result[0] + '0'; //Convert the integer at index 0 to character
                           writedata(result_char );        // Echo the key pressed to LCD
    result_char = (char)result[1] + '0'; //Convert the integer at index 1 to character
                           writedata(result_char );        // Echo the key pressed to LCD
                        }
                        else
                        {
                          result_char = (char)result_int + '0';
                          writedata(result_char );        // Echo the key pressed to LCD
                        }
                }
        else
                writedata(key);        // Echo the key pressed to LCD
  }
}


void cct_init(void)
{
      P0 = 0x00;  //not used
 P1 = 0xf0;  //used for generating outputs and taking inputs from Keypad
      P2 = 0x00;  //used as data port for LCD
      P3 = 0x00;  //used for RS and E
}

void delay(int a)
```

```c
{
  int i;
  for(i=0;i<a;i++);  //null statement
}

void writedata(char t)
{
  RS = 1;          // This is data
  P2 = t;          //Data transfer
E  = 1;          // => E = 1
delay(150);
  E  = 0;          // => E = 0
  delay(150);
}


void writecmd(int z)
{
  RS = 0;          // This is command
  P2 = z;          //Data transfer
E  = 1;          // => E = 1
delay(150);
  E  = 0;          // => E = 0
  delay(150);
}

void lcdinit(void)
{
 ///////////// Reset process from datasheet /////////
   delay(15000);
writecmd(0x30);
delay(4500);
writecmd(0x30);
delay(300);   writecmd(0x30);
   delay(650);
 //////////////////////////////////////////////////
  writecmd(0x38);   //function set
  writecmd(0x0c);   //display on,cursor off,blink off
writecmd(0x01);   //clear display
  writecmd(0x06);   //entry mode, set increment
}

void Return(void)    //Return to 0 location on LCD
{
writecmd(0x02);
delay(1500);
```

```
}

char READ_SWITCHES(void)
{
 RowA = 0; RowB = 1; RowC = 1; RowD = 1;  //Test Row A

        if (C1 == 0) { delay(10000); while (C1==0); return '7'; }
if (C2 == 0){ delay(10000); while (C2==0); return '8'; }      if (C3
== 0) { delay(10000); while (C3==0); return '9'; }
        if (C4 == 0) { delay(10000); while (C4==0); return '/'; }

 RowA = 1; RowB = 0; RowC = 1; RowD = 1;  //Test Row B

        if (C1 == 0) { delay(10000); while (C1==0); return '4'; }
if (C2 == 0) { delay(10000); while (C2==0); return '5'; }      if (C3
== 0) { delay(10000); while (C3==0); return '6'; }    if (C4 == 0)
{ delay(10000); while (C4==0); return 'x'; }

 RowA = 1; RowB = 1; RowC = 0; RowD = 1;  //Test Row C

        if (C1 == 0) { delay(10000); while (C1==0); return '1'; }
if (C2 == 0) { delay(10000); while (C2==0); return '2'; }      if (C3
== 0) { delay(10000); while (C3==0); return '3'; }
        if (C4 == 0) { delay(10000); while (C4==0); return '-'; }

 RowA = 1; RowB = 1; RowC = 1; RowD = 0;  //Test Row D

        if (C1 == 0) { delay(10000); while (C1==0); return 'C'; }
if (C2 == 0) { delay(10000); while (C2==0); return '0'; }      if (C3
== 0) { delay(10000); while (C3==0); return '='; }    if (C4 == 0)
{ delay(10000); while (C4==0); return '+'; }
        return 'n';                // Means no key has been pressed
}

char get_key(void)        //get key from user
{
        char key = 'n';           //assume no key pressed

        while(key=='n')           //wait untill a key is pressed
 key = READ_SWITCHES();  //scan the keys again and again  array[key_count++]= key;

        return key;              //when key pressed then return its value }
```
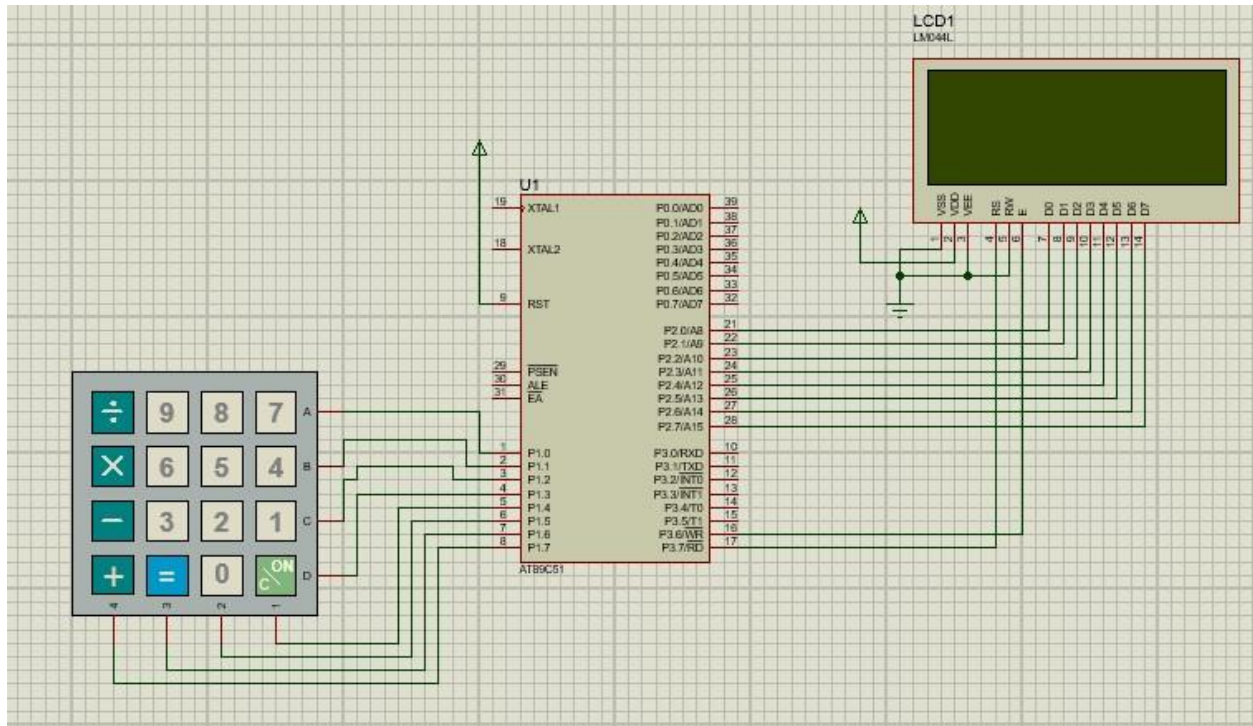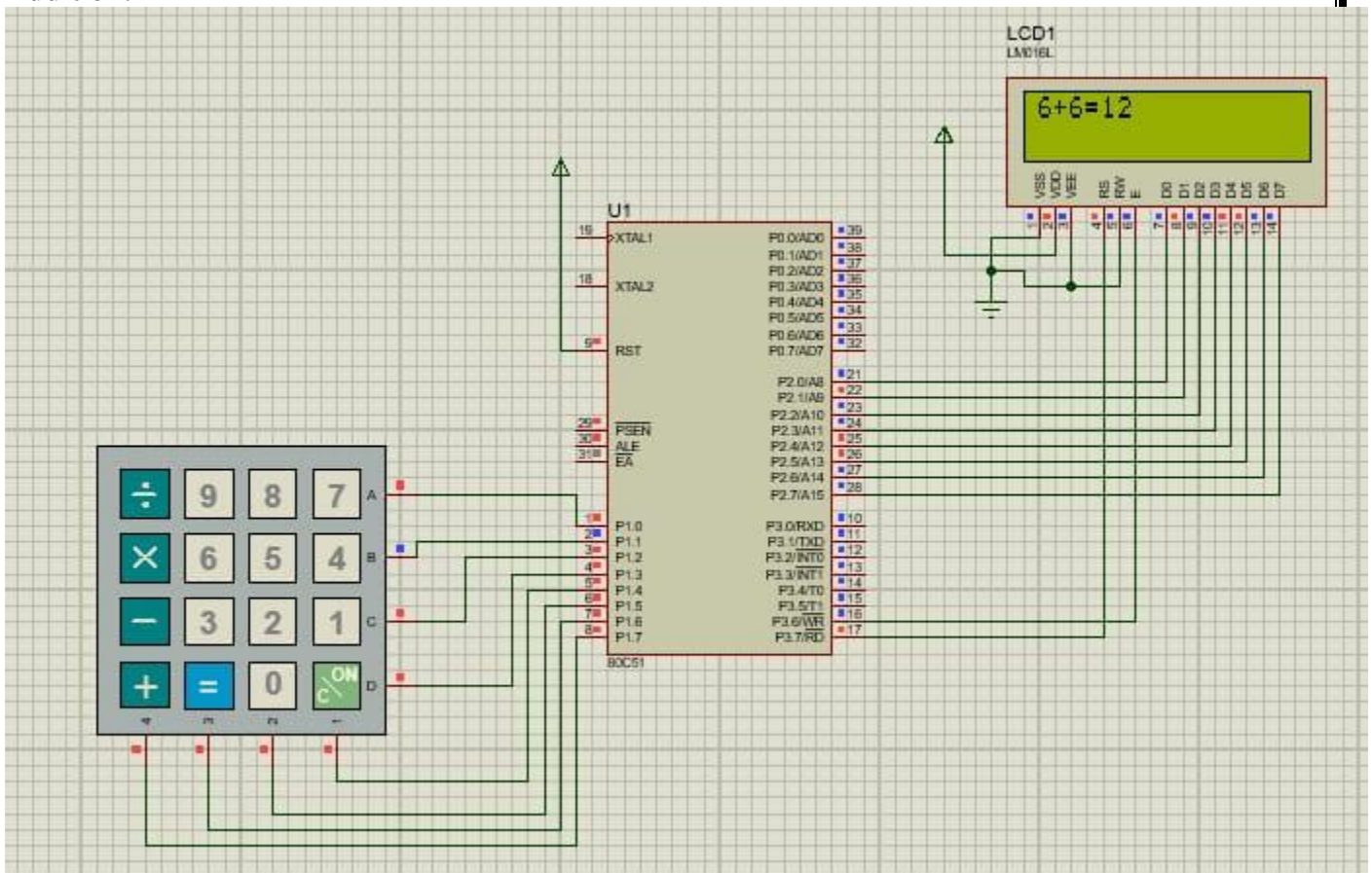
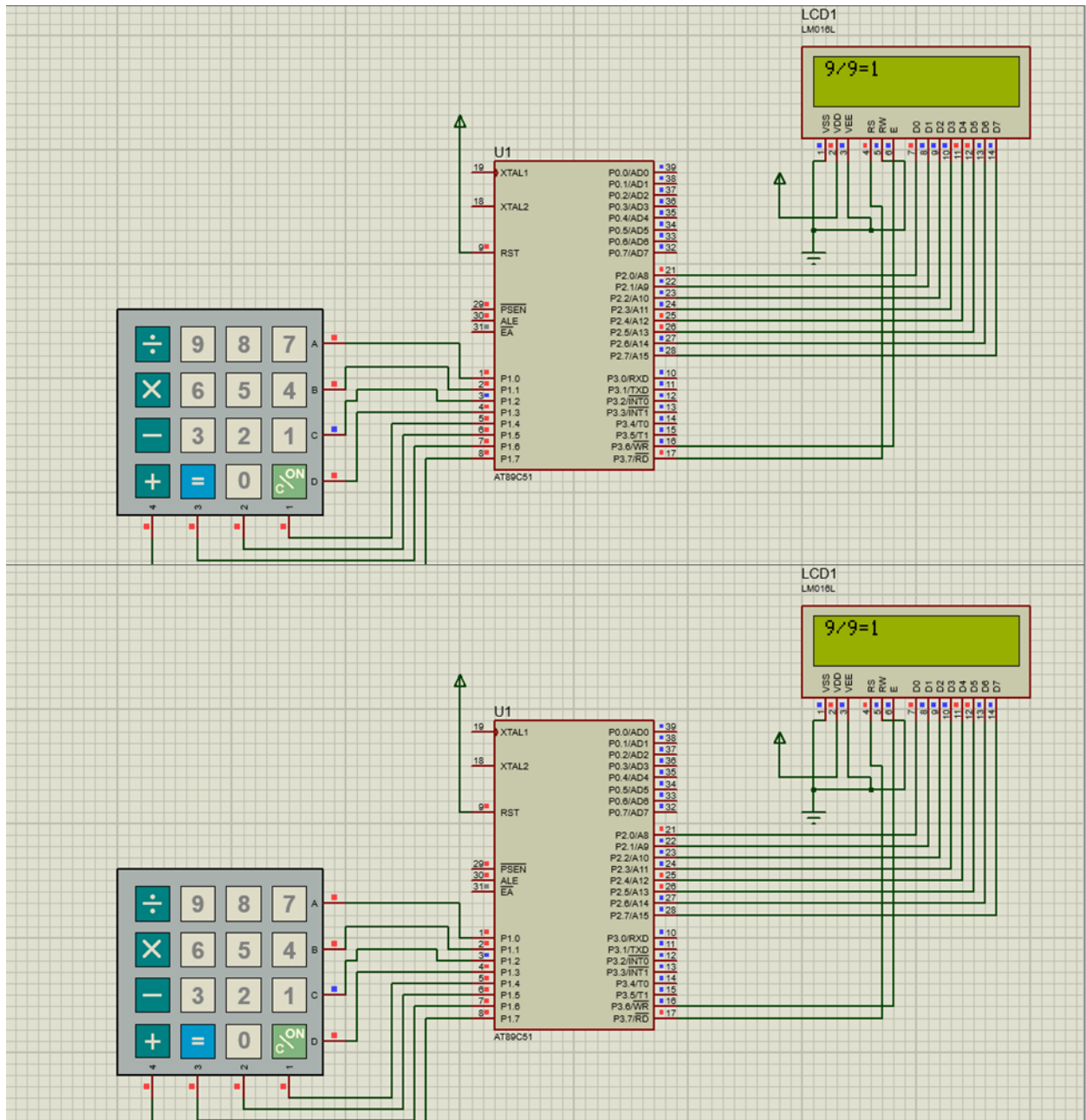**Output / Graphs / Plots / Results:**
Schematic:

Addition:

Division:

Multiplication: