Department of Electronics And Communication

Project Report on

"design a simple calculator"

TEAM MEMBERS :

1. Praveen Magadum
2. Prajwal Kamble
3. Prashant Patil
4. Pratik Patil
5. Prajwal Hosakoti

Department of Electronics And Communication

Project Title :

Interfacing keypad and LCD with 8051 to design a simple calculator

Description :

- **Micrcontroller** : A microcontroller is a compact integrated circuit designed to govern a specific operation in an embedded system.

- A microcontroller processes data given to it's input pins using it's CPU and gives output via output pins.

- The calculator we are going to design in this post is quite basic calculator, it will only perform 4 tasks, which are as follows:

- When you press the **(+)** button then it will add the two digits. For example, you want to add 2 and 3 then you need to press **2 + 2 =** these four buttons in sequence and when you press the = button it will automatically will give you the sum.

- When you press **(-)** button it will subtract the two digits like **3 - 2 =** and it will give you the result.

- When you press **(x)** button it will multiply the two digits.

- When you press the **(/)** button it will simply divide the two digits.

- Whenever you press the **(=)** button, it will give you the output depending on the function you used before and if you press (=) in the start then it will give "Wrong Input".

- Finally, there's **(ON/C)** button on the Calculator, when you press this it will simply reset the code and will clear the LCD

Simulated Circuit :

- The Proteus Simulation of this Calculator with 8051 Microcontroller is same as we used for Interfacing of Keypad with 8051 Microcontroller and is shown in below figure:
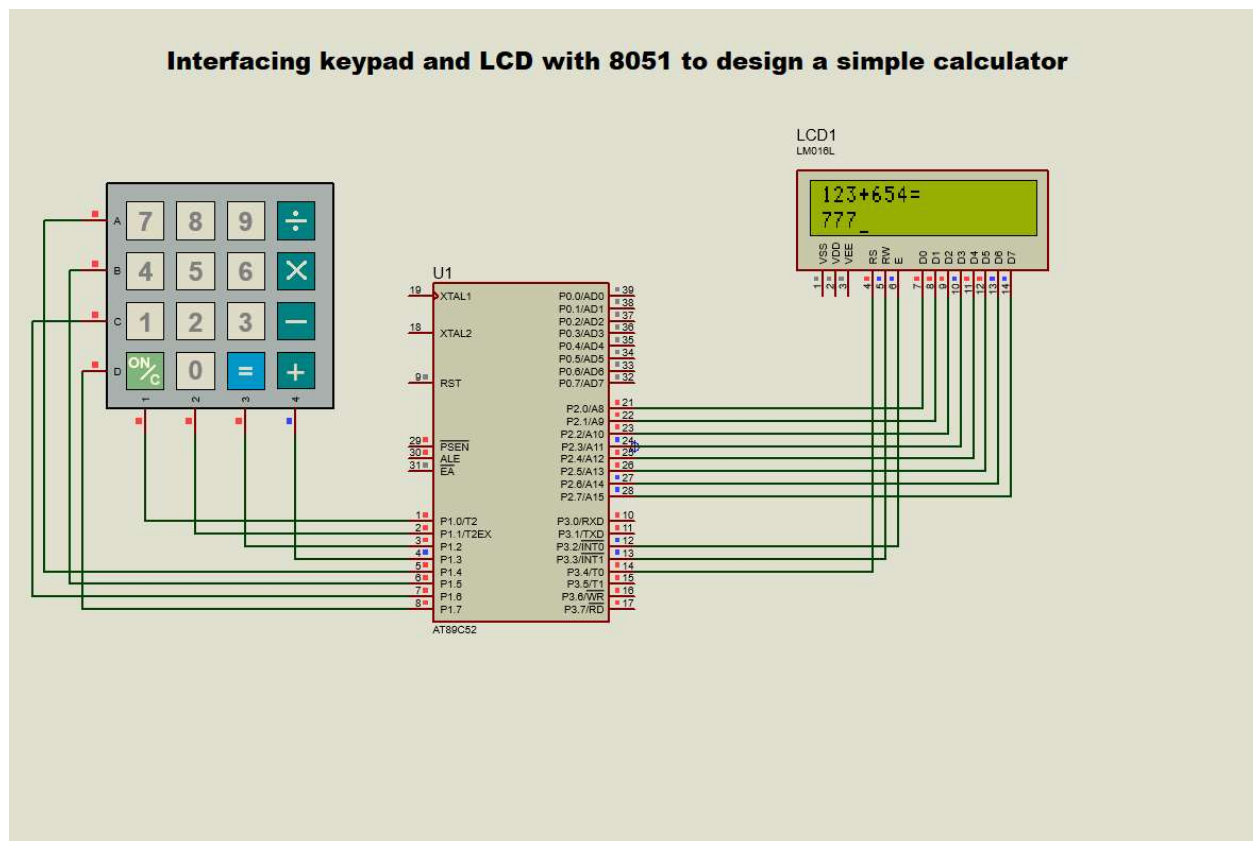
Programming code :

```c
#include <regx52.h>
#include <stdio.h>
void init();
void command(unsigned int);
void write_data(unsigned char);
void delay(unsigned char );
void msDelay(unsigned int);
char process_key(int,int,int,int,char,char,char,char);
void num_generator(char,int *);
void write_result(int,char);
sbit EN = P3^2; // Enable LCD
sbit RW = P3^3; // Read Write
sbit RS = P3^4; // Register Select
void main()
{
int i,j,* operand ,operand1=0,operand2=0,result=0;
int i_arr[4][4] = {{0,1,1,1},{1,0,1,1},{1,1,0,1},{1,1,1,0}};
char pressed_key,operator1='a',negative='N';
char c_arr[4][4] = {{'7','4','1','o'},{'8','5','2','0'},{'9','6','3','='},{'/','*','-
','+'}},error[10]={'M','A','T','H',' ','E','R','R','O','R'};
P1 = 0xff;
operand = &operand1;
while(1)
{
for(i=0;i<4;i++)
 {
pressed_key =
process_key(i_arr[i][0],i_arr[i][1],i_arr[i][2],i_arr[i][3],c_arr[i][0],c_arr[i][1],c_arr[i][2],
c_arr[i][3]);
  if(pressed_key=='/'||pressed_key=='*'||pressed_key=='-

'||pressed_key=='+')
  {
   operand = &operand2;
```

```
operator1 = pressed_key;
  }
  if((pressed_key != 'Z')&&(pressed_key!= '=')&&(pressed_key!= '/')&&(pressed_key!=
'*')&&(pressed_key!= '-')&&(pressed_key!= '+'))
  {
   num_generator(pressed_key,operand);
  }
  // Make a function generate_result
  if(pressed_key == '=')
  {
   // Make a function generate_result in which all the mess below this line will be
processed..
   if(operator1== '*')
   result = operand1*operand2;

   if(operator1== '/')
   if(operand2==0)
   {
    command(0xC0);
    for(j=0;j<10;j++)
    write_data(error[j]);
    msDelay(500);
    init();
   }
   else
    result = operand1/operand2;

   if(operator1== '-')
   {
   if (operand1>operand2)

  result = operand1-operand2;
   else
   {
   result = operand2-operand1;
   command(0xC0);
   negative = 'Y';
```

```
  }
  }

  if(operator1== '+')
  result = operand1+operand2;

  write_result(result,negative);
  }
 }
}
}
// LCD initilaizer function
void init()
{
 delay(3500);
 command(0x38);
 delay(3500);
 command(0x38);
 delay(3500);
 command(0x38);
 delay(350);
 command(0x38);
 command(0x1C);
 command(0x0E);
 command(0x06);
 command(0x01);
delay(3500);
 command(0x00);
 main();
}
// Sends different commands to LCD
void command(unsigned int comm)
{
 RW = 0;
 RS = 0;

 P2 = comm;
 EN = 1;
```

```c
 delay(3500);
 EN = 0;
}
// Process and check the press of key
char process_key(int a,int b,int c,int d,char A,char B,char C,char D)
{
 char ch = 'Z';
 P1_0=a;
 P1_1=b;
 P1_2=c;
 P1_3=d;
 if(P1_4==0)
 {
  ch = A;
 }
 if(P1_5==0)
 {
  ch = B;
 }

 if(P1_6==0)
 {
  ch = C;
 }
 if(P1_7==0)
 {
  if (D == 'o')
  init();
  else
   ch = D;
 }
 if(ch!= 'Z')
 {
write_data(ch);
  msDelay(100);
 }
 return ch;
}
```

```
// Used to write data in char on LCD
void write_data(unsigned char ch)
{
  RW = 0;
  RS = 1;
  P2 = ch;
  EN = 1;
  delay(3500);
  EN = 0;
}

void num_generator(char ch,int *operand)
{
int digit;

digit = ch - '0';
*operand = digit + (*operand*10);
}

void write_result(int num,char neg)
{
 int i=0,j,rem;
 char rev_num[20];
 command(0xC0);
 if(neg == 'Y')
 {
 write_data('-');
 }
 do
   {
   rem = num%10;

 num = num /10;
   rev_num[i] = (char)rem+'0';
   i++;
   }while(num>0);
 for(j=i-1;j>=0;j--)
```

Department of Electronics And Communication

```c
 {
   RW = 0;
  RS = 1;
  P2 = rev_num[j];
  EN = 1;
  delay(3500);
  EN = 0;
  }
}

//Function for generation of delay
void delay(unsigned char c)
{
unsigned int i;
unsigned char j;
for(i=0;i<=3;i++)
{
for(j=0;j<=c;j++);
}
}
// Generates delay in milli seconds
void msDelay(unsigned int time)
{
 TL0 = 0xEF;
 TH0 = 0xAF;
 TR0 = 1;
 while(time--)
 {
  while(TF0 == 0);
  TF0 = 0;
  TL0 = 0xEF;
  TH0 = 0xAF;
 }
 TR0 = 0;
}
```