Experiment 13 **KEYPAD INTERFACING USING 8051 USING KEIL AND PROTEUS**

**AIM:**

Write an assembly language program for Keypad Interfacing Using 8051 using Keil and Proteus

**SOFTWARE REQUIRED:**

* Keil software 5.
* Proteus 8 software.

**KEIL PROCEDURE:**

1.open the software ,click on project and open new version project.

2.create a new project file

3. enter AT89C51

4.click no

5.click ctrl n and write code

6. open project and click target build

7. open target build and open source file and ADD, CLOSE

8. click target build

9. next debug start and stop

10.open peripherals and select port 2

11. run the program in debug

12. open project and click optional properties and in that give output as hexa file.

**PROTEUS PROCEDURE:**

**1. Open proteus by clicking run as administrator.**

**2. Open new project and enter the file name.**

**3. Click next, next, next and finish.**

**4. click P symbol and search keyword and place it’**

**AT89C51**

**KEYPAD-PHONE**

**LM016L**

**Using terminal click ground and place it two times.**

**5. Connecting pin number 21-1 up to 28-8**

**6. Connecting the ground to pin 20.**

**7. Connecting the pin no. 19-13 to the wire between pin 20 and GND.**

**8. Connecting pin 19 to the C1 and 18 to C2.**

**9. Connect C1 and C2**

**10. Connect another GND to the wire between C1 and C2.**

**11. Connect the crystal pin no.1 to C1 and pin 2 to C2.**

**12. Give input in crystal as 16MHz.**

**13. Give input to C1 and C2 as 33pF.**

**14. Give input to AT89C51 as HEX file.**

**PROGRAM:**

#include<reg51.h>

#define display\_port P2 //Data pins connected to port 2 on microcontroller

sbit rs = P3^2; //RS pin connected to pin 2 of port 3

sbit rw = P3^3; // RW pin connected to pin 3 of port 3

sbit e = P3^4; //E pin connected to pin 4 of port 3

sbit C4 = P1^0; // Connecting keypad to Port 1

sbit C3 = P1^1;

sbit C2 = P1^2;

sbit C1 = P1^3;

sbit R4 = P1^4;

sbit R3 = P1^5;

sbit R2 = P1^6;

sbit R1 = P1^7;

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;

rw=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;

rw=0;

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(0x81); // bring cursor to position 1 of line 1

msdelay(10);

}

void row\_finder1() //Function for finding the row for column 1

{

R1=R2=R3=R4=1;

C1=C2=C3=C4=0;

if(R1==0)

lcd\_data('1');

if(R2==0)

lcd\_data('4');

if(R3==0)

lcd\_data('7');

if(R4==0)

lcd\_data('\*');

}

void row\_finder2() //Function for finding the row for column 2

{

R1=R2=R3=R4=1;

C1=C2=C3=C4=0;

if(R1==0)

lcd\_data('2');

if(R2==0)

lcd\_data('5');

if(R3==0)

lcd\_data('8');

if(R4==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)

lcd\_data('3');

if(R2==0)

lcd\_data('6');

if(R3==0)

lcd\_data('9');

if(R4==0)

lcd\_data('#');

}

void row\_finder4() //Function for finding the row for column 4

{

R1=R2=R3=R4=1;

C1=C2=C3=C4=0;

if(R1==0)

lcd\_data('A');

if(R2==0)

lcd\_data('B');

if(R3==0)

lcd\_data('C');

if(R4==0)

lcd\_data('D');

}

void main()

{

lcd\_init();

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();

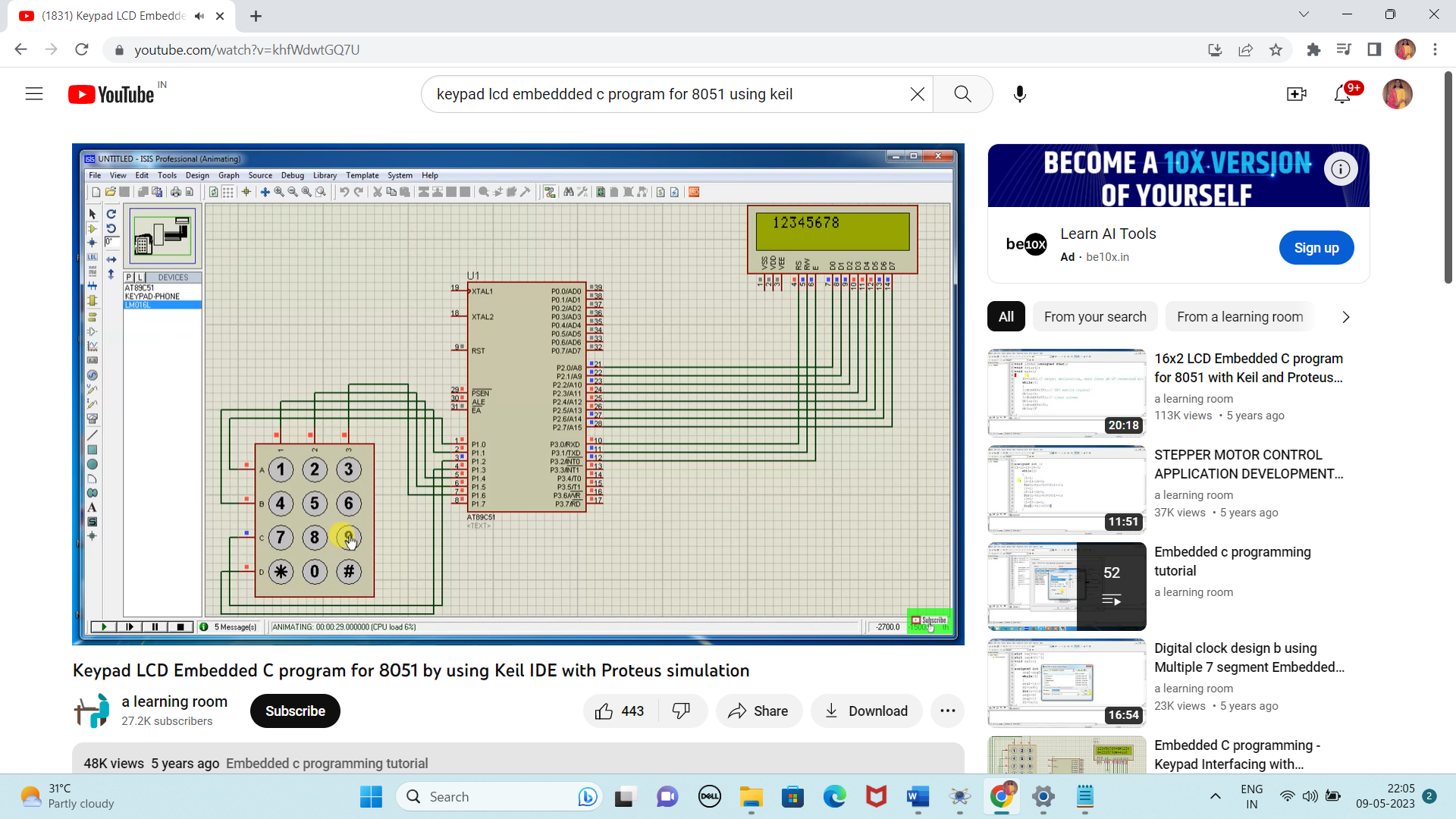
else if(C4==0)

row\_finder4();

}

}

**Circuit Diagram:**



**RESULT:**

Thus the program has been successfully verified and executed.