Experiment 15 **AUTOMATIC DOOR LOCKING SYSTEM USING 8051 USING KEIL AND PROTEUS**

**AIM:**

Write an assembly language program for Automatic Door Locking System 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 Type the code

6. Open project and click Build target

7. Open Build target and open source file and ADD, CLOSE

8. Click build target

9. Next debug start and stop

10. Open peripherals and select port 2

11. Now run the program in Debug

12. Open project and click optional properties and in that give output as hex file.

13. Create hex file.

**PROTEUS PROCEDURE:**

* Open proteus by clicking run as administrator.
* Open new project and enter the file name.
* Click next, next, next and finish.
* Click P symbol and search keyword and place the required components

The components required are:

* AT89C51
* L293D
* MOTOR DC
* BUTTON
* Using terminal click ground and place it two times.
* As per declaration so do your connection

5. The button which is connected with [1]

6. And the one end is connected with IO and the other end is connected with groud

7. Touch the button and name it in string as Sensor here

8. You need to add driver IC with two inputs these two which as connected with 2&3

9. Motor is connected with correct corresponding output lines

10. The both GND should connect with both lines and enable lines also

11. Give direct power connection and you have to place the control and should connected with power line

12. And at last ground pin connected with GND line

13. Give input to AT89C51 as HEX file.

14. Start the simulation process

**PROGRAM:**

#include<reg52.h>

sbit r0=P2^0; sbit r1=P2^1; sbit r2=P2^2; sbit r3=P2^3; sbit c0=P2^5; sbit c1=P2^6;

sbit c2=P2^7; sbit en=P3^6; sbit rs=P3^5; sbit rw=P3^7; sbit lock=P3^0;

char t1[]="Enter PIN:";

char t2[]="Access Granted";

char t3[]="Access Denied";

char pin[]="1234";

char pinEntered[4];

unsigned int m = 0;

unsigned int flag = 0;

void delay(unsigned int no)

{

unsigned int i,j;

for(j=0;j<=no;j++) for(i=0;i<=10;i++);

}

void lcdcmd(unsigned int command){

P1=command; rw=0; rs=0; en=0; delay(1000); en=1; delay(1000); en=0;

}

void lcddata(char data1)

{

P1=data1; rw=0; rs=1; en=0; delay(1000); en=1; delay(1000); en=0;

}

void lcdint(){

lcdcmd(0x30); delay(1000); lcdcmd(0x30); delay(1000); lcdcmd(0x30); delay(1000);

lcdcmd(0x30); delay(1000); lcdcmd(0x30); delay(1000); lcdcmd(0x38); delay(1000);

lcdcmd(0x01); delay(1000); lcdcmd(0x0F); delay(1000); lcdcmd(0x80); delay(1000);

}

char keypad()

{

char c='a';

while(c!='s'){

r0=0;r1=1;r2=1;r3=1;

if(c0==0){lcddata('1');P0=0xF0;delay(10000);c='s';return '1';}

if(c1==0){lcddata('2');P0=0xF0;delay(10000);c='s';return '2';}

if(c2==0){lcddata('3');P0=0xF0;delay(10000);c='s';return '3';}

r0=1;r1=0;r2=1;r3=1;

if(c0==0){lcddata('4');P0=0xF0;delay(10000);c='s';return '4';}

if(c1==0){lcddata('5');P0=0xF0;delay(10000);c='s';return '5';}

if(c2==0){lcddata('6');P0=0xF0;delay(10000);c='s';return '6';}

r0=1;r1=1;r2=0;r3=1;

if(c0==0){lcddata('7');P0=0xF0;delay(10000);c='s';return '7';}

if(c1==0){lcddata('8');P0=0xF0;delay(10000);c='s';return '8';}

if(c2==0){lcddata('9');P0=0xF0;delay(10000);c='s';return '9';}

r0=1;r1=1;r2=1;r3=0;

//if(c0==0){lcddata('\*');P0=0xF0;delay(10000);c='s';return '1';}

if(c1==0){lcddata('0');P0=0xF0;delay(10000);c='s';return '0';}

// if(c2==0){lcddata('#');P0=0xF0;delay(10000);c='s';return '1';}

}

}

void main()

{

unsigned int i=0;

P1=0x00; P2=0xF0; P3=0x00;

lcdint();

while(1){

i=0;

while(t1[i]!='\0')

{

lcddata(t1[i]);

i++;

}

lock=0;

lcdcmd(0xC0);

for(i=0;i<=3;i++)

pinEntered[i] =keypad();

i=0;

lcdcmd(0x01);

if(pinEntered[0]==pin[0])

{ i++;

if(pinEntered[1]==pin[1])

{

if(pinEntered[2]==pin[2])

{

if(pinEntered[3]==pin[3])

{

lock=1; i=0;

while(t2[i]!='\0')

{

lcddata(t2[i]);

i++;

}i=0;

}

}

}

}else{ i=0;

while(t3[i]!='\0')

{

lcddata(t3[i]);

i++;

} i=0; }

if(i!=0)

{ //Download Full Code For Better Understanding

i=0;

while(t3[i]!='\0')

{

lcddata(t3[i]);

i++;

}

}

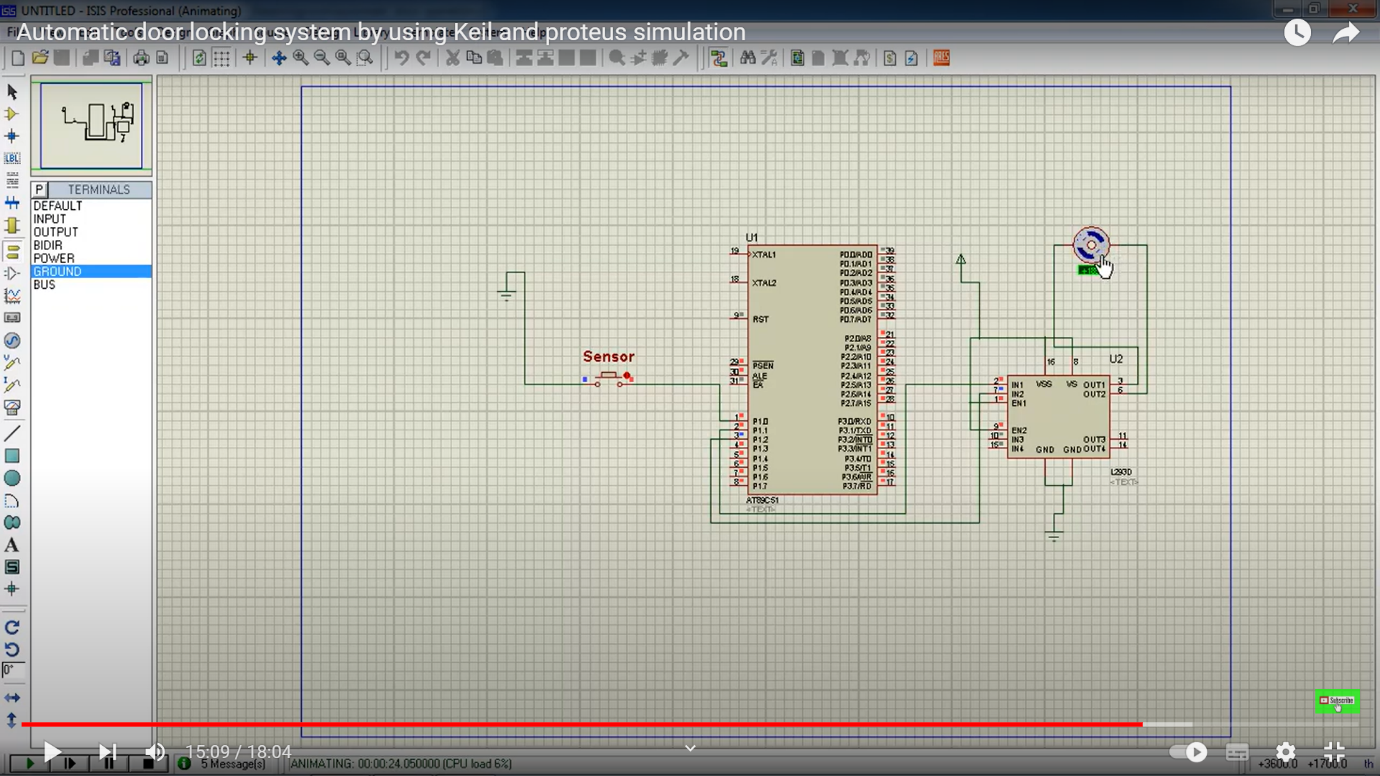
delay(1000000);

lcdcmd(0x01);

lcdcmd(0x80);

i=0;

**CIRCUIT DIAGRAM:**



**RESULT:**

Thus the program has been successfully verified and executed.