

LEROTHOLI POLYTECHNIC

**SCHOOL OF ENGINEERING**

**AND**

**TECHNOLOGY**

|  |  |
| --- | --- |
| **Student Number:** | 201900510CSE |

|  |  |
| --- | --- |
| **Surname & Other Names:** | **Maphalla Thato** |

|  |  |
| --- | --- |
| **Program:** | **Diploma: Computer Systems Engineering** |

|  |  |
| --- | --- |
| **Subject Name:** | **Microcontroller Applications III** |

|  |  |
| --- | --- |
| **Subject Code:** | **MRC321** |

|  |  |
| --- | --- |
| **Assignment Number:** | 6 |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Due Date:** | **0** | **6** | **0** | **6** | **2** | **0** | **2** | **3** |

|  |  |
| --- | --- |
| **Lecturer:** | **Raliete** |

**Declaration of own work**

I hereby declare that this assignment is my own work and that it has not been copied from any other person or document.

........................................... ..….............................

Signature Date

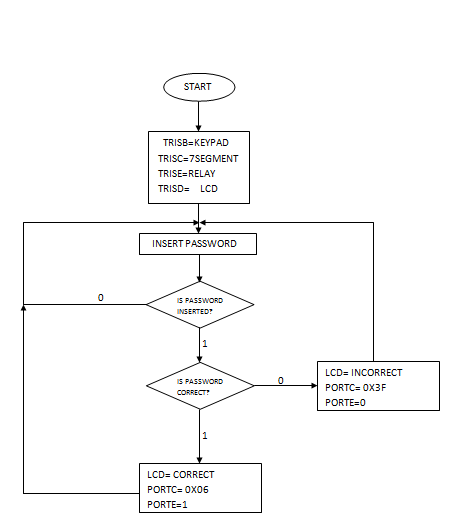
**Question**

**Practical Assignment 6**

Design a **Password operated door**, the door must only be open when the password is correct.

Use a keypad at the input and LCD on the output together with the bulb to demonstrate access and denial processes. Generate a 4-digot unique password, however if the password is correct or not the LCD should always display a suitable message. The bulb should only switch-on when the password is correct as to demonstrate the door opening process**. Note: 230v should be tapped from the plug.**

**Flow Chart**



**Code**

**unsigned char kp,kp2,kp3,kp4; //VARIABLES DEACLERATION**

**char keypadPort at PORTB; //CONNECTION THE KEYPAD A PORTB**

**sbit LCD\_EN at RD5\_bit;**

**sbit LCD\_RS at RD4\_bit;**

**sbit LCD\_D4 at RD0\_bit;**

**sbit LCD\_D5 at RD1\_bit;**

**sbit LCD\_D6 at RD2\_bit;**

**sbit LCD\_D7 at RD3\_bit;**

**sbit LCD\_RS\_Direction at TRISD4\_bit;**

**sbit LCD\_EN\_Direction at TRISD5\_bit;**

**sbit LCD\_D4\_Direction at TRISD0\_bit;**

**sbit LCD\_D5\_Direction at TRISD1\_bit;**

**sbit LCD\_D6\_Direction at TRISD2\_bit;**

**sbit LCD\_D7\_Direction at TRISD3\_bit;**

**void delay(){delay\_ms(500);}**

**void initialization() //FUNTION DEACLERATION**

**{**

**ANSEL=ANSELH=0X00; //CONFIGURING ALL THE OI'S AS DIGITAL PINS**

**TRISB=0XFF; //KEYPAD**

**TRISD=0X00; //LCD**

**TRISC=0X00; //MAKING PORTC AN OUTPUT**

**PORTC=0X00; //TURNING PORTC OFF**

**LCD\_INIT(); //LCD DEACLERATION**

**LCD\_CMD(\_LCD\_CURSOR\_OFF); //TURNING THE LCD CURSOR OFF**

**LCD\_OUT(1,2,"ENTER PASSWORD"); //DISPLAYING A TEXT ON THE LCD**

**KEYPAD\_INIT(); //KEYPAD DEACLERATION**

**}**

**//void delay(){delay\_ms(50);}**

**void keyPadInput ()//FUNCTION DEACLERATION**

**{**

**kp = 0;**

**do {**

**kp = keypad\_key\_click (); //ASSIGNING THE kp VALUE TO KEYPAD KEY**

**}**

**while( kp == 0); //WAITIONG FOR A KEYPAD KEY TO BE PRESSED**

**}**

**void keyPadInput2 () //FUNCTION DEACLERATION**

**{**

**kp2 = 0;**

**do {**

**kp2 = keypad\_key\_click (); //ASSIGNING THE kp VALUE TO KEYPAD KEY**

**}**

**while( kp2 == 0); //WAITIONG FOR A KEYPAD KEY TO BE PRESSED**

**}**

**void keyPadInput3 () //FUNCTION DEACLERATION**

**{**

**kp3 = 0;**

**do {**

**kp3 = keypad\_key\_click (); //ASSIGNING THE kp VALUE TO KEYPAD KEY**

**}**

**while( kp3 == 0); //WAITIONG FOR A KEYPAD KEY TO BE PRESSED**

**}**

**void keyPadInput4 () //FUNCTION DEACLERATION**

**{**

**kp4 = 0;**

**do {**

**kp4 = keypad\_key\_click (); //ASSIGNING THE kp VALUE TO KEYPAD KEY**

**}**

**while( kp4 == 0); //WAITIONG FOR A KEYPAD KEY TO BE PRESSED**

**}**

**void main() //STATE OF THE MAIN FUNCTION**

**{**

**initialization(); //FUNCTION CALL**

**while (1) // STATE OF THE WHILE LOOP**

**{**

**keyPadInput(); //FUNCTION CALL**

**keyPadInput2(); //FUNCTION CALL**

**keyPadInput3(); //FUNCTION CALL**

**keyPadInput4(); //FUNCTION CALL**

**switch (kp) //CHECKING THE STATE OF THE INPUT**

**{**

**case 2: kp = 50;break; //2**

**}**

**switch (kp2) //CHECKING THE STATE OF THE INPUT**

**{**

**case 6: kp2 = 53;break; //5**

**}**

**switch (kp3) //CHECKING THE STATE OF THE INPUT**

**{**

**case 7: kp3 = 54;break; //6**

**}**

**switch (kp4) //CHECKING THE STATE OF THE INPUT**

**{**

**case 9: kp4 = 55;break; //7**

**}**

**//PASSWORD = 2567**

**if(kp==50 && kp2==53 && kp3==54 && kp4==55 ) //CHECKING THE STATE OF THE INPUT**

**{**

**lcd\_cmd(\_lcd\_clear); //CLEARING THE LCD**

**Lcd\_Out(1,1,"CORRECT PASSWORD"); //WRITING A TEXT ON THE LCD**

**Lcd\_Out(2,4,"DOOR OPEN"); //WRITING A TEXT ON THE LCD**

**PORTC.RC5=1; //TURNING PIN RC5 ONN**

**}**

**else**

**{**

**lcd\_cmd(\_lcd\_clear); //CLEARING THE LCD**

**Lcd\_Out(1,2,"WRONG PASSWORD"); //WRITING A TEXT ON THE LCD**

**Lcd\_Cmd(\_Lcd\_Clear); //CLEARING THE LCD**

**Lcd\_Out(1,1,"ENTER CORRECT"); //WRITING A TEXT ON THE LCD**

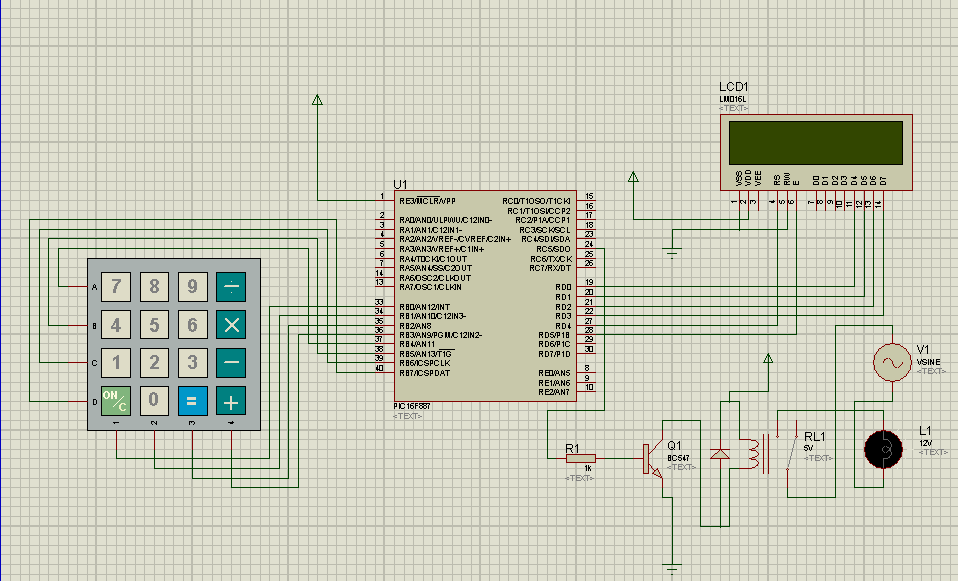
**Lcd\_Out(2,5,"PASSWORD"); //WRITING A TEXT ON THE LCD**

**PORTC.RC5=0; //TURNING PIN RC5 OFF**

**}**

**} //END OF THE WHILE LOOP**

**} //END OF THE MAIN FUNCTION**

**Circuit Diagram**