Embedded Systems

6th june 2022

Smart Home

Team Members:

Eslam Elmowafy Nawara

Toka Mahmoud Bedeer

Salma Khaled Elsayed

Shrouk Mahmoud Hejazi

Farha Emad Mohammed

Mona Ahmed Nafea

Mahetab Mohammed Saad

Under supervision:

Dr/ Mohammed Moawad

Abstract

Here are two functions of a smart home, the user first is supposed to enter the right password using a **keypad** to get into the home and the **LCD** displays if it's the wrong or right one and **two LEDs** Red for the wrong password and Green for the Right password which then opens **the door** for seconds and closes after seconds in the opposite direction.

The LCD also displays the Temperature by using **LM25** "temperature sensor" which controls the fan speed and the **heater.**

Objectives and Introduction

Our objective is to make use of most of the concepts we have this semester which is:

- Power Supplies
- LCD
- Keypad
- Motors
- ADC "Temperature sensor"
- PWM "The fan speed"

Components:

- PIC16F887
- LM3
- Power Supplies
- LCD 16X2
- KeyPad
- CD Motor
- Push Button
- Crystal 8MHz
- Lithium battery
- Jumpers
- LEDs
- Transistor 2n2222
- H-bridge l298n
- Resistors
- Capacitors 22pF
- Breadboard.
- FAN.
- Regulator 7805.
- Diod

Specifications:

- First, The LCD (16 x 2) is put on the home door to display ‘Enter Pass’ and the temperature.

- A keypad is put on the home door so the user can enter his password.

- If the user entered his password correctly, then the LCD displays ‘Welcome’, a green LED lights up, the door is opened for 500 ms, and after the user enters the door is closed and the green LED lights down.

- If not, then the LCD displays ‘Wrong Password’ and a red LED lights up for 500 ms.

- The temperature sensor (LM35) reads the temperature and displays it on the LCD.

- If the temperature is less than 30, the heater is turned on.

- If it ranges between [30: 32[, then the fan works at 80% of its velocity.

- If it ranges between [32: 34[, then the fan works at 90% of its velocity.

- If it is greater than 34, then the fan works at all of its velocity.

Design & Implementation

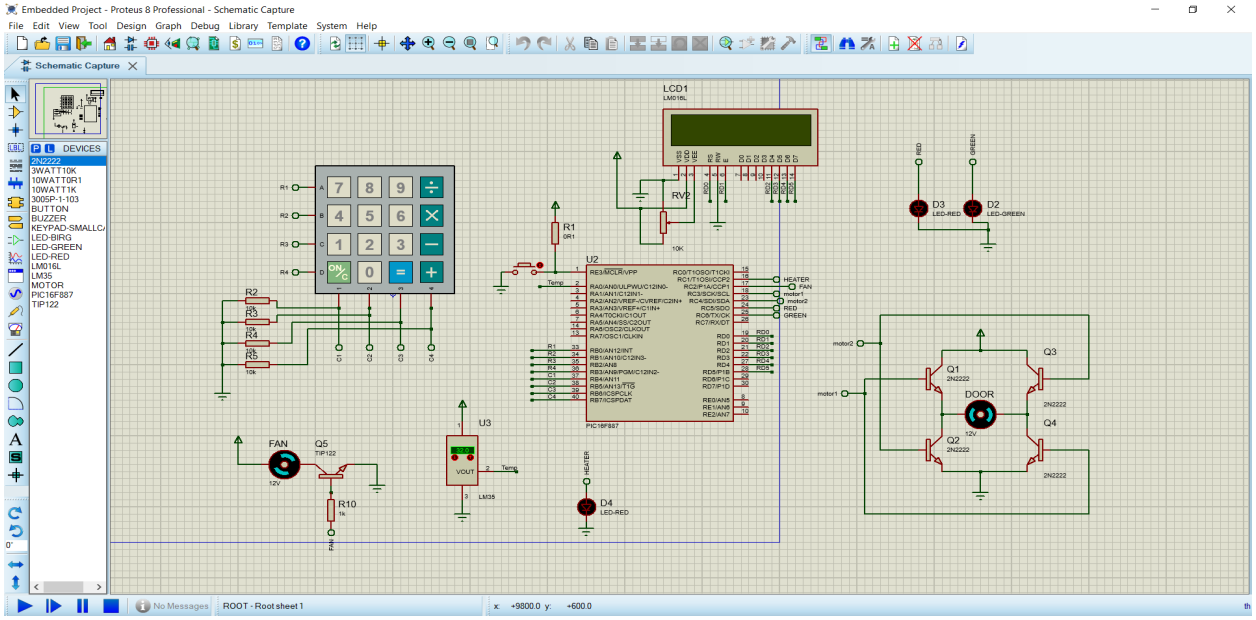
Input Pins:

Component	Connected to
Temperature sensor	RA0
KeyPad	RB0 to RB7

Output Pins:

Component	Connected to
LCD	RD0 to RD5
Heater	RC1
FAN	RC2
Motor1	RC3
Motor2	RC4
Red LED	RC5
Green LED	RC6

Hardware Design & Implementation



Software Design & Implementation:

Initialize and declaration:

```
sbit LCD_RS_Direction at TRISD0_bit;
sbit LCD_RS at RD0_bit;
sbit LCD_EN_Direction at TRISD1_bit;
sbit LCD_EN at RD1_bit;
sbit LCD_D4_Direction at TRISD2_bit;
sbit LCD_D4 at RD2_bit;
sbit LCD_D5_Direction at TRISD3_bit;
sbit LCD_D5 at RD3_bit;
sbit LCD_D6_Direction at TRISD4_bit;
sbit LCD_D6 at RD4_bit;
sbit LCD_D7_Direction at TRISD5_bit;
sbit LCD_D7 at RD5_bit;

char i;
char pass[] = {'7','7','7','7'}; //Door password
char check[5];
char sw_press = 0;
char x=0;
char passwordValid;
char text[7];
unsigned int buffer ;
int volt;
unsigned int temp;

char key_val()
{
    portb=1;
    delay_ms(20);
    if(portb.b4 == 1) return '1';
    else if(portb.b5 == 1) return '2';
    else if(portb.b6 == 1) return '3';
    portb=2;
    delay_ms(20);
    if(portb.b4 == 1) return '4';
    else if(portb.b5 == 1) return '5';
```

```
    else if(portb.b6 == 1) return '6';
    portb=4;
    delay_ms(20);
    if(portb.b4 == 1)  return '7';
    else if(portb.b5 == 1)  return '8';
    else if(portb.b6 == 1)  return '9';
    return '@';
}

void main() {
    ANSELH = 0x00;  //configure PORTB as digital
    ANSEL = 0x01;   //configure PORTA pin0 as analog
    TRISB = 0xF0;
    TRISC = 0x00;
    TRISA = 0x01;
    portc =0;
    portb =0;
    PWM1_init(1000);
    LCD_Init();
    lcd_cmd(_lcd_clear);
    lcd_cmd(_LCD_CURSOR_OFF);
    delay_ms(10);
    lcd_out(1,1,"Enter pass: ");

    for(i=0; i<4; i++)  eeprom_write(i,pass[i]);

    while(1)
    {
        /*PASSWORD*/
       sw_press = key_val();
        if(sw_press != '@')
        {
            check[x] = sw_press;
            lcd_chr(1,12+x,sw_press);
            x++;
```

Password check:

```
71          if(x>3)
            {
                passwordValid =1;
                for(i=0; i<4; i++) if(check[i] != eeprom_read(i)) passwordValid = 0;

                if(passwordValid){
                    lcd_cmd(_lcd_clear);
                    lcd_out(1,1,"WLECOME");
                    portc.b6 = 1; //green led on
80                  portc.b3 = 1;  //door motor turns clockwise to open
                    delay_ms(500);
                    portc.b3 = 0;   //stop door motor to enter home
                    delay_ms(500);
                    portc.b4 = 1;  //door motor turns counter clockwise to close
                    delay_ms(500);
                    portc.b4 = 0; //stop door motor
                    portc.b6 = 0; //turn off green led
                    lcd_cmd(_lcd_clear);
                    lcd_out(1,1,"Enter pass: ");
90              }else{
                    portc.b5 = 1; //red led on
                    lcd_cmd(_lcd_clear);
                    lcd_out(1,1,"WORNG PASSWROD!");
                    delay_ms(500);
                    portc.b5 = 0; //turn off red led
                    lcd_cmd(_lcd_clear);
                    lcd_out(1,1,"Enter pass: ");
                }
                x=0;
100         }
        }
        sw_press = '@';
        /***Temprature reading**/
        buffer = ADC_read(0);
        volt = buffer*4.88;
```

Controlling Fan speed

```
        temp = volt/10;
        Inttostr(temp,text);
        LCD_OUT(2,1,"Temp= ");
        LCD_OUT_CP(text);
        LCD_Chr_CP(" ");
        LCD_Chr_CP('C');
        /***Fan***/
        if(temp>=30 && temp<32)
        {
            PWM1_stop();
            PWM1_set_duty(204); //speed of fan is 80%
            PORTC.b1=0;
        }
        else if(temp>=32 && temp<34)
        {
            PWM1_stop();
            PWM1_set_duty(230); //speed of fan is 90%
            PORTC.b1=0;
        }
        else if(temp>=34)
        {
            PWM1_stop();
            PWM1_set_duty(255); //speed of fan is 100%
            PORTC.b1=0;
        }
        else if(temp<30)
        {
            PWM1_stop();
            PWM1_set_duty(0);  //turn off fan
            PORTC.b1=1;  //turn on heater.
        }
        delay_ms(10);
        PWM1_start();
    }
}
```

Testing :

Testing all functions separately.