

Universidad de San Carlos de Guatemala

Facultad de Ingeniería

Escuela de Ciencias y Sistemas

Arquitectura de Computadoras y Ensambladores 1

Ing. Otto Escobar

Aux. José Valdéz

Aux. Marielos Herrera



Diego Andrés Obín Rosales

201903865

Erick José André Villatoro Revolorio

201900907

Bryan Eduardo Gonzalo Méndez Quevedo

201801528

Elder Anibal Pum Rojas

201700761

Guatemala, 13 de Agosto del 2021

MANUAL TÉCNICO

PRÁCTICA #2

Descripción de la Práctica:

- Que se adquieran conocimientos, se aplique e interactué con el microcontrolador Arduino.

Objetivos de la Práctica:

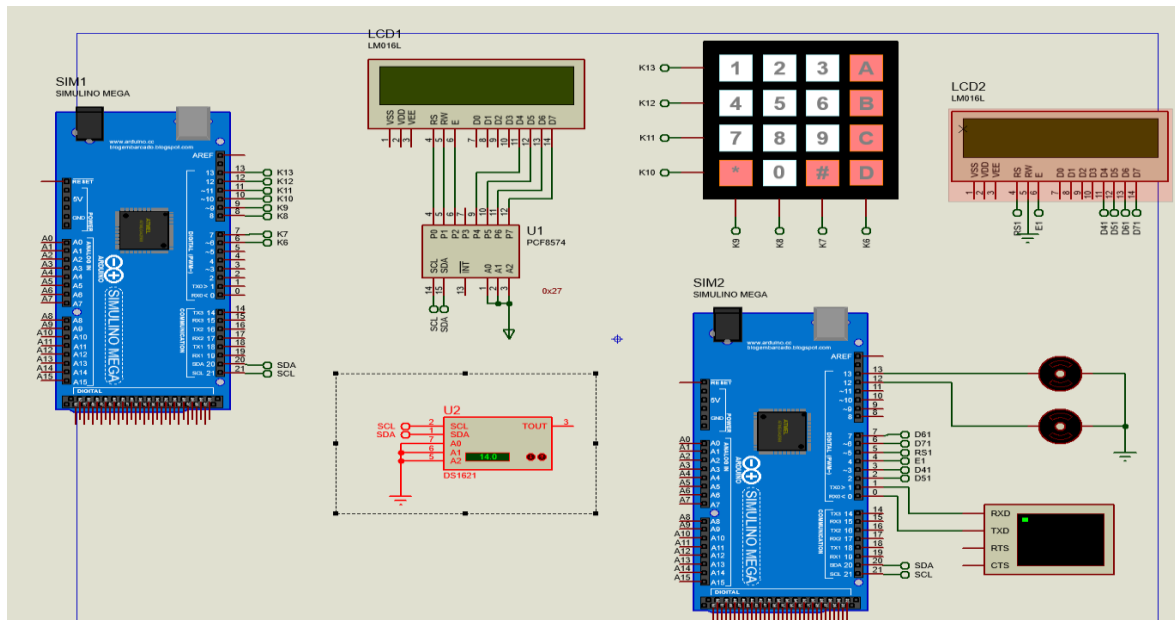
- Comprender el funcionamiento de las entradas y salidas, tanto digitales como análogas del microcontrolador Arduino.
- Comprender la configuración de 2 Arduinos en configuración maestro-esclavo mediante comunicación I2C.
- Conocer las funciones básicas de la salida serial.
- Aplicar el lenguaje C para estructuras de control en Arduino.

Componentes Utilizados:

1. Controlador LCD PCF8574
2. Sensor de temperatura: DS1621
3. 2 pantallas LCD
4. 2 Arduinos
5. 2 motores DC
6. Puente H L293d (opcional y no utilizado en esta práctica).
7. Pad numérico.
8. Software Proteus v8.10
9. Software Arduino IDE

Diseño de Simulación en Proteus

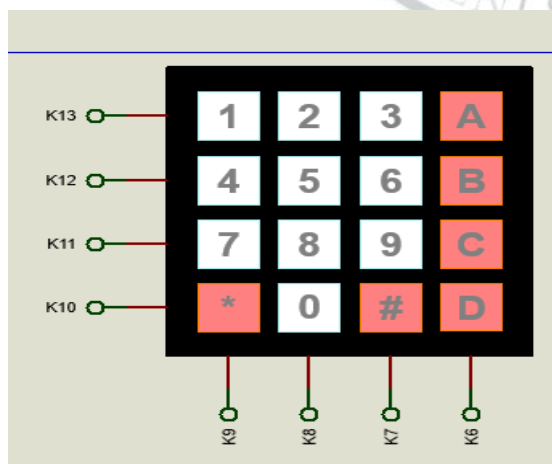
En la siguiente figura se muestra los componentes que se han incluido en el proyecto de Proteus para la simulación de esta práctica:



Código de Arduino

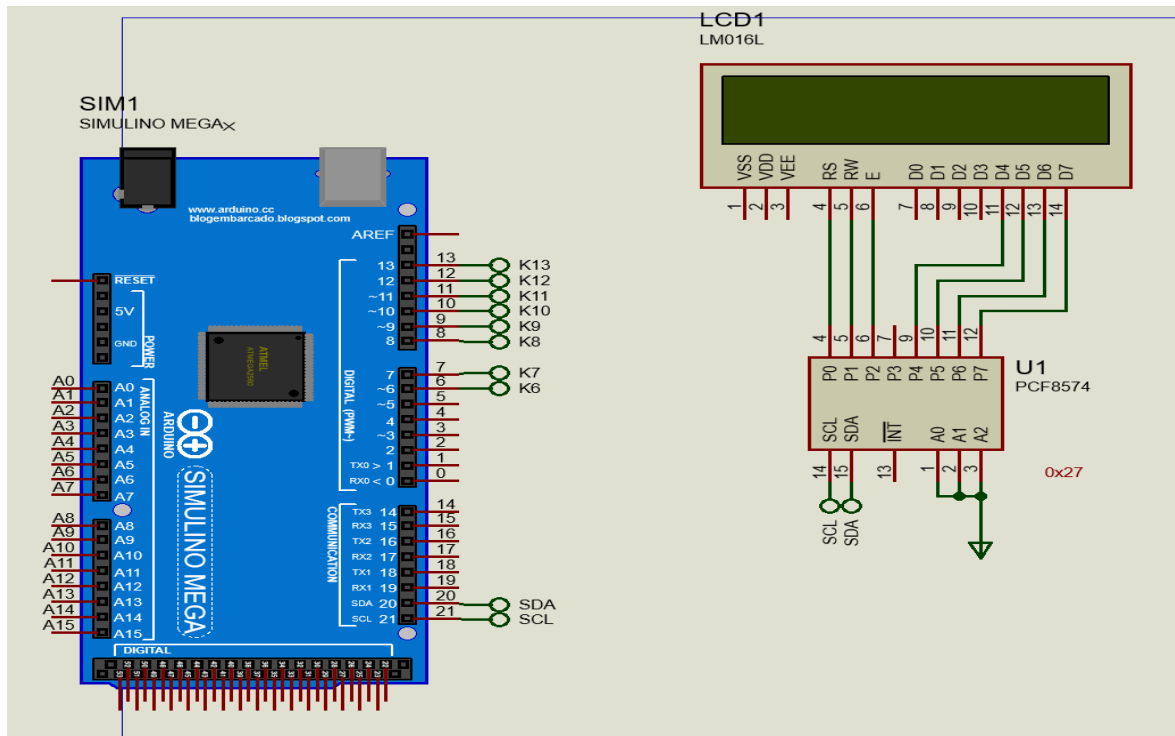
Para la configuración de la práctica se realizaron 2 programas en Arduino IDE, ambos utilizando la configuración del Simulino Mega en Proteus. Cada programa corresponde a 1 Simulino Mega (1 para el maestro y 1 para el esclavo). El código de ambos se encuentra en anexos.

Keypad



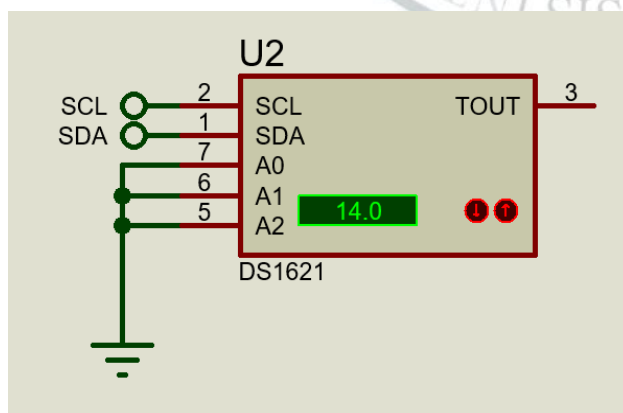
El keypad funciona para corroborar la contraseña que se va a ingresar y hacer funcionar la casa inteligente. Si se presiona “*” hará la validación de la contraseña ingresada y si se presiona “#” apagarán los motores (si uno o ambos estuvieran encendidos).

Configuración del Maestro



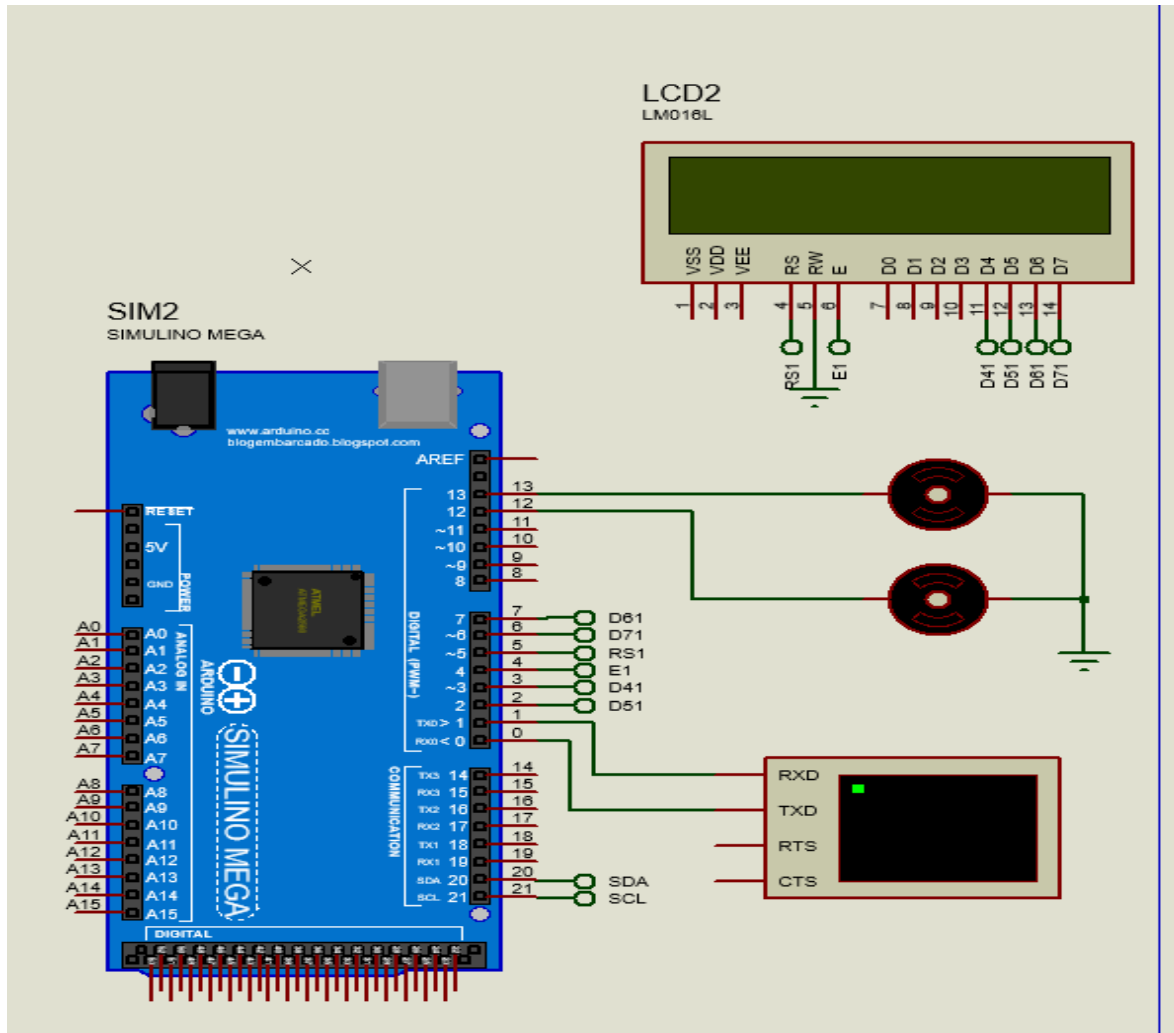
Este es el primer Arduino colocado en la práctica, el cual recibe los datos del Keypad y los valida, así mismo desplegando la respuesta en su respectivo LCD (en este caso el LCD1). Además, utilizando el bus de datos I2C podemos conectar al maestro y al esclavo con el componente PCF8574 y a su vez, conectarlo al sensor de temperatura.

Sensor de Temperatura



Este pequeño componente es uno de los principales protagonistas en la práctica, puesto que es el que permite ingresar los valores de temperatura (en °C). El Arduino esclavo es quien verifica la temperatura que este despliega y hace las respectivas validaciones para la parte de los motores.

Configuración del Esclavo



Este es el segundo Arduino de la práctica, el denominado esclavo. Esta configuración nos permite leer lo que el DS1621 toma de temperatura y lee las siguientes validaciones:

Si la temperatura < 18 °C	Ningún motor se enciende
Si la temperatura >= 18 °C y <= 25 °C	Un motor se enciende
Si la temperatura > 25 °C	Ambos motores se encienden

Además, este devuelve la validación al Arduino maestro y así, con el uso del keypad, puede presionarse la tecla “#” que nuevamente el maestro le envía al esclavo y de esa forma, mediante I2C, se pueden apagar los motores que en ese momento estén funcionando.

Estructura del Código

Arduino Maestro:

```
const byte FILA = 4;
const byte COLS = 4;

char hexaKeys[FILA][COLS] = {
  {'1','2','3','A'},
  {'4','5','6','B'},
  {'7','8','9','C'},
  {'*','0','#','D'},
};
```

Se realiza la declaración del KeyPad a utilizar y la posición de sus respectivos botones.

```
void setup() {

  lcd.init();
  lcd.backlight();

  lcd.setCursor(1,0);
  lcd.print("CASA ACYE1");
  lcd.setCursor(1,1);
  lcd.print("B-G01-S2");

  Wire.begin();
  setupTermometro();
}
```

Inicia el programa con los datos iniciales para la LCD #1

```
void loop() {
  char teclitaRica = tecladitoRico.getKey();

  if(teclitaRica == '#'){
    acceso = false;
  }
}
```

```

if(teclitaRica == '*'){
    if(compararCONTRA()){
        acceso = true;
        lcd.clear();
        print_text_lcd("BIENVENIDO ",0);
        print_text_lcd("A CASA ", 1);
    }else{
        acceso = false;
        lcd.clear();
        print_text_lcd("ERROR EN ",0);
        print_text_lcd("CONTRASENIA : ",1);
        delay(3000);
        lcd.clear();
        print_text_lcd(contraKEY,1);
    }
}else if(teclitaRica!=NULL){
    contraKEY[contadorcontra] = teclitaRica;
    contadorcontra++;
    if (contadorcontra==6){
        BorrarContra();
        contadorcontra=0;
    }
}

if(acceso){
    enviarAEsclavo(leerTemperatura());
}else {
    enviarAEsclavo(0);
}
}

```

Se ejecuta la comparación de la contraseña ingresada en el KeyPad, cuándo su longitud de cadena llega a 6 y aún no se ha ingresado la contraseña correcta, esta se reiniciará. De caso contrario si la contraseña es la correcta, dejará ingresar al usuario mostrando el mensaje de “Bienvenido a Casa” y al mismo tiempo enviando la señal al SIM2 o al Arduino Esclavo para la activación de la LCD de Temperatura y los Motores.

```
int leerTemperatura(){
    int temperatura = 0;
    Wire.beginTransmission(I2C_TERM);
    Wire.write(0xAA);
    Wire.endTransmission(false);
    Wire.requestFrom(I2C_TERM,1);
    temperatura = Wire.read();
    return temperatura;
}
```

Este método regresa la temperatura del DS1621.

```
void enviarAEsclavo(int a){
    delay(100); //milisegundos
    valor = a;
    Wire.beginTransmission(I2C_ESCLAVO);
    Wire.write(valor);
    Wire.endTransmission();
}
```

Este método regresa los datos de la Temperatura marcada en el DS1621 al SIM 2 o Arduino Esclavo.

ARDUINO ESCLAVO:

```
void setup() {

    pinMode(motor2, OUTPUT);
    pinMode(motor1, OUTPUT);

    // la pantalla
    lcd.begin(16,2);
    Serial.begin(9600);

    print_text_lcd("Apagado",0);

    // I2c
    Wire.begin(I2C_MAESTRO); //Acá decimos que el esclavo se va a identificar
    con el # 1
}
```



```
Wire.onReceive(lectura);  
}
```

Envía los datos iniciales a la LCD #2 que es controlada por el SIM 2 o el Arduino Esclavo. A un inicio los motores se encuentran apagados al igual que el sensor de la temperatura. Recibe los datos del Arduino Master.

```
void lectura(){  
  valorp= Wire.read();  
  
  if(valorp ==0){  
    lcd.clear();  
    print_text_lcd("Apagado",0);  
  } else {  
    activarMotores(valorp);  
    Serial.println(valorp);//imprimimos en el monitor serial del esclavo  
  }  
}
```

Si el valor recibido por parte del Arduino Maestro es diferente de cero entonces este dato será enviado para Activar o Desactivar los motores.

```
void activarMotores(int temperaturaC){  
  if (temperaturaC <= 18){  
    if (temperaturaC < 0) {  
      temperaturaC = abs(temperaturaC);  
      sprintf(cBuffer, "-  
%02u.%1u%cC", temperaturaC, temperaturaC % 10, 223);  
    }  
    else {  
      temperaturaC = abs(temperaturaC);  
      sprintf(cBuffer, "%02u.%1u%cC", temperaturaC, temperaturaC % 10, 223);  
    }  
    digitalWrite(motor1, LOW);  
    digitalWrite(motor2, LOW);  
  
    print_text_lcd("LVL: 1", 1);  
  }  
  else if (temperaturaC >= 18 && temperaturaC <= 25){  
    temperaturaC = abs(temperaturaC);  
    sprintf(cBuffer, "%02u.%1u%cC", temperaturaC, temperaturaC % 10, 223);  
  
    digitalWrite(motor1, HIGH);  
    digitalWrite(motor2, LOW);  
    print_text_lcd("LVL: 2", 1);  
  }  
}
```

```
}  
else{  
    temperaturaC = abs(temperaturaC);  
    sprintf(cBuffer, "%02u.%1u°C", temperaturaC, temperaturaC % 10, 223);  
    digitalWrite(motor1, HIGH);  
    digitalWrite(motor2, HIGH);  
  
    print_text_lcd("LVL: 3", 1);  
}  
  
    print_text_lcd(cBuffer, 0);  
}
```

Si la temperatura esta debajo de los 18° C ninguno de los 2 motores será encendido. Se mostrará en la LCD #2 "LVL: 1"

Si la temperatura esta entre los 25° y 18° C se encenderá uno de los dos motores. Se mostrará en la LCD #2 "LVL: 2"

Si la temperatura esta sobre los 25°C se encenderán los 2 motores. Se mostrará en la LCD #2 "LVL: 3"

