

Universidad de San Carlos  
Facultad de Ingeniería  
Escuela de Ciencias y Sistemas  
Arquitectura de Computadores y Ensambladores 1  
Sección N

GRUPO:

**#10**

**Práctica #2**



**Integrantes:**

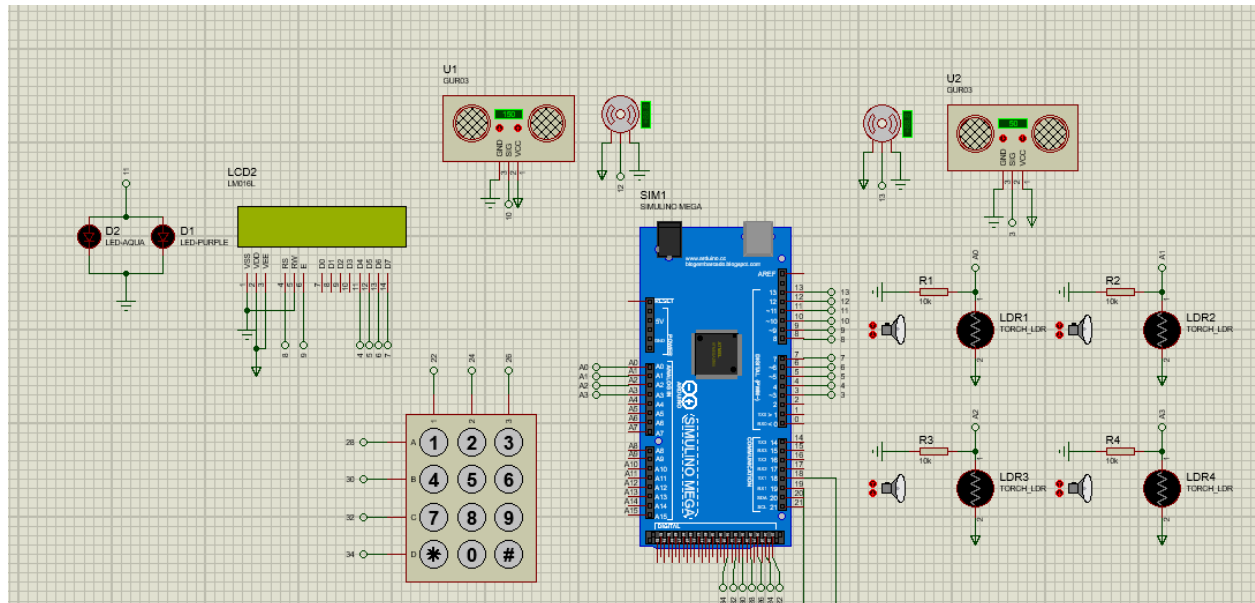
<b>Carné:</b>	<b>Nombre:</b>
201020252	Sergio Ariel Ramírez Castro
201800469	José Alejandro Lorenty Herrera
201801366	Josue Guillermo Orellana Cifuentes
201801397	Cristian Francisco Meoño Canel
201800555	César Alejandro Sosa Enríquez

**Fecha de entrega: 17 de junio de 2021**

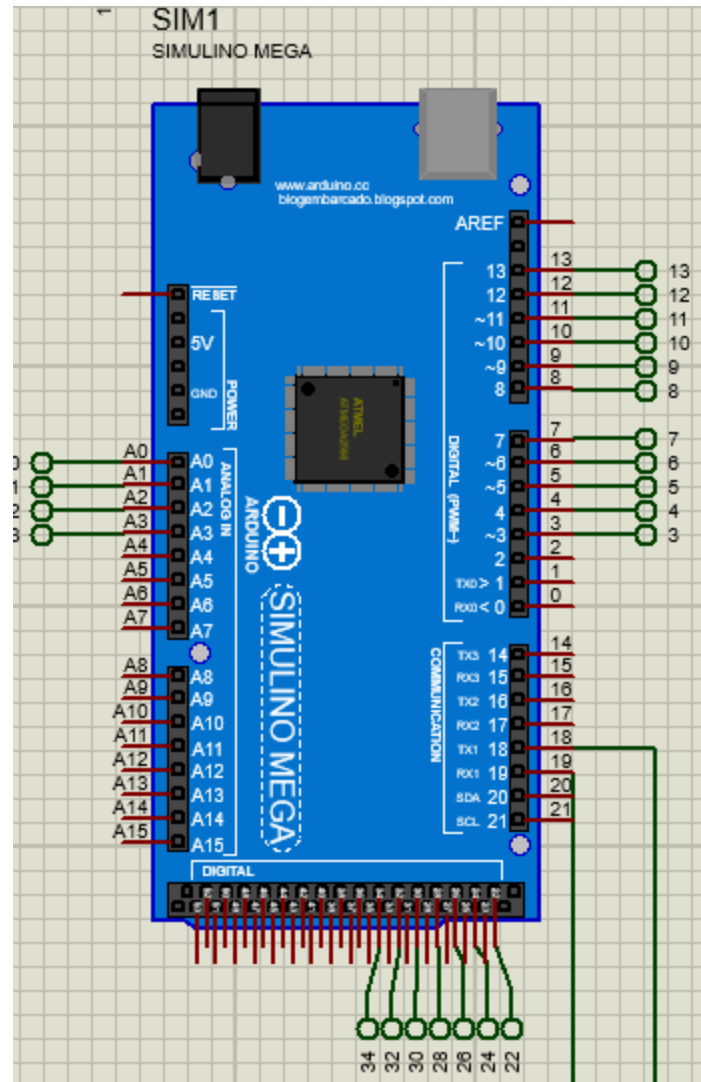
## **Introducción**

En el mundo de los componentes y la electrónica se tienen varias combinaciones las cuales pueden generar distintas opciones para crear y desarrollar actividades de aprendizaje como laborales, en el área de controladores y dispositivos electrónicos se tiene el dispositivo y herramienta conocida como ARDUINO, este dispositivo más su desarrollo de programación basado en C, crea la oportunidad de desarrollar mecánicas de interacción y desarrollo para que se pueda conectar e interactuar con dispositivos análogos y digitales, como sensores, foto resistencias, pantallas led, etc.

## Circuitos desarrollados:

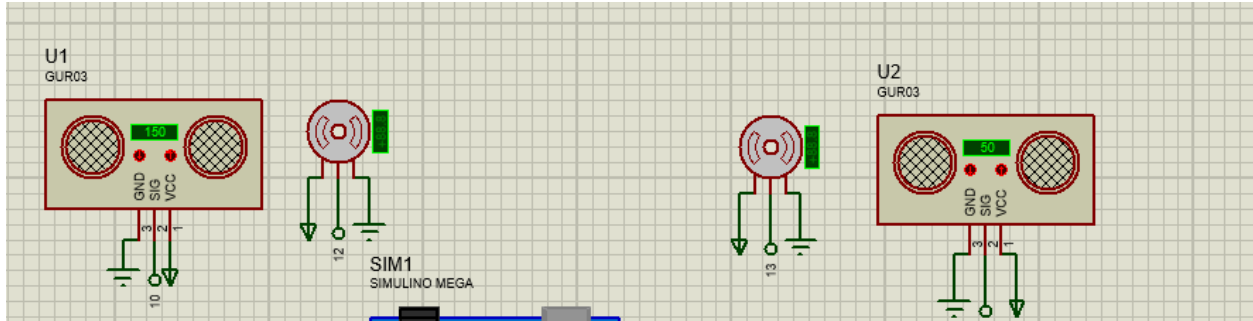


Circuito general de toda la aplicación, se tienen los dos sensores super sónicos para la simulación de las puertas que están con motores que giran a 90 grados para abrir y cerrar las puertas de la casa, las 4 foto resistencias simulando las luces de las cuatro habitaciones de la casa, además los mensajes existentes en la pantalla led de 16x2 para mostrar los mensajes correspondientes de esta y por último el tablero numérico para el ingreso de la contraseña.



En el ARDUINO, SIMULINO MEGA 250, se pines para los sensores s3nicos y el teclado num3rico, adem3s para los servos motores se tienen sus propios pines. Y en los an3logos se encuentras las fotos resistencias para las luces.

En el caso de dem3s pines, todos son entradas de lectura para el ARDUINO, todos estos dan pulsos de ceros y unos el cual este dispositivo tiene la capacidad de entender y con l3gica de programaci3n entrar y generar programas.

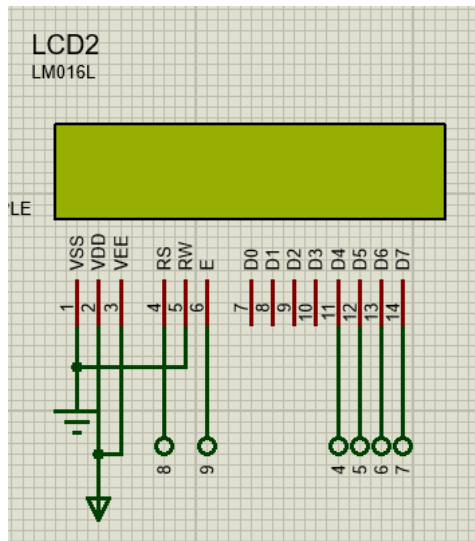


Los sensores sónicos son para tener una simulación de el acercamiento de una persona para activar la contraseña para entrar y para detectar que se cerrará la puerta.

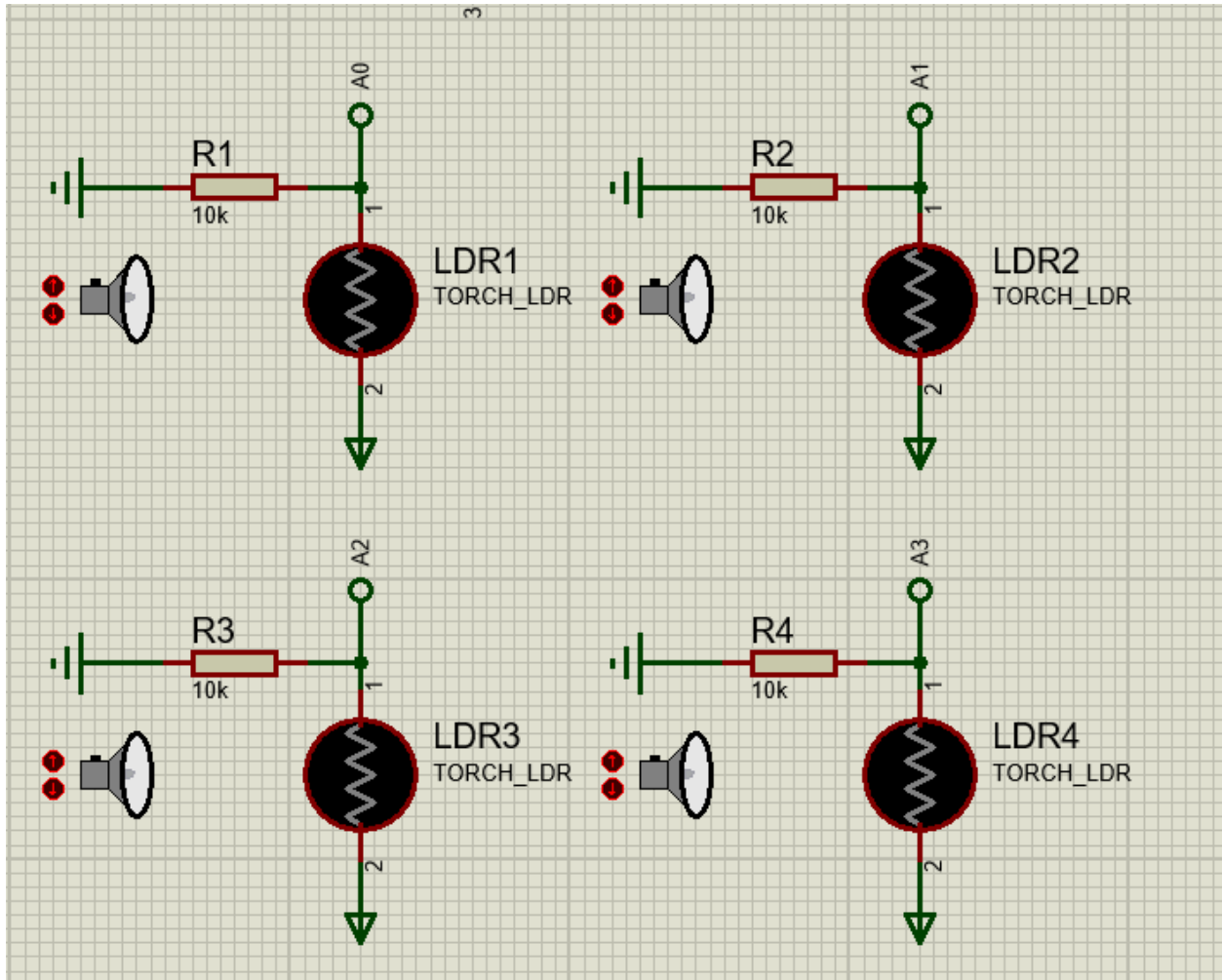
Estos están codificados con la siguiente porción de código:

```
long readUltrasonicDistance(int triggerPin, int echoPin){
  pinMode(triggerPin, OUTPUT);
  digitalWrite(triggerPin, LOW);
  delayMicroseconds(2);
  digitalWrite(triggerPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(triggerPin, LOW);
  pinMode(echoPin, INPUT);
  return pulseIn(echoPin, HIGH);
}
```

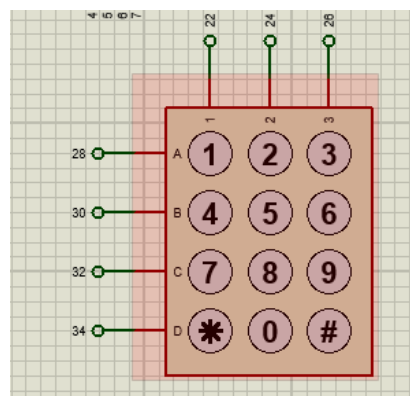
Donde se leen los pines de proyección y la del eco, para que tenga el sensor si el sonido tiene un tope y rebota o no tienen ningún retorno detonando que no tiene ningún objeto delante.



La pantalla led tiene los pines del 4 a 7 como entrada para poder recibir el mensaje a mostrar dependiendo de la situación.



Las 4 foto resistencias, están en conjunto para determinar análogamente si una luz de un cuarto está conectada o no y mostrar el mensaje en la pantalla.



**Código Utilizado:**

```

#include <LiquidCrystal.h>
#include <Servo.h>
#include <Keypad.h>

```

Las librerías utilizadas son para controlar de manera simple la pantalla led, los servo motores y el tablero numérico para que todo esté conectado y pueda tener control de dicho elemento.

```

//Configuracion pantalla LCD
const int d4 = 4, d5 = 5, d6 = 6, d7 = 7, rs = 8, enabled = 9;

LiquidCrystal pantalla(rs, enabled, d4, d5, d6, d7);

...

```

La mayoría de los pines de salida para la pantalla están en orden numérico de conexión, esto para facilitar la conexión en el propio ARDUINO, además estos pines sirven para crear el objeto de la pantalla.

```

//Configuracion teclado
const byte columnas = 3, filas = 4;

char teclas[filas][columnas] = {
  {'1', '2', '3'},
  {'4', '5', '6'},
  {'7', '8', '9'},
  {'*', '0', '#'}
};

byte pinesFila[filas] = {28, 30, 32, 34};
byte pinesColumna[columnas] = {22, 24, 26};

Keypad teclado = Keypad(makeKeymap(teclas), pinesFila, pinesColumna, filas, columnas);

```

Para el teclado numérico, se establece el mapeado del teclado con un byte del número de columnas y filas, con ello se puede crear el arreglo de pines para que el ARDUINO reconozca tales pines como entrada del número.



```

char enie = 0xEE;
String contra = "";
int contadorErronea = 0;

//Configuracion ultrasonico
int pinUltraIn = 10;
int pinUltraOut = 3;
int cmIn = 0;
int cmOut = 0;

//Configuracion servomotor
int pinServoIn = 12;
int pinServoOut = 13;

```

El CHAR “ENIE” contiene el byte de la letra “ñ”, para poder ingresarla a la pantalla debido a que estas pantallas no tienen como símbolo para la letra “ñ”; la contraseña, y el contador de la contraseña están como variables, los pines para los sensores sónicos y los servos motores, además de los contadores de distancia de la puerta de entrada y salida.

```

Servo servoIn;
Servo servoOut;

//verificador de estadia
boolean ingreso = false;

```

Se crean los objetos de los motores y la variable de ingreso booleana para el cambio de entrada y salida.

```

void setup() {
  Serial1.begin(9600);

  pantalla.begin(16, 2);

  pinMode(11, OUTPUT);

  servoIn.attach(pinServoIn);
  servoIn.write(0);
  servoOut.attach(pinServoOut);
  servoOut.write(0);
}

```

En el SETUP se inicializan los motores con cero, la pantalla con sus 16 espacios y 2 filas, además el pin 11 como salida para la alarma de leds.

```
void loop() {
  if(!ingreso){
    //Puerta entrada
    cmIn = 0.01723 * readUltrasonicDistance(pinUltraIn, pinUltraIn);
    if(cmIn <= 150){
      validacionContra();
    }else{
      mensajeBienvenida();
    }
  }else{
    //control de luces
    controlLuces();
    pantalla.clear();

    //puerta de salida
    cmOut = 0.01723 * readUltrasonicDistance(pinUltraOut, pinUltraOut);
    if(cmOut <= 150){
      movimientoServo(servoOut);
      ingreso = false;
    }
  }
}
```

El LOOP tiene la selección del mensaje y la habilitación de la contraseña para poder acceder a la casa y todos los ciclos de toda la aplicación

```
void mensajeBienvenida() {
  pantalla.print("Casa inteligente");
  pantalla.setCursor(0, 1);
  pantalla.print("ACE1");
  delay(1500);
  pantalla.clear();
  delay(1000);
  pantalla.print("ACE1-A-G3-S1");
  delay(1500);
  pantalla.clear();
  delay(1000);
}
```

El Método mensaje de bienvenida imprime la primera sección de la práctica en la pantalla con un DELAY para cambio de texto y apagarse cada cierto tiempo.

```
void controlLuces() {
    int cuartos[4] = {analogRead(A0), analogRead(A1), analogRead(A2), analogRead(A3)};
    for(int i = 1; i <= 4; i++){ mostrarEstadoLuz(i, cuartos[i-1]); }
}

void mostrarEstadoLuz(int cuarto, int estado){
    pantalla.clear();
    pantalla.write("Habitacion ");
    pantalla.print(cuarto, DEC);
    pantalla.setCursor(0, 1);
    if(estado == 10){ pantalla.write("Luz apagada"); }
    else{ pantalla.write("Luz prendida"); }
    delay(2000);
}
```

El método de luces tiene el chequeo de ver el estado de las foto resistencias para tener en cuenta si están apagadas o encendidas; además busca y si encuentra el estado encendido en la pantalla determina si está apagada o encendida.

```
void lecturaContra() {
    while(true) {
        char tecla = teclado.getKey();
        if(tecla != NO_KEY) {
            if(tecla != '*') { contra += tecla; }
        }
        if(tecla == '*') { break; }
    }
}
```

Para la lectura de la contraseña, se almacena el conjunto de números hasta que se presiona el "\*" para intentar validar la contraseña.

```

void validacionContra() {
    pantalla.clear();
    pantalla.print("Ingrese su");
    pantalla.setCursor(0, 1);
    pantalla.print("constrase");
    pantalla.print(enie);
    pantalla.print("a");

    contra = "";

```

Para validar la contraseña se tiene la impresión en la pantalla de ingresar contraseña:

```

lecturaContra();

if(contra == "20213"){
    contadorErronea = 0;
    pantalla.clear();
    pantalla.print("Bienvenido a");
    pantalla.setCursor(0, 1);
    pantalla.print("casa ^_^");
    movimientoServo(servoIn);
    ingreso = true;
    delay(1000);
}else{
    pantalla.print("Error en");
    pantalla.setCursor(0, 1);
    pantalla.print("constrase");
    pantalla.print(enie);
    pantalla.print("a");
    contadorErronea += 1;
    if(contadorErronea == 3){
        contadorErronea = 0;
        pantalla.clear();
        pantalla.print("Acceso no");
        pantalla.setCursor(0, 1);
        pantalla.print("autorizado");
        unsigned long inicioAlarma = millis();

```

Luego de recibir la contraseña la compara con la que es solicitada si es correcta simplemente entra al menú de la casa y se puede continuar con la aplicación, si no se empieza a contar los malos intentos para activar la alarma y reiniciar el sistema.

```

void movimientoServo(Servo servo){
  for(int i = 0; i < 90; i++){
    servo.write(i);
    delay(15);
  }
  delay(3000);
  for(int i = 90; i >= 0; i--){
    servo.write(i);
    delay(15);
  }
}

```

Para el servo Motor se tiene el movimiento con un ciclo FOR que genera los 90 pulsos para mover el motor 90 grados.

```

,

void intermitencia(){
  digitalWrite(11, HIGH);
  delay(1000);
  digitalWrite(11, LOW);
  delay(1000);
}

```

Y el método de intermitencia, para encender y apagar las luces.