

Universidad Politécnica de Madrid

Grado en Ingeniería Electrónica Industrial y Automática

Asignatura

**INFORMÁTICA**

Curso 2018-2019



Universidad Politécnica de Madrid

Grado en Ingeniería Electrónica Industrial y Automática

Datos del Grupo

*Adrián González Denia 54647*

*Jorge Fernández Jiménez 54591*

# Introducción

El trabajo esta desarrollado para la utilización del un cerrojo, a través del giro de un servomotor. Este cerrojo se activa con un código que se escribe en un keyboard y se muestra en un LCD. Además, el código va acompañado con unos leds y un buzzer que se encienden y suenan en función de si el código es correcto o incorrecto. Para la simulación de un cerrojo lo hemos realizado en una caja.

# Desarrollo circuito

Esquema del circuito:

Imagen que contiene valla, mesa, edificio

Descripción generada automáticamente

# Desarrollo del código

Imagen que contiene texto

Descripción generada automáticamente

//LIBRERIAS UTILIZADAS

//LIBRERÍA DEL SERVOMOTOR CON LA CLASE SERVO PARA DEFINIR EL NOMBRE CON EL CUAL SE VA A UTILIZAR

#include <Servo.h>

Servo servoMotor;

//LIBRERÍA LIQUIDCRYSTAL\_I2C PARA ESTABLECER LAS FILAS Y COLUMNAS DEL LCD JUNTO AL NOMBRE CON EL CUAL SE VA A UTILIZAR

#include <LiquidCrystal\_I2C.h> //Actualizacion de libreria LiquidCrystal

LiquidCrystal\_I2C lcd(0x27, 16, 2);

//LIBRERÍA DEL TECLADO MATRICIAL

#include <Keypad.h> //Necesaria descarga libreria Keypad.h

#include <Key.h>

//CONSTANTES

#define DELAY 400

#define FILAS 4 //Filas del teclado matricial

#define COLUMNAS 3 //Columnas del teclado matricial

//PINES A LOS QUE ESTAN CONECTADOS LOS COMPONENTES (LEDS Y BUZZER)

#define buzzer 12

#define ledverde 9

#define ledrojo 10

void setup() {

Serial.begin(9600);

servoMotor.attach(11); //PIN DEL SERVOMOTOR

pinMode(buzzer, OUTPUT);

pinMode(ledverde, OUTPUT);

pinMode(ledrojo, OUTPUT);

servoMotor.write(0); //POSICION INICIAL DEL SERVOMOTOR (0º)

//INICIA EL LCD MEDIANTE UNA FUNCION

IniciarLCD();

}

void loop() {

//Definir el teclado

static char teclado[FILAS][COLUMNAS] = {

{ '1','2','3' },

{ '4','5','6' },

{ '7','8','9' },

{ '\*','0','#' }

};

static byte pinColumnas[COLUMNAS] = { 4,3,2 };

static byte pinFilas[FILAS] = { 8,7,6,5 };

static Keypad keypad = Keypad{ makeKeymap(teclado), pinFilas, pinColumnas, FILAS, COLUMNAS }; //rastreo del teclado

//variables

static int cont = 0, activar = 0, cerrar = 0, intentos = 0, incorrecto = 0, cambio = 0, cursor = 0, RESTART = 0, dim = 1, i = 6, error = 0;

static char contrasenaantigua[] = " "; //contraseña antigua, se guardara cuando cambie la contraseña

static char contrasena[] = "0000"; //aquí escribimos la contraseña inicial de 4 dígitos

static char codigo[] = " "; //Cadena donde se guardaran los caracteres de las teclas presionadas

static char aceptar[] = " "; //Cadena aceptar

//FUNCION CHAR TECLAS = KEYPAD.GETKEY(); GUARDA LAS TECLAS PRESIONADAS EN LA VARIABLE LLAMADA TECLAS, QUE AL SER PULSADA LEE EL VALOR GUARDADO

char teclas = keypad.getKey();

if (teclas) //LEE LA VARIABLE GUARDADA CADA VEZ QUE SE PULSE UNA TECLA (ESTE IF LLEGA HASTA QUE SE ACABA DE PEDIR LA CONTRASEÑA PARA GIRAR EL SERVO)

{

//INICIO DEL PROGRAMA CON PIN INICIAL:

//OPCION # CONTINUAR UTILIZANDO EL PIN INICIAL

//OPCION \* CAMBIAR PIN

if (cambio == 0 && teclas == '#') //Continuar con pin inicial

{

cambio = 8;

}

if (cambio == 0 && teclas == '\*') //Cambiar pin

{

lcd.clear();

cursor = 0;

cambio = cambio + 1;

}

//OPCION CAMBIAR PIN:

if (cambio < 5 && cambio > 0) //Cambiar la contraseña, pide contraseña antigua para establecer la nueva

{

lcd.display();

lcd.setCursor(2, 0);

lcd.print("PIN ANTIGUO");

//HAY QUE INTRODUCIR CORRECTAMENTE LA CONTRASEÑA ANTERIOR PARA PODER CAMBIARLA, SI AL PULSAR LAS TECLAS, MEDIANTE UN CONTADOR, NO COINCIDE CON LA CADA NUMERO DE LA CONTRASEÑA ANTIGUA SE INICIA UN NUEVO CONTADOR DE ERROR.

if (teclas == contrasena[cursor] && teclas != '#' && teclas != '\*')

{

contrasenaantigua[cursor] = contrasena[cursor];

lcd.setCursor(i, 1);

lcd.print(teclas);

}

if (teclas != contrasena[cursor] && teclas != '#' && teclas != '\*')

{

contrasenaantigua[cursor] = teclas;

lcd.setCursor(i, 1);

lcd.print(teclas);

error = error + 1;

}

i = i + 1;

cursor = cursor + 1;

cambio = cambio + 1;

if (teclas == '\*' || teclas == '#')

{

clearLCD();

i = 6;

cursor = 0;

cambio = 1;

}

}

//UNA VEZ PUESTA LA CONTRASEÑA ANTERIOR, EXAMINA SI ES CORRECTA O NO, LA CONTRASEÑA ANTIGUA ES CORRECTA SI EL CONTADOR DE ERROR ES NULO

if(cambio==5)

{

if (cursor==4)

{

//EN CASO DE QUE SEA CORRECTA, AL PULSAR # PARA CONFIRMAR, TE DEJA ESTABLECER UNA NUEVA

if (error == 0) //Contraseña antigua correcta

{

if (teclas == '#')

{

clearLCD();

i = 6;

cursor = 0;

cambio = cambio + 1;

}

}

//EN CASO DE QUE SEA INCORRECTA, AL PULSAR # PARA CONFIRMAR, TE VUELVE A PEDIR LA CONTRASEÑA ANTIGUA

if (error != 0) //Contraseña antigua incorrecta, vuelve a pedirla

{

if (teclas == '#')

{

clearLCD();

i = 6;

cursor = 0;

cambio = 1;

error = 0;

}

}

}

}

//UNA VEZ CONFIRMADO Y LA CONTRASEÑA ANTIGUA CORRECTA TE PIDE INTRODUCIR LA NUEVA CONTRASEÑA

if (cambio == 6) //Pedir nueva contraseña

{

lcd.display();

lcd.setCursor(2, 0);

lcd.print(" PIN NUEVO ");

if (teclas == '#' || teclas == '\*')

{

i = 6;

cursor = 0;

clearLCD();

}

//LEE LA NUEVA CONTRASEÑA AL PULSAR CADA TECLA, MEDIANTE UN CONTADOR HASTA LLEGAR AL NUMERO MAXIMO DE DIGITOS, EN ESTE CASO 4, CUANDO LLEGA EL CONTADOR AL MAXIMO, TE PIDE CONFIRMAR LA NUEVA CONTRASEÑA

else

{

contrasena[cursor] = teclas;

lcd.setCursor(i, 1);

lcd.print(teclas);

i = i + 1;

cursor = cursor + 1;

}

//UNA VEZ CONFIRMADA LA CONTRASEÑA NUEVA SE ENCIENDE EL LED VERDE DURANTE UN SEGUNDO

if (cursor == 4)

{

lcd.clear();

lcd.display();

lcd.setCursor(3, 0);

lcd.print("CONFIRMAR");

lcd.setCursor(6, 1);

lcd.print(contrasena);

LedVerde();

Apagarled();

cambio = cambio + 1;

}

}

if (cambio == 7) //Confirmar nueva contraseña para empezar a usarla

{

if (teclas == '#')

{

cambio = cambio + 1;

i = 6;

cont = 0;

activar = 0;

incorrecto = 0;

}

//SI QUIERES VOLVER A CAMBIARLA ANTES DE SU USO AL PULSAR \* TE LLEVA AL PRINCIPIO DEL PROGRAMA, EN ESTE CASO TODAS LAS VARIABLES SE REINICIAN

if (teclas == '\*' && intentos!= 3) //Volver a cambiar contraseña

{

lcd.clear();

lcd.display();

lcd.setCursor(4, 0);

lcd.print("CAMBIAR");

lcd.setCursor(7, 1);

lcd.print("PIN");

cursor = 0;

cambio = 0;

cont = 0;

activar = 0;

incorrecto = 0;

error = 0;

i = 6;

}

}

//UNA VEZ ESTABLECIDA LA NUEVA CONTRASEÑA, O EN SU DEFECTO, HABER DECIDIDO SEGUIR CON EL PIN INICIAL EN LAS OPCIONES AL PRINCIPIO DEL PROGRAMA, SE PROCEDE A L TRABAJO DEL CERROJO

//Abrir el cerrojo

if (cambio == 8)

{

//MEDIANTE UN CONTADOR HASTA LLEGAR AL MAXIMO NUMERO DE DIGITOS, SE LEEN LAS TECLAS PULSADAS.

if (cont < 4) //Pide el pin

{

lcd.clear();

IntroducePin();

//SI LAS TECLAS COINCIDEN UNA A UNA CON LA CONTRASEÑA ESTABLECIDA, SE INICIA UN NUEVO CONTADOR LLAMADO ACTIVAR

if (teclas == contrasena[cont] && teclas != '#' && teclas != '\*')

{

codigo[cont] = contrasena[cont];

lcd.setCursor(i, 1);

lcd.print(teclas);

activar = activar + 1;

}

//SI LAS TECLAS COINCIDEN UNA A UNA CON LA CONTRASEÑA ESTABLECIDA, SE INICIA UN NUEVO CONTADOR LLAMADO INCPRRECTO

if (teclas != contrasena[cont] && teclas != '#' && teclas != '\*')

{

codigo[cont] = teclas;

lcd.setCursor(i, 1);

lcd.print(teclas);

incorrecto = incorrecto + 1;

}

i = i + 1;

cont = cont + 1;

//SI EN SU DEFECTO SE PULSA LA TECLA # ANTES DE FINALIZAR EL CONTADOR (4) LA CONTRASEÑA PUESTA SE BORRARA Y VOLVERA A PEDIRLA

if (teclas == '#') //Borrar pin puesto, vuele a pedirlo

{

i = 6;

cont = 0;

incorrecto = 0;

}

}

//SI SE PULSA LA TECLA \* TE LLEVARA A LA OPCION INICIAL DE CAMBIAR LA CONTRASEÑA Y PODER ESTABLECER UNA NUEVA, EN ESTE CASO TODAS LAS VARIABLES SE REINICIAN

if (teclas == '\*' && cont == 0) //Cambiar contraseña

{

lcd.clear();

lcd.display();

lcd.setCursor(4, 0);

lcd.print("CAMBIAR");

lcd.setCursor(7, 1);

lcd.print("PIN");

cursor = 0;

cambio = 1;

cont = 0;

incorrecto = 0;

activar = 0;

error = 0;

i = 6;

}

}

//UNA VEZ PUESTA LA CONTRASEÑA (LLEGADO EL CONTADOR A 4) TE PIDE CONFIRMAR LA ELECCION

if (cont == 4) //Una vez puesta la contraseña

{

if (teclas == '#') //Confirmar eleccion

{

//SI EL CONTADOR ACTIVAR ES IGUAL A 4 (NUMERO DE DIGITOS QUE TIENE LA CONTRASEÑA) SIGNIFICA QUE LA CONTRASEÑA INTRODUCIDA POR EL TECLADO ES CORRECTA, LUEGO SE PROCEDERA A ABRIR EL CERROJO

if (activar == 4) //C0ntraseña correcta

{

aceptar[0] = teclas;

cont = cont + 1;

}

//SI EL CONTADOR INCORRECTO ES MAYOR QUE 1, SIGNIFICA QUE LA CONTRASEÑA PUESTA POR EL TECLADO ES INCORRECTA, LUEGO VUELVE A PEDIRLA, INICIANDO UN NUEVO CONTADOR LLAMADO INTENTOS, ENCENDERA EL LED ROJO UNA VEZ Y DURANTE UN SEGUNDO MEDIANTE LA FUNCION LEDROJO1()

if (incorrecto <= 4 && incorrecto > 0) //Contraseña incorrecta, vuelve a pedirla

{

LedRojo1();

Apagarled();

clearLCD();

cont = 0;

activar = 0;

incorrecto = 0;

i = 6;

intentos = intentos + 1;

}

}

//SI SE PULSA LA TECLA \* TE LLEVARA A LA OPCION INICIAL DE CAMBIAR LA CONTRASEÑA Y PODER ESTABLECER UNA NUEVA, EN ESTE CASO TODAS LAS VARIABLES SE REINICIAN

if (teclas == '\*') //Cambiar contraseña

{

lcd.clear();

lcd.display();

lcd.setCursor(4, 0);

lcd.print("CAMBIAR");

lcd.setCursor(7, 1);

lcd.print("PIN");

cursor = 0;

cambio = 1;

cont = 0;

incorrecto = 0;

activar = 0;

error = 0;

i = 6;

}

}

}

//UNA VEZ PUESTO LA CONTRASEÑA Y QUE ESTA SEA CORRECTA SE PROCEDEA ABRIR EL CERROJO

if (aceptar[0] == '#'&&codigo[0] == contrasena[0] && codigo[1] == contrasena[1] && codigo[2] == contrasena[2] && codigo[3] == contrasena[3])

{

//ENCIENDE EL LED VERDE

LedVerde();

lcd.clear();

lcd.display();

lcd.setCursor(3, 0);

lcd.print("CONFIRMADO");

//EL CERROJO SE ABRE CON LA FUNCION ABRIR QUE GIRA EL SERVOMOTOR A UNA POSICION DE 90º

abrir();

//PARA CERRAR EL CERROJO HABRA QUE PULSAR LA TECLA \* QUE INICIARA LA VARIABLE RESTART LA CUAL REINICIA LAS VARIABLES AL PUNTO DEL PROGRAMA QUE TE PIDE INTRODUCIR EL PIN PARA VOLVER A ABRIR EL CERROJO

if (teclas == '\*') //Cierra el cerrojo

{

//APAGA EL LED VERDE Y ENTRA EN LA FUNCION CIERRA PARA GIRAR EL SERVOMOTOR A 0º

Apagarled();

cierra();

RESTART = 1;

aceptar[0] = teclas;

}

}

//SI POR EL CONTRARIO, LA CONTRASEÑA ES INCORRECTA TRES VECES (EL CONTADOR INTENTOS ES IGUAL A 3) SE INICIARA EL BUZZER CONSTANTEMENTE EN MODO DE ALARMA, A SU VEZ QUE EL LED ROJO HASTA QUE SE CANCELE

if (intentos == 3)

{

lcd.clear();

lcd.setCursor(4, 0);

lcd.print("DENEGADO");

tone(buzzer, 900);

delay(350);

noTone(buzzer);

LedRojo();

delay(DELAY);

cierra();

//PARA APAGAR LA ALARMA HABRA QUE PULSAR LA TECLA \* QUE INICIARA LA VARIABLE RESTART LA CUAL REINICIA LAS VARIABLES AL PUNTO DEL PROGRAMA QUE TE PIDE INTRODUCIR EL PIN PARA VOLVER A ABRIR EL CERROJO

if (teclas == '\*')

{

Apagarled();

RESTART = 1;

}

}

VARIABLE RESTART LA CUAL REINICIA LAS VARIABLES AL PUNTO DEL PROGRAMA QUE TE PIDE INTRODUCIR EL PIN PARA VOLVER A ABRIR EL CERROJO

if (RESTART == 1) //Cerrar y pedir pin

{

cursor = 0;

error = 0;

cont = 0;

cambio = 8;

activar = 0;

incorrecto = 0;

intentos = 0;

i = 6;

lcd.clear();

IntroducePin();

}

RESTART = 0;

}

# FUNCIONES:

## LCD:

1.- INICIA LCD E IMPRIME EL PIN INICIAL:

void IniciarLCD()

{

lcd.init();

lcd.backlight();

lcd.clear();

lcd.display();

lcd.setCursor(2, 0);

lcd.print("PIN INICIAL");

lcd.setCursor(6, 1);

lcd.print("0000");

}

2.- IMPRIME LA OPCION DE INTRODUCIR EL PIN

void IntroducePin()

{

lcd.setCursor(1, 0);

lcd.print("INTRODUCE PIN");

lcd.setCursor(6, 1);

lcd.print("----");

}

3.- LIMPIA LA PARTE DEL LCD EN LA QUE ESTA IMPRESO EL PIN

void clearLCD()

{

lcd.setCursor(0, 1);

lcd.print(" ");

lcd.display();

}

## LEDS:

1.- INICIA LED ROJO UNA VEZ

void LedRojo1()

{

digitalWrite(ledrojo, HIGH);

delay(1000);

color(255, 0);

}

2.- INICIA LED ROJO CONSTANTEMENTE (ALARMA)

void LedRojo()

{

digitalWrite(ledrojo, HIGH);

color(255, 0);

delay(450);

digitalWrite(ledrojo, LOW);

}

3.- INICIA LED VERDE UNA VEZ

void LedVerde()

{

digitalWrite(ledverde, HIGH);

delay(1000);

color(0, 255);

}

4.- APAGA LEDS

void Apagarled()

{

digitalWrite(ledrojo, LOW);

digitalWrite(ledverde, LOW);

}

5.- SELECCIONA EL COLOR DEL LED BICOLOR QUE SE VA A ENCENDER

void color(int valorrojo, int valorverde)

{

digitalWrite(ledrojo, valorrojo);

digitalWrite(ledverde, valorverde);

}

## SERVOMOTOR:

1.- GIRA EL SERVOMOTOR A LA POSICION DE 90º (ABRIR EL CERROJO)

void abrir()

{

servoMotor.write(90);

}

2.- GIRA EL SERVOMOTOR A LA POSICION DE 0º (CIERRA EL CERROJO)

void cierra()

{

servoMotor.write(0);

}

# Bibliografía

Moodle UPM departamento de informática.

Massimo Banzi. Introducción a Arduino. (2016) **ISBN-10:** 8441537445;

**ISBN-13**:978-8441537446.

Página oficial Arduino.ino para consultar librerías usadas.

Página web Prometec <https://www.prometec.net/displays-lcd/>

Canal de Youtube Javier Arcengui <https://www.youtube.com/user/Ivajer>

Página web Programarfacil <https://programarfacil.com/tutoriales/fragmentos/servomotor-con-arduino/>