

# **Sistema de Automatización de Alarma y Luces**

Profesor: Diego Azcurra,

Trabajo para la Cátedra de Sistemas Embebidos, de la Licenciatura en Sistemas en la  
Universidad Nacional de Lanús.  
Lanús 11 de Diciembre del 2017, Buenos Aires, Argentina

Alumno: Diego Cañete, [diisimulalo@yahoo.es](mailto:diisimulalo@yahoo.es)

## **1 El Proyecto**

El siguiente proyecto es un circuito programado que, haciendo uso de Arduino Mega 2560 y unos cuantos componentes o dispositivos electrónicos, nos permite hacer un sistema de alarma y control de luces del hogar. Básicamente el proyecto se programó para poder acceder a través de un control de acceso con clave a una sección del hogar, haciendo uso de un KeyPad 4x4 o un teclado 4x4, un display LCD 20x4, dos diodos LED para dar aviso si la alarma está activada o desactivada, un buzzer como alarma sonora del circuito, un sensor de movimiento PIR y un Relay para prender o apagar las luces del hogar dependiendo del movimiento y de si la alarma está o no activada.

En este documento, se encontrará una sección que habla del desarrollo del hecho sobre Arduino Mega 2560, desde la programación hasta los componentes que se usaron en el mismo; luego también se puede ver un Anexo I donde se ven las imágenes del proyecto terminado y además un diagrama esquemático del mismo. Y para Finalizar, quiero cerrar este documento con un Anexo II donde se puede ver el código desarrollado para hacer posible el sistema de automatización de alarma y luces.

## **2 El Desarrollo**

Para el desarrollo del sistema de automatización de alarma y luces se utilizó un Arduino Mega 2560 haciendo uso de 14 de sus 53 pines digitales dos de los cuales son exclusivamente de comunicación (Pin SDA, Pin SCL) para el protocolo I2C del Display LCD, el cual nos permite usar pocas líneas de control.

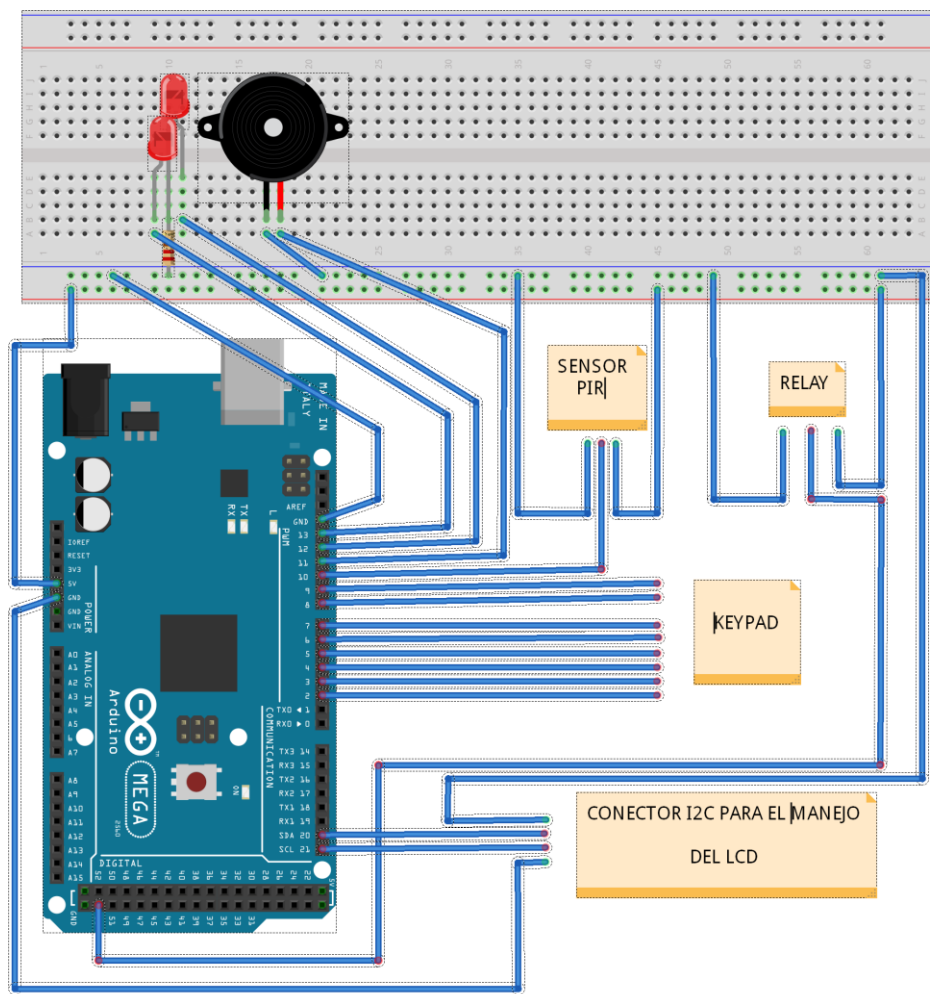
Ahora vamos a comenzar con describir el diagrama esquemático que se encuentra en el Anexo I; en el mismo, básicamente tenemos Arduino Mega 2560, dos diodos LED conectados a los pines 12 y 13 por un lado, y al GND o tierra de la protoboard a

través de una resistencia de 220 ohms por otro lado, un Buzzer también conectado al GND de la protoboard y además al pin 11 de Arduino, un sensor de movimiento PIR conectado al positivo o a los 5v y por otra conexión al GND (ambos de la protoboard), y un tercer conector al pin 10. También, el esquema, cuenta con un Relay ya que me encontré trabajando con un circuito de bajo voltaje (el Arduino) y con otro de alto voltaje (los 220v del enchufe de la lámpara o de las luces del hogar) para automatizar el prendido o apagado de las mismas, este relay tiene de un lado tres terminales, una conectada al GND y otra a los 5v también ambas de la protoboard, y su tercera al pin 53 del Arduino y por otro lado otras 3 conexiones de las cuales conecté 2, una en el medio o polo de la conexión, y otra en el extremo izquierdo o conector Normalmente Abierto, todo esto para dejar pasar los 220v en caso de que sea necesario. Otro componente que use en el circuito es el KeyPad para el control de acceso, que es una matriz 4x4 formada por 8 pines de las cuales los primeros 4 pines son para las filas y los siguientes 4 pines son las columnas del keypad; estos pines se encuentran conectados en los pines del 2 al 9 de mi Arduino. Por último, pero no menos importante, agregue al circuito un Display LCD 20x4 interconectado con un controlador especial que nos permite eliminar la cantidad de líneas de control y reducirlas a solamente 2 líneas para controlar toda la pantalla del Display como tal, esto es gracias al protocolo I2C; este contiene 4 pines de conexión, uno conectado al GND que se encuentra debajo de los 5v del Arduino, el segundo conectado a los 5v de la protoboard y, los dos pines SDA (Serial Data) y SCL (Serial Clock) conectados a los pines 20 y 21.

Con todo lo descripto, lo mostrado en el Anexo I y con toda la programación propuesta en el AnexoII pude realizar este “sistema de automatización de alarma y luces”, el cual muestra el LED en rojo cuando la alarma esta activada; además en estas circunstancias el circuito posee un sensor que permite detectar si existe movimiento en la habitación para dar un aviso a través de un buzzer. De lo contrario cuando la alarma esta desactivada el diodo LED que se enciende es el azul, y cuando el sensor detecta movimiento, se le da aviso al relay para encender las luces de la habitación. La clave de acceso se digita a través de un keypad, siendo la correcta “1A2B3”; el display LCD, como se pueden ver en las imágenes del Anexo I se visualizan tanto los dígitos ingresados de la contraseña como, como los avisos de que el pin fue correcto o erróneo (a su vez acompañado de un código sonoro del buzzer) y, también se puede ver el estado (activado o desactivado) de la alarma.

## Anexo I

Imágenes del desarrollo:



fritzing

Imagen 1.



Imagen 2.



Imagen 3.



**Imagen 4.**



**Imagen 5.**

## Anexo II

//Código del desarrollo

```
#include <Password.h> //Incluyo la libreria Password
#include <Keypad.h> //Incluyo la libreria Keypad
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

Password password = Password("1A2B3"); //Defino el Password
int dlugosc = 5; //Largo del Password
boolean estado = false;
LiquidCrystal_I2C lcd(0x27, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE);

//Creo las Variables de salida
int buzzer = 11;
int redLED = 12;
int blueLED = 13;
int relay = 53;
int sPIR = 10;
int mov = 0;

int ilosc; //Numero de Clicks

const byte ROWS = 4; // Cuatro Filas
const byte COLS = 4; // Cuatro Columnas

// Defino el Keymap
char keys[ROWS][COLS] = {
  {'1','2','3','A'},
  {'4','5','6','B'},
  {'7','8','9','C'},
  {'*','0','#','D'}
};

byte rowPins[ROWS] = { 2, 3, 4, 5 };
byte colPins[COLS] = { 6, 7, 8, 9 };

Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );

void setup() {
  Serial.begin(9600);
  keypad.addEventListener(keypadEvent);
  pinMode(redLED, OUTPUT);
  pinMode(blueLED, OUTPUT);
  pinMode(buzzer, OUTPUT);
  pinMode(relay, OUTPUT);
  pinMode(sPIR, INPUT);

  digitalWrite(relay, HIGH);
  digitalWrite(redLED, HIGH);
  digitalWrite(blueLED, LOW);

  //Seteo los valores del LCD
```

```

    lcd.begin(20, 4);
    lcd.setCursor(4,0);
    lcd.print("*BIENVENIDO*");
    //DIA
    lcd.setCursor(1,1);
    lcd.print(11, DEC);
    lcd.print('/');
    //We print the day
    lcd.print(12, DEC);
    lcd.print('/');
    lcd.print(2017, DEC);
    lcd.print(' ');
    //HORA
    lcd.setCursor(13,1);
    lcd.print(10, DEC);
    lcd.print(':');
    lcd.setCursor(16,1);
    lcd.print(30, DEC);
    lcd.setCursor(2,2);
    lcd.print("ALARMA ACTIVADA!");
    lcd.setCursor(3,3);
    lcd.print("INGRESE SU PIN");
}

void loop() {

    keypad.getKey();
    if(estado) {
        //ALARMA DESACTIVADA
        digitalWrite(redLED, LOW);
        digitalWrite(blueLED, HIGH);
        digitalWrite(buzzer, LOW);

        mov=digitalRead(sPIR);
        delay(50);
        if(mov==HIGH){
            digitalWrite(relay, LOW);
            mov=0;
            delay(4000);
            digitalWrite(relay, HIGH);
        }
    } else {
        //ALARMA ACTIVADA
        digitalWrite(redLED, HIGH);
        digitalWrite(blueLED, LOW);
        //Obligo a apagar las Luces de la casa si la alarma esta
        activada
        digitalWrite(relay, HIGH);

        mov=digitalRead(sPIR);
        delay(50);
        if(mov==HIGH){
            digitalWrite(buzzer, HIGH);
            mov=0;
        }
    }
}
}

```

```

void keypadEvent(KeypadEvent eKey) {
    switch (keypad.getState()) {
        case PRESSED:
            /*
            int i;
            for( i = 1; i <= 1; i++ ) {
                digitalWrite(buzzer, HIGH);
                delay(200);
                digitalWrite(buzzer, LOW);
                delay(100);
            }
            */
            Serial.print("Pressed: ");
            Serial.println(eKey);

            switch (eKey) {
                default:
                    ilosc=ilosc+1;
                    password.append(eKey);
            }

            if(ilosc == 1) {
                lcd.clear();
                lcd.setCursor(5,1);
                lcd.print("<< PIN >>");
                lcd.setCursor(7,2);
                lcd.print("*_");
            }
            if(ilosc == 2) {
                lcd.clear();
                lcd.setCursor(5,1);
                lcd.print("<< PIN >>");
                lcd.setCursor(7,2);
                lcd.print("**_");
            }
            if(ilosc == 3) {
                lcd.clear();
                lcd.setCursor(5,1);
                lcd.print("<< PIN >>");
                lcd.setCursor(7,2);
                lcd.print("***_");
            }
            if(ilosc == 4) {
                lcd.clear();
                lcd.setCursor(5,1);
                lcd.print("<< PIN >>");
                lcd.setCursor(7,2);
                lcd.print("****_");
            }
            if(ilosc == 5) {
                lcd.clear();
                lcd.setCursor(5,1);
                lcd.print("<< PIN >>");
                lcd.setCursor(7,2);
                lcd.print("*****_");
            }
        }
    }
}

```



```

        if(ilosca == dlongosc) {
            delay(250);
            checkPassword();
            ilosc = 0;
        }
    }
}

void checkPassword() {
    if (password.evaluate()) {

        int i;
        for( i = 1; i <= 3; i++ ) {
            digitalWrite(buzzer, HIGH);
            delay(120);
            digitalWrite(buzzer, LOW);
            delay(70);
        }

        ilosc = 0;
        password.reset();

        Serial.println("Correcto");
        lcd.clear();
        lcd.setCursor(1,1);
        lcd.print("<< PIN CORRECTO >>");
        delay(2000);

        if(estado) {
            estado = false;
            AlarmaActivada();
        } else {
            estado = true;
            AlarmaDesactivada();
        }

    } else {

        int i;
        for( i = 1; i <= 1; i++ ) {
            digitalWrite(buzzer, HIGH);
            delay(300);
            digitalWrite(buzzer, LOW);
            delay(100);
        }

        ilosc = 0;
        password.reset();

        Serial.println("Error");

        lcd.clear();
        lcd.setCursor(1,1);
        lcd.print("<< PIN ERRONEO >>");
        delay(2000);

        if(estado) {
            AlarmaDesactivada();
        }
    }
}

```

```

    } else {
        digitalWrite(buzzer, HIGH);
        AlarmaActivada();
    }
}
}

```

```

void AlarmaDesactivada(){
    lcd.clear();
    lcd.setCursor(4,0);
    lcd.print("*BIENVENIDO*");
    //DIA
    lcd.setCursor(1,1);
    lcd.print(11, DEC);
    lcd.print('/');
    //We print the day
    lcd.setCursor(12, DEC);
    lcd.print('/');
    lcd.print(2017, DEC);
    lcd.print(' ');
    //HORA
    lcd.setCursor(13,1);
    lcd.print(10, DEC);
    lcd.print(':');
    lcd.setCursor(16,1);
    lcd.print(30, DEC);
    lcd.setCursor(1,2);
    lcd.print("ALARMA DESACTIVADA");
    lcd.setCursor(3,3);
    lcd.print("INGRESE SU PIN");
}

```

```

void AlarmaActivada() {
    lcd.clear();
    lcd.setCursor(4,0);
    lcd.print("*BIENVENIDO*");
    //DIA
    lcd.setCursor(1,1);
    lcd.print(11, DEC);
    lcd.print('/');
    //We print the day
    lcd.setCursor(12, DEC);
    lcd.print('/');
    lcd.print(2017, DEC);
    lcd.print(' ');
    //HORA
    lcd.setCursor(13,1);
    lcd.print(10, DEC);
    lcd.print(':');
    lcd.setCursor(16,1);
    lcd.print(30, DEC);
    lcd.setCursor(2,2);
    lcd.print("ALARMA ACTIVADA!");
    lcd.setCursor(3,3);
    lcd.print("INGRESE SU PIN");
}

```