

# DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO DE SISTEMA DOMÓTICO CON PROGRAMACIÓN EN ARDUINO

*Autores: Carlos Andrés Pacagui Vargas 23551724501 y David Alexander Celis Rey 23551726144*

*Facultad de Ingeniería Mecánica, Electrónica y Biomédica.*

*Tecnología en mantenimiento electromecánico industrial*

*Universidad Antonio Nariño*

*Villavicencio-Sede Villavicencio*

*[cpacagui79@uan.edu.co](mailto:cpacagui79@uan.edu.co), [dcelis23@uan.edu.co](mailto:dcelis23@uan.edu.co)*

*Director*

*Carlos Arturo Monroy Moyano*

*[Carlosmonroy83@uan.edu.co](mailto:Carlosmonroy83@uan.edu.co)*

**RESUMEN:** Se le llama domótica a los sistemas capaces de automatizar viviendas en sus distintas áreas tales como iluminación, seguridad, comunicaciones y muchas más. En este documento se presenta el control de iluminación, también el control remoto por aplicación y por llamada para la apertura de una chapa electromecánica y de un sistema de alarmas a través de sensores de Gas, Humo y de Movimiento de una casa generando alarmas y un mensaje de texto informándole al usuario la falla generada, para esto se utiliza dos placas de Arduino uno la cual tendrá la respectiva programación para que a través de la programación reciba la señal de los sensores y conlleve a las respectivas alarmas y el correcto envío de mensajes de estas mismas. Este proyecto con respecto a la domótica puede resultar útil en nuestro campo de tecnología para diversos usos que pueden pasar de lo residencial a lo industrial dependiendo las necesidades que se tengan, siempre teniendo los conceptos y bases claras de los equipos que se pueden implementar en los proyectos.

**PALABRAS CLAVE:** Domótica, bluetooth, electroválvulas, Dimmer, chapa electromecánica.

**Abstract:** It is called home automation systems capable of automating housing in its various areas such as lighting, security, communications and many more. In this document present the lighting control, also of the remote control by application and by call for the opening of an electromechanical lock and of an alarm system through sensors of Gas, Smoke and Movement of a house generating alarms and a message of text informing the user of the generated fault, for this use two Arduino one which will have the respective programming so that through the programming it receives the signal from the sensors, and leads to the respective alarms and the correct sending of messages of these same. This project with regard to home automation can be useful in our field of technology for various uses that can go from residential to industrial depending on the needs we have, always having the concepts and clear bases of the equipment that can be implemented in the projects.

**KEYWORDS:** Domotics, Bluetooth, electrovalves, buzzer, electromechanical lock.

## 1. INTRODUCCIÓN

A continuación se muestra el lenguaje de programación de la placa Arduino Uno con respecto al módulo sim y el modulo bluetooth. También se evidencia que al generarse las alarmas de gas, humo y movimiento esta envía un mensaje de texto al usuario con la alarma y se activan los relevos que dan la potencia a los solenoides y buzzer, el control de iluminación y el dimmer. Se muestra el paso a paso o avance en la lógica cableada de l montaje, así como los planos eléctricos especificando las conexiones hechas. El módulo de relevos el cual permite que con señales de 5VCC de las salidas del Arduino se active los solenoides de las electroválvulas que trabajan con 12VDC, la chapa electromecánica que funciona a 12VAC y la iluminación que requiere 120VAC

## 2. MARCO TEÓRICO

En el prototipo se tienen dispositivos, actuadores y sensores a los cuales se realiza una breve descripción.

### 2.1 Tarjeta Arduino

La tarjeta Arduino se utiliza para desarrollar elementos que sean autónomos y que se puedan conectar a diferentes dispositivos para así poder interactuar con el hardware y software. Permite controlar diferentes elementos y actuadores como motores, electroválvulas, bombillos entre otros.

### 2.2 Módulo SIM 900

Este módulo se encarga de transmitir los mensajes de texto a los números de celular inscritos en la tarjeta Arduino y previo procesamiento de la información. Para el envío de la información es necesario y después de la selección del shield de comunicaciones que tuviera compatibilidad en el lenguaje de programación y en rangos de voltajes y corrientes con la tarjeta Arduino al igual que manejara los protocolos de comunicación de telefonía GSM.

### 2.3 Sensor MQ5

El MQ5 es usado para la detección de fugas de gas natural, GLP y en equipos de gas en aplicaciones de consumo y la industria.

Con un potenciómetro se puede ajustar la sensibilidad.

### 2.4 Sensor MQ9

El sensor MQ9 detecta gases de monóxido de carbono. Tiene aplicaciones para detección de fugas de gas doméstico e industrial. Se puede ajustar de acuerdo a la sensibilidad, es muy práctico y se puede utilizar fácilmente en aplicaciones.

### 2.5 Sensor de movimiento PIR

Los sensores de movimiento infrarrojo pasivo (PIR). Detectan algún tipo de cambio en la radiación infrarroja recibida. Una ventaja de este sensor es que no emite radiaciones, sino que las recibe.

### 2.6 Electrovalvulas

Una electroválvula es una válvula electromecánica, prácticamente es usa para controlar el paso de un líquido a través de un tubo. La válvula se mueve mediante una bobina solenoide. Usualmente tiene dos posiciones: abierto y cerrado.

### 2.7 Arduino Bluetooth

El software de Arduino consta con un monitor de puerto serie que accede a los datos de texto simples para ser enviados hacia y desde la placa Arduino través de la conexión bluetooth.

**Tabla 1. Características básicas de la placa arduino Bluetooth**

Voltaje de operación	5V
Voltaje de Entrada (recomendado)	7-12 Vdc
Digital pines I / O	14 (de los cuales 6 proporcionan salida PWM)
Pines de entrada analógica	6
Corriente DC	40 mA
Corriente continúa	3.3V 500 mA (con una fuente de energía capaz 1.5 A)
Corriente DC	5V 1000 mA
Memoria Flash	32 KB( de los cuales 2 KB usados por bootloader)
SRAM	1 KB (ATmega168) o 2 KB (ATmega328)
EEPROM	1KB
Velocidad del reloj	16 MHz.
BT modulo	2.1 WT11i-A-A14

### 3. DISEÑO E IMPLEMENTACIÓN

En el prototipo del sistema domótico se tiene en cuenta la siguiente lista de materiales.

#### 3.1. MATERIALES

- Placa arduino uno
- Módulo de relevos arduino
- Módulo Sim 900
- Modulo Bluetooth
- Módulo Dimmer
- Borneras
- cable de control 18 AWG
- 20 jumper macho
- 40 jumper macho-hembra
- Mdf 50cm x 100cm
- Protoboard
- Adaptador 5VDC
- Adaptador 12VDC
- Adaptador 120VAC
- Sensor de movimiento PIR
- Sensor de gas MQ5
- Sensor de Humo MQ9
- Electroválvulas acople rápido
- Ventilador de 12VDC
- Buzzer 12VDC
- 3 Rosetas y 3 bombillos incandescentes
- Chapa electromecánica de 12VAC
- Accesorios bronce, bushing, acople rápido aire comprimido y tornillería.

#### 3.2 PROGRAMACIÓN ARDUINO UNO, SENSORES Y RELEVOS

La placa Arduino fig. 1 uno es una tarjeta programable la cual es la encargada del control de todo el montaje, esta tarjeta está compuesta por salidas analógicas y digitales las cuales pueden ser controladas como el usuario lo desee dependiendo del código de programación fig.2 que se digite y transfiera.



Figura 1. Placa Arduino uno

Fuente: [www.arduino.cc/](http://www.arduino.cc/).

#### PROGRAMACION ARDUINO BLUETOOTH

```

dimmer1.ino
ReadMe.adoc

1  int pin_estado = 0;
2  int E_Anterior = 1;
3  int E_cero = 0;
4  int T_periodo = 7930;
5  int V_int_control = 7920;
6
7  String Luz_1;
8  String Luz_2;
9  String V_control;
10 String V_reles;
11 String datos;
12
13 void setup() {
14   pinMode(2, INPUT);
15   pinMode(3, OUTPUT);
16   pinMode(4, OUTPUT);
17   pinMode(5, OUTPUT);
18   pinMode(6, OUTPUT);
19   digitalWrite(3,HIGH);
20   digitalWrite(4,HIGH);
21   digitalWrite(6,HIGH);
22   Serial.begin(9600);
23   Serial.setTimeout(500);
24 }
25
26 void loop() {
27   paso_por_cero(); //Llama a la func
28   if (Serial.available()) { //Si hay un dato
29     Leer_bluetooth(); //Se llama a la f
30   } //Si no hay datos
31   Control_luces(); //Se llama la fun
32 }
33
34 /*Funcion que se encarga de detectar cuando la
35 y asi controlar la intensidad de una bombilla m
36 void paso_por_cero() {
37   pin_estado = digitalRead(2);
38   if (pin_estado == 0) {
39     if (E_Anterior == 1) {
40       E_cero = 0;
41     }

```

```

42 }
43 else if (E_Anterior == 0) {
44     E_cero = 0;
45     E_Anterior = 1;
46 }
47 }
48 }
49 /*Funcion encargada de leer los datos del puerto serial,
50 si los datos recibidos son de tipo numerico para luego s
51 void Leer_bluetooth() {
52     V_control = "";
53     V_reles = "";
54     datos = Serial.readString();
55     for (int contador = 0; contador < 2; contador ++ ) {
56         if (isDigit(datos[contador])) {
57             V_control = V_control + datos[contador];
58         } else {
59             V_reles = V_reles + datos[contador];
60         }
61     }
62     if(V_control != ""){
63         V_int_control = V_control.toInt();
64         V_int_control = map(V_int_control, 0, 99, 7920, 10);
65     }
66 }
67
68 /*Funcion encargada de separar y guardar los estado de
69 ademas, podra cambiar la intensidad de la bombilla 4 */
70 void Control_luces() {
71
72 /*-----
73 Aqui separamos y guardamos en nuevas variable el valor d
74 para recibir nuevos datos, si no se hace esto podria cua
75 -----
76     if (V_reles=="LU"){
77         Luz_1=V_reles;
78     }
79     else if (V_reles=="Lu"){
80         Luz_1=V_reles;
81     }
82     else if (V_reles=="LD"){

```

```

82     else if (V_reles=="LD"){
83         Luz_2=V_reles;
84     }
85     else if (V_reles=="Ld"){
86         Luz_2=V_reles;
87     }
88     else if (V_reles=="PU"){
89         digitalWrite(6,LOW);
90         delay(500);
91         digitalWrite(6,HIGH);
92         V_reles="";
93     }
94
95     if (Luz_1=="LU"){
96         digitalWrite(3,LOW);
97     }else if (Luz_1=="Lu"){
98         digitalWrite(3,HIGH);
99     }
100
101     if (Luz_2=="LD"){
102         digitalWrite(4,LOW);
103     }else if (Luz_2=="Ld"){
104         digitalWrite(4,HIGH);
105     }
106

```

```

107     if (V_int_control < 7920) {
108         if (digitalRead(2)==0) {
109             delayMicroseconds(V_int_control);
110             digitalWrite(5, HIGH);
111             delayMicroseconds((T_periodo - V_int_control));
112             digitalWrite(5, LOW);
113             E_cero = 1;
114         }else {
115             digitalWrite(5, LOW);
116         }
117     }else{
118         digitalWrite(5, LOW);
119     }
120 }
121

```

### PROGRAMACIÓN SIM 900

	SEGURIDAD_UNO.ino	ReadMe.adoc
<pre> 1 #include &lt;SoftwareSerial.h&gt; 2 SoftwareSerial SIM900(7,8); 3 4 int cerradura = 3; 5 int rlbuzzer = 4; 6 int rlgas = 5; 7 int rlagua = 6; 8 int sensorhumo = 10; 9 int sensorpir = 11; 10 int sensorgas = 12; 11 12 String Numeros_A[] = {"3165391510", "3114673963", "322589764"}; 13 int numtotal = 3; //Cantidad de numeros de telefono autorizados 14 String Llamada_M = ""; 15 16 String init_sms = "AT+CMGF=1\r\n"; 17 String num_sms = "AT+CMGS=\"+573114673963\"\r\n"; 18 String sms_1 = "El sensor detecto movimiento de un Intruso!!!\r\n"; 19 String sms_2 = "El sensor ha detectado humo!!!\r\n"; 20 String sms_3 = "El sensor ha detectado una fuga de gas!!!\r\n"; 21 char fin_sms = 26; 22 23 void setup() { 24     pinMode(sensorpir,INPUT); 25     pinMode(sensorhumo,INPUT); 26     pinMode(sensorgas,INPUT); 27     pinMode(9,OUTPUT); 28     pinMode(13,OUTPUT); 29     pinMode( cerradura, OUTPUT); // cerradura como sali 30     digitalWrite(cerradura,HIGH); 31     pinMode(rlbuzzer,OUTPUT); 32     digitalWrite(rlbuzzer,HIGH); 33     pinMode(rlagua,OUTPUT); 34     digitalWrite(rlagua,HIGH); 35     pinMode(rlgas,OUTPUT); 36     digitalWrite(rlgas,LOW); 37     Serial.begin(9600); 38     SIM900.begin(9600); 39     SIM900.setTimeout(100); 40     SIM900.print("AT+CLIP=1\r\n"); // Activa la identifica 41     digitalWrite(9,HIGH); 42     delay(1000); 43     digitalWrite(9,LOW); 44     digitalWrite(13,HIGH); 45     delay(3000); 46     digitalWrite(13,LOW); 47 } 48 49 void loop() { 50     if (SIM900.available()&gt;0) 51     { 52         Llamada_M=SIM900.readString(); // lee e identifica 53         identifica_llamada (); 54     } 55     if(digitalRead(sensorpir) == HIGH){ 56         digitalWrite(rlbuzzer,LOW); 57         SIM900.print(init_sms); 58         delay(1000); 59         SIM900.print(num_sms); 60         delay(1000); 61         SIM900.print(sms_1); 62         delay(1000); 63         SIM900.print(fin_sms); 64         delay(1000); </pre>		

```

55 ▾ if(digitalRead(sensorpir) == HIGH){
56     digitalWrite(rlbuzzer,LOW);
57     SIM900.print(init_sms);
58     delay(1000);
59     SIM900.print(num_sms);
60     delay(1000);
61     SIM900.print(sms_1);
62     delay(1000);
63     SIM900.print(fin_sms);
64     delay(1000);
65     while(digitalRead(sensorpir) == HIGH){}
66     digitalWrite(rlbuzzer,HIGH);
67
68 ▾ }else if(digitalRead(sensorhumo) == LOW){
69     digitalWrite(rlagua,LOW);
70     SIM900.print(init_sms);
71     delay(1000);
72     SIM900.print(num_sms);
73     delay(1000);
74     SIM900.print(sms_2);
75     delay(1000);
76     SIM900.print(fin_sms);
77     delay(1000);
78     while(digitalRead(sensorhumo) == LOW){}
79     digitalWrite(rlagua,HIGH);
80
81 ▾ }else if(digitalRead(sensorgas) == LOW){
82     digitalWrite(rlgas,HIGH);
83     SIM900.print(init_sms);
84     delay(1000);
85     SIM900.print(num_sms);
86     delay(1000);
87     SIM900.print(sms_3);
88     delay(1000);
89     SIM900.print(fin_sms);
90     delay(1000);
91     while(digitalRead(sensorgas) == LOW){}
92     digitalWrite(rlgas,LOW);
93 }

```

```

94 }
95
96 void identifica_llamada(){
97     int finllamada = Llamada_M.indexOf("RING");
98     if (finllamada >= 0)
99     {
100         for ( int i = 0; i < numtotal; i++)
101         {
102             int pos = Llamada_M.indexOf(Numeros_A[i]);
103             if (pos >= 0)
104             {
105                 activar_puerta();
106             }
107         }
108         Llamada_M = "" ; //Bórralo para la próxima ve
109     }
110 }
111
112 void activar_puerta() {
113     digitalWrite(cerradura, LOW);
114     delay(500);
115     digitalWrite(cerradura, HIGH);
116 }

```

Figura 2. Lenguaje de programación Arduino  
Fuente: Los Autores

Para el proyecto se usó salidas digitales activadas desde el arduino y desde mensajes de texto fue con el módulo sim 900 (fig. 3) y el modulo bluetooth (fig. 4). Además se utilizaron los siguientes sensores; el sensor de movimiento PIR (fig. 5), el sensor de gas MQ5 (fig. 6) y el sensor de humo MQ9 (fig. 7). Los cuales envían las señales de entrada al módulo arduino para activar o desactivar las salidas digitales. Estas salidas digitales también controlaran el módulo de relevos (fig. 8) activando y desactivando cada uno dependiendo de la señal digital controlada. De igual forma el Dimmer (fig. 9) que se controla mediante la aplicación del celular.



Figura 3. Modulo Sim900

Fuente: <https://www.prometec.net/gprs-llamar-enviar-sms/>

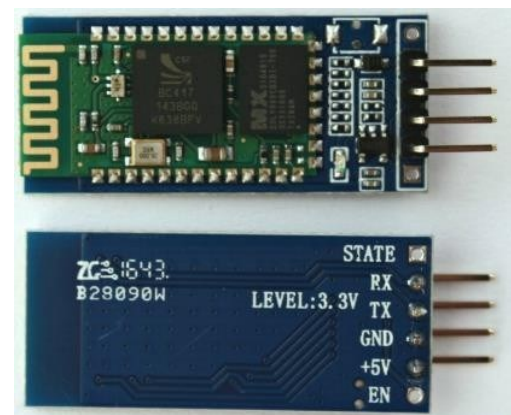


Figura 4. Modulo Bluetooth.

Fuente: <https://amiyoled.es/>



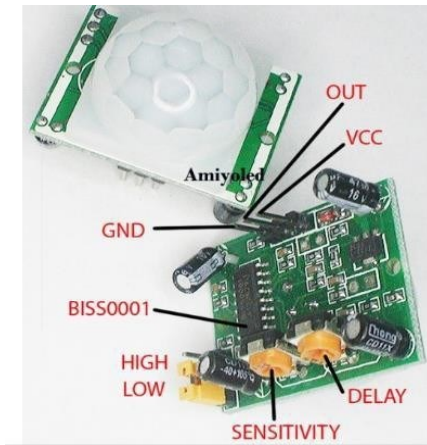


Figura 5. Sensor de movimiento PIR.  
Fuente: <https://amiyoled.es/>



Figura 6. Sensor de Gas natural MQ5.  
Fuente: <https://www.luisllamas.es/arduino-detector-gas-mq/>



Figura 7. Sensor de Humo o Gas inflamables MQ9.  
Fuente: <https://amiyoled.es/64-arduino-electronica>



Figura 8. Módulo de relevos.  
Fuente: <https://store.prometec.net/>

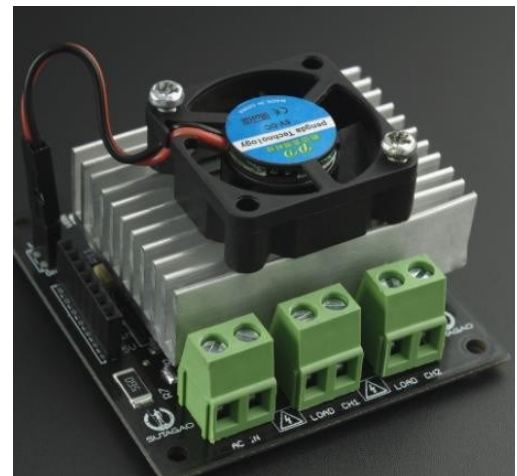


Figura 9. Módulo Dimmer.  
Fuente: <https://amiyoled.es/>

Para la creación de la aplicación móvil se realizó en la página Mit app Inventor donde se diseña la forma de imagen o plantilla que saldrá de presentación en la app y mediante de una programación por bloques (fig. 10) se configura los distintos botones y las funciones que se quieren realizar conectado a la arduino por medio de bluetooth y coordinado con la programación hecha en arduino y así crear una comunicación eficaz y por medio de la cual se ejecuten las salidas que se desean controlar por medio del móvil (fig. 11).

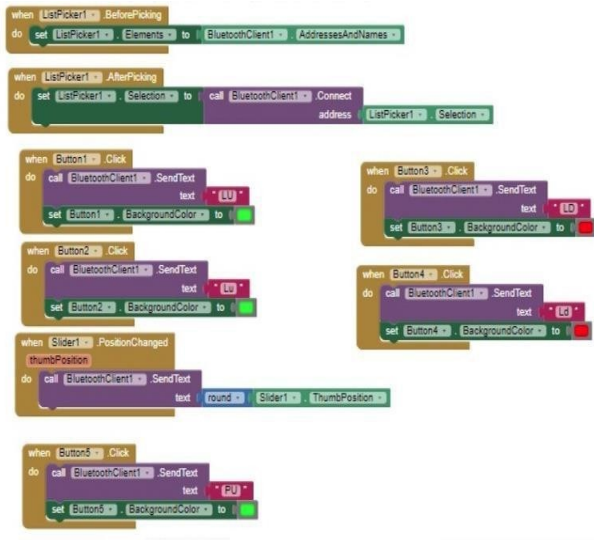


Figura 10. Lenguaje de programación por bloques app inventor.

Fuente: Los Autores

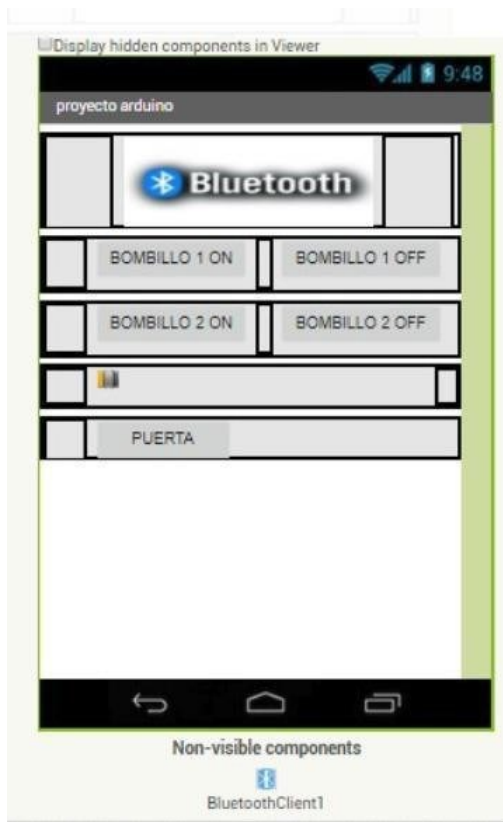


Figura 11. Aplicación vista desde móvil.

Fuente: Los Autores

### 3.3 MONTAJE

El prototipo domótico se divide en dos etapas importantes: la de control y la de potencia, además del montaje para la estructura y soporte de las alarmas y sensores.

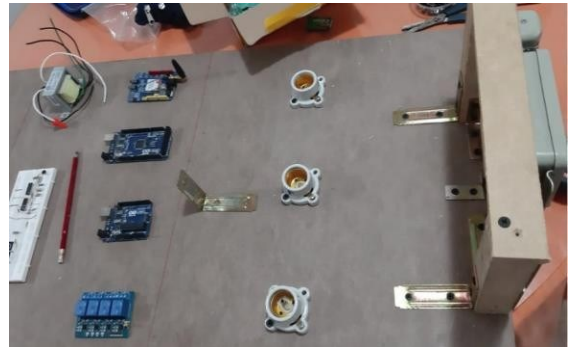


Figura 12. Elementos requeridos

Fuente: Los Autores



Figura 13. Montaje accesorios sistema hidráulico y neumático

Fuente: Los Autores



Figura 14. Montaje de chapa electromecánica y accesorios de soporte.

Fuente: Los Autores

Esta etapa estructural consta de los soportes, que soportan el montaje del sistema electrónico y el sistema actuador cuyo caso se usa electroválvulas. Para enviar la señal a la electroválvula se utilizara un transformador de entrega 12VDC. Así, como un transformador de 12VAC para la activación de este solenoide. De la figura 15 al 21 la fuente son los Autores.



Figura 15. Iluminación, sensores y chapa electromecánica.

Estos relés son controlados desde la placa arduino que tienes ya el código de programación subido, y que es controlado por la información que se genera de la programación y de los sensores, donde también se comunica con el módulo sim 900 y el módulo bluetooth.



Figura 18. Circuito de potencia 12VDC Y 12VAC

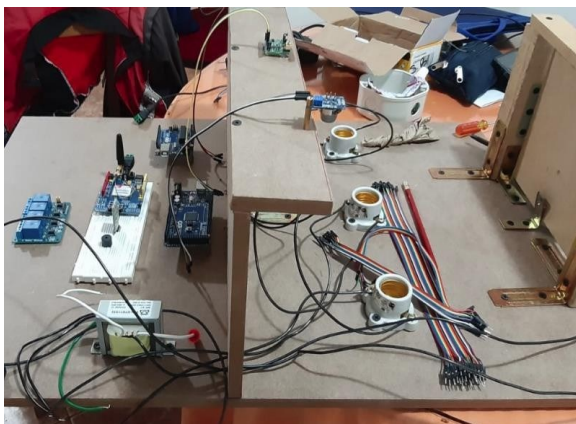


Figura 16. Posicionamiento de elementos de control y potencia

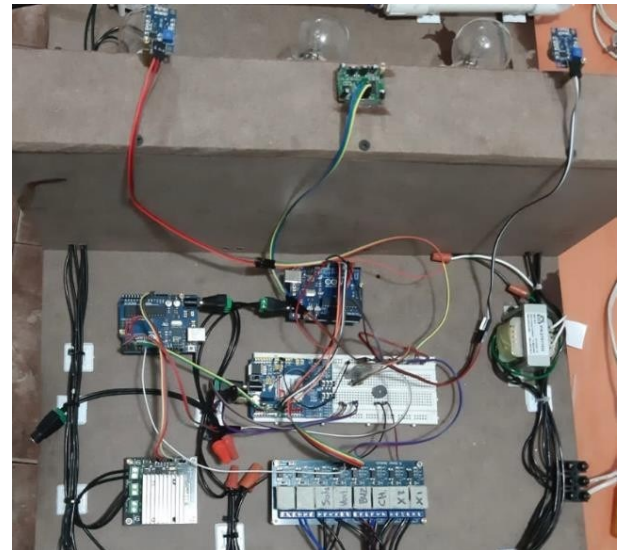


Figura 19. Circuito de control



Figura 17. Módulo de relés de potencia.

Para este proyecto se diseñó un circuito eléctrico o lógica cableada entre los componentes y tarjetas electrónicas programables para el cual se diseñó previamente el diagrama eléctrico (fig. 20-21).



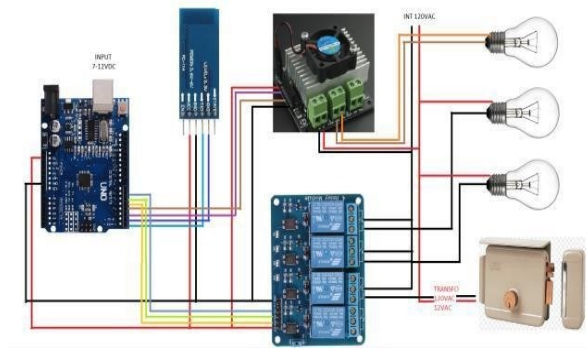


Figura 20. Diagrama eléctrico de control de gas, humo y movimiento

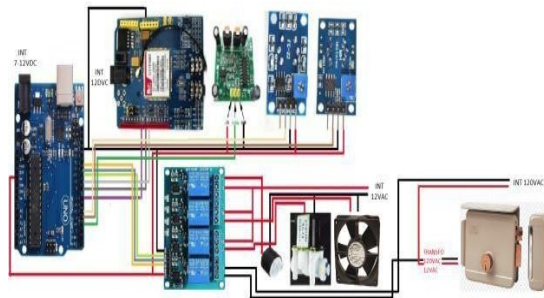


Figura 21. Diagrama eléctrico sensores de gas, movimiento.

#### 4. CONCLUSIONES

- Los avances tecnológicos de la mano con los sistemas domóticos, logran enormes beneficios para advertir a las personas de peligros que pueden correr frente a condiciones como un incendio, una fuga de gas e incluso a un intruso que no solo puede causar daño a la persona en sí, sino también a las cosas materiales y bienes.
- Con la realización del montaje de un sistema de seguridad contra-intrusos con sensor de movimiento se generó la alerta cuando se detectó una variación y se envió un mensaje de texto al número de celular registrado.
- Se generaron las alarmas contra fuga de gas domiciliario y alarma contra incendio, mediante método de detección, enviando un mensaje de texto al número celular registrado y que adicional envió señal para abrir o cerrar los actuadores de línea de gas y agua.
- Con la realización de este proyecto, no solo se logró los objetivos planteados, sino que también se adquirió conocimientos de suma importancia que

pueden ser de gran utilidad en la vida diaria, para la protección personal, la protección de las personas cercanas, las empresas en las cuales se labora, los sistemas domóticos y de seguridad de la mano con las nuevas tecnologías pueden salvar vidas.

#### 5. RECOMENDACIONES

- En el campo académico se realizan proyectos que pueden ser implementados y comercializados, se sugiere incentivar estas iniciativas las cuales pueden servir como emprendimientos y fuentes de trabajo.
- Para una implementación real del sistema domótico se recomienda realizar la instalación en los hogares con las debidas normas técnicas y con el personal capacitado.

#### 6. REFERENCIAS

- Design Themes.2020 “MÓDULO GSM/GPRS: Llamar y Enviar SMS.” Tienda y Tutoriales Arduino, [www.prometec.net/gprs-llamar-enviar-sms/](http://www.prometec.net/gprs-llamar-enviar-sms/).
- Monarca Electrónica, 2020. <https://www.monarcaelectronica.com.ar/producto/m%20q5-modulo-sensor-glp-y-gas-natural-arduino-mona/>
- Amiyoled, Mayo 2020. <https://amiyoled.es/>
- “Home.” Arduino 2020. [www.arduino.cc/](http://www.arduino.cc/).
- Luis Llamas, 30 Nov. 2017,Luis.“Detector De Gases Con Arduino y La Familia De Sensores MQ.” [www.luisllamas.es/arduino-detector-gas-mq/](http://www.luisllamas.es/arduino-detector-gas-mq/).
- Mit app inventor, 2012-2020 Massachusetts, technology.<http://web.mit.edu/>
- López Frías, G. D. J. (2013). Diseño e implementación de software y hardware de un registrador de variables eléctricas con comunicaciones Ethernet basado en tecnología Arduino y sistema de supervisión

HMI (Doctoral dissertation,  
ATACUNGA/ESPE/2013).

- Chandi, C. X. R., & Orellana, C. F. V. (2018). Metodología para evaluación de usabilidad del entorno de desarrollo integrado de Arduino. *SATHIRI*, 13(1), 214-226.
- Thomas Erviti, G. (2013). Diseño y test de una mesa de mezclas analógica con efectos sonoros usando Arduino.
- Roberto Hernández Sanpieri, Carlos Fernández Collado, Pilar Baptista Lucio, (2010), Metodología de la Investigación, (5ª Edición), McGraw-Hill, Madrid.
- De la Rosa Flores, R., Morales, D. E. M., Mendoza, E. P., Flores, J. L. M., & Zacatelco, H. C. (2018). AUTOMATIZACIÓN PARA EL CONTROL DE ACCESO UTILIZANDO DISPOSITIVOS MÓVILES Y RFID.
- López, C. I. Y., Laureano, E. V., Benítez, R. F., & González, M. A. P. (2019). Activación de cargas eléctricas a través de comandos de voz vía módulo HC-05 y Arduino. *ConCiencia Tecnológica*, (57), 5-11.