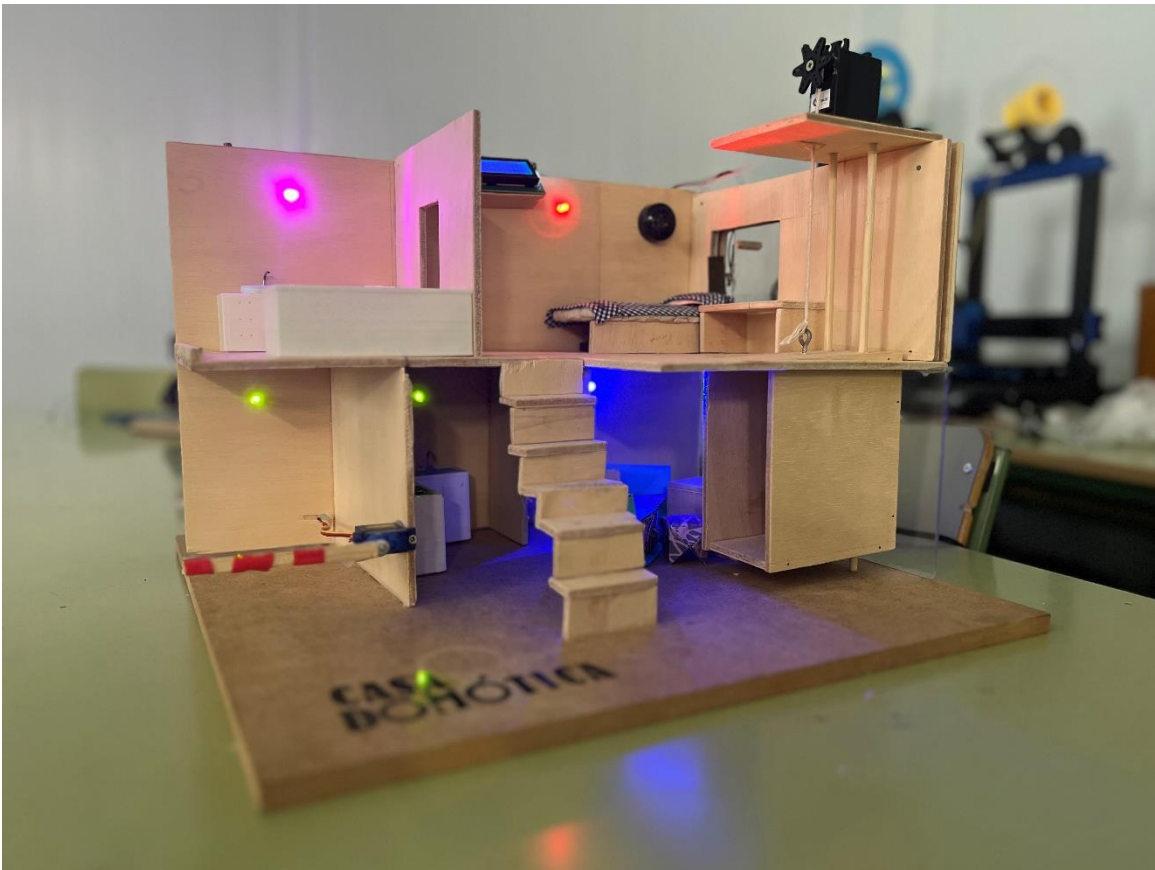


CASA DOMÓTICA (CADI)
CASA AUTOMATIZADA DOMÓTICA INTELIGENTE



PROYECTO REALIZADO POR: **MARCOS RODRIGUEZ, ERNESTO PORCAR, ADRIÁN PLANA PIÑANA**

TUTORIZADO POR: **ELENA GIL y CAROLINA CABALLERO**

IES LA PATACONA (ALBORAYA-VALENCIA)

Índice

1)	Resumen	3
2)	Introducción	6
3)	Objetivos.....	6
4)	Materiales y métodos empleados	6
5)	Resultados y análisis.....	19
6)	Conclusiones.....	19
7)	Agradecimientos.....	20
8)	Bibliografía.....	20

1) Resumen



Ilustración 1. Diseño en SketchUp de nuestra casa

Este proyecto se trata de una casa domótica controlada mediante un mando a distancia que funciona por infrarrojos. La casa consta de cuatro habitaciones y un garaje, con una luz LED en cada estancia, es decir, cuenta con un total de 5 LEDs. Además, uno de los LEDs de la vivienda puede ser activado a través de un circuito con una LDR (Light Dependent Resistance), haciendo que se apague o se encienda en función de la luz ambiental.

En cuanto a los diferentes motores que se utilizan en la casa, destacamos dos tipos: la barrera del garaje, que puede subir y bajar gracias a un servomotor posicional y el ascensor que sube y baja entre los dos pisos, mediante un servomotor continuo.

Por otro lado, hay una ventana conectada a un sensor de contacto que, si está activado y la ventana abierta, suena una alarma hasta que la ventana se cierre o hasta desactivarla manualmente mediante el mando.

Por último, hay una LCD con varias funciones, la principal es indicar la acción que se está realizando en ese momento, como, por ejemplo, cuando se activa la alarma, que en el display indicaría "Alarma activada". Además, también hemos implementado otras funciones de entretenimiento como el conocido juego "Simón dice".

- **LEDs:**
 - Circuitos de iluminación: Cada luz LED se enciende y se apaga desde su respectivo pulsador en el mando.



Ilustración 2 Leds disponibles en el taller

- Circuito con LDR: Una de las habitaciones cuenta con un circuito con una LDR que, al ser activada, enciende o apaga el LED dependiendo de la luz ambiental.

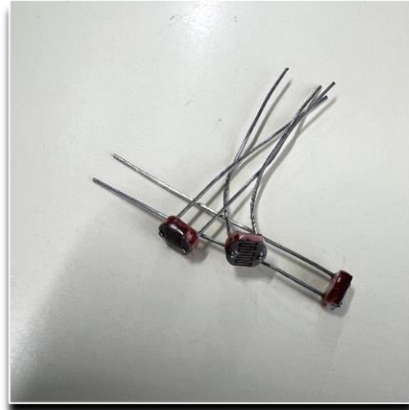


Ilustración 3 LDRs para la maqueta

- **Servomotores:**

- Servo Posicional: La barrera de acceso al garaje está controlada por un servo posicional cuya posición se programa por grados.



Ilustración 4. Servomotor SG-90

- Servo Continuo: El ascensor está controlado por un servo continuo, para que comience a moverse hay que pulsar un botón en el mando y para que se detenga hay que volver a pulsar este mismo botón.



Ilustración 5. Servo de rotación continua

- **Buzzer:**

- Alarma: Una ventana está directamente conectada al buzzer de manera que, al estar la ventana abierta y la alarma activada, esta comienza a sonar. Al desactivar la alarma y/o cerrar la ventana el buzzer deja de sonar.

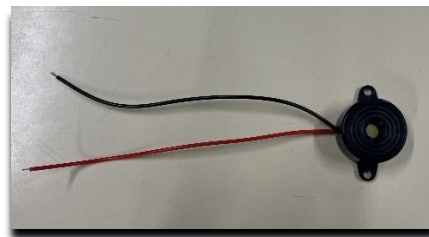


Ilustración 6 Buzzer disponible en nuestro taller

- Música: Una función extra es la posibilidad de introducir un código específico que hace sonar una de las diferentes melodías, que se han implementado en nuestro código de Arduino.

- **LCD:**

- Función Principal: Principalmente, la LCD indica en todo momento la acción que se está realizando, ya sea encender un led, subir o bajar la barrera del garaje, reproducir una canción, activar o desactivar la alarma, etc.
- Contraseñas y códigos extra: Hay diversos códigos que se pueden introducir para realizar acciones extra como una contraseña para subir o bajar la puerta del garaje.
- Juegos: También se puede jugar al clásico Simón dice con las diferentes funciones de la casa. Se puede jugar en nivel fácil o difícil.

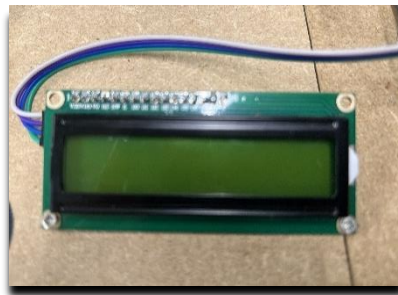


Ilustración 7 Display LCD

2) Introducción

La domótica es un sistema eléctrico centralizado que permite **automatizar y controlar** cualquier espacio o vivienda de manera inteligente y remota. En nuestro caso, el control se lleva a cabo desde un mando de infrarrojos. En un principio, tan solo íbamos a implementar a nuestra casa unas funciones básicas como las luces en cada habitación, un ascensor y la barrera del garaje. Conforme fuimos avanzando nos dimos cuenta de que podíamos mejorar nuestro proyecto, añadiendo funcionalidades como; una LDR que nos permite determinar la luz ambiental y de esa forma poder ahorrar energía, ya que, cuando hay luz exterior el led se apagaría. También añadimos una alarma, que está conectada a una ventana, para simular robos, haciendo que esta suene al abrirse. Por último y no menos importante, implementamos un sistema automatizado, que consiste en un display LCD que nos informa cada vez que realizamos una función de la casa. Esto incluye muchas otras funciones como; reproducir una canción introduciendo un código, bloquear la barrera del garaje de forma que sería necesaria una contraseña para subirla y bajarla, o incluso algunos juegos clásicos como el Simón dice.

3) Objetivos

La casa domótica persigue diferentes objetivos cruciales para una vida más fácil gracias a la automatización de algunas tareas. Estos objetivos son los siguientes; un ahorro energético gracias a la LDR, una facilitación del control de la casa para la gente con movilidad reducida, ya que la casa se puede controlar de forma remota. Además, ésta no dispone de escaleras, sino de un ascensor que permite movilidad libre por la vivienda para personas con movilidad reducida. Por supuesto, tampoco nos olvidamos del entretenimiento y por eso incluimos juegos y códigos secretos como el Simón dice.

4) Materiales y métodos empleados

Material	Cantidad	Precio unitario	Precio total
Madera contrachapada	100x30x0,5 cm	9,80 €/m ²	4,20 €
Madera DM	40x40x1 cm	3,04 €/m ²	0,48€
Servomotor posicional	1	4 €	4 €
Servomotor continuo	1	8,35 €	8,35 €
LED	5	0,025 €	0,125 €

Material	Cantidad	Precio unitario	Precio total
Sensor IR	1	1 €	1 €
Buzzer	1	2 €	2 €
LCD	1	12 €	12 €
Cables	1 mazo	0,08 €	4 €
Resistencias	8	0,02 €	0.16 €
Láminas de cobre pequeñas	4,5x1,5x0,05 cm	39 €/m ²	0,02 €
Metacrilato	8x7x0,3 cm	33,88 €/m ²	2,60 €
LDR	1	0,47 €	0,47 €
Placa protoboard	1	4,96 €	4,96 €
Listones de madera	Varios	9 €	9 €
Arduino Mega	1	42 €	42 €
TOTAL			95.37€

Ilustración 8 Presupuesto

Lo primero que hicimos para empezar el proyecto fue hacer posible la programación y utilización de todos los componentes electrónicos en la casa domótica, para esto hemos tenido que aprender a programar todas las funciones de la casa por separado siendo el proyecto un conjunto de las prácticas. Estas son las siguientes y como es obvio van de menos dificultad a mayor dificultad.

Práctica 1:

Como hemos comentado anteriormente la casa consta de 5 LEDs, un diodo Led es un diodo que cuando está polarizado directamente emite luz. Polarizado directamente quiere decir que la patilla larga del diodo (positivo) irá conectada al polo positivo de la pila o batería y la patilla corta (negativo) se conectará al polo negativo de la pila. Un LED no soporta bien voltajes superiores a 3 V y, para evitar que se queme, se coloca siempre una resistencia en serie con el diodo, es decir, a continuación del diodo, en la misma rama de un circuito.

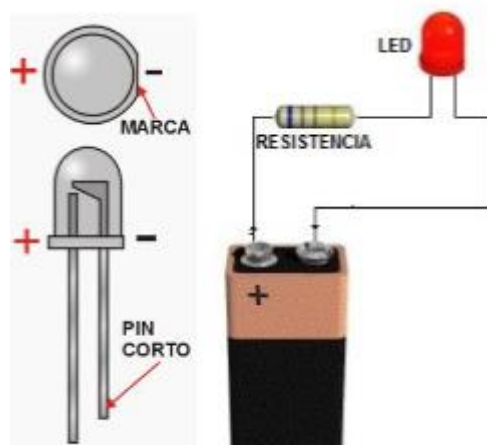
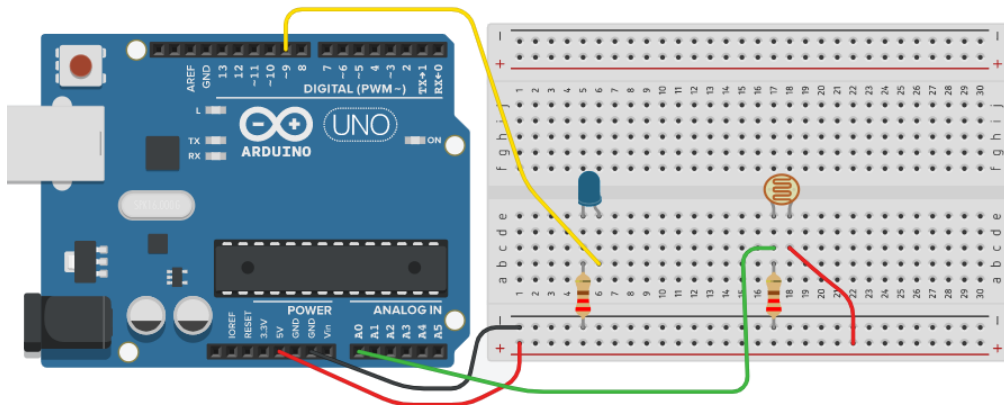


Ilustración 9 Práctica 1 del manual de clase

Práctica 2:

En la segunda práctica aprendemos a programar la LDR, la LDR es un sensor de luz cuyo funcionamiento es muy simple: tiene una resistencia interna que cambia de valor en función de la luz que recibe, a menos luz, mayor es su resistencia a que pase la corriente eléctrica, esta se conecta a una entrada analógica (A0, A1...) ya que, según la luz, el valor de entrada variará entre 0 y 1023.



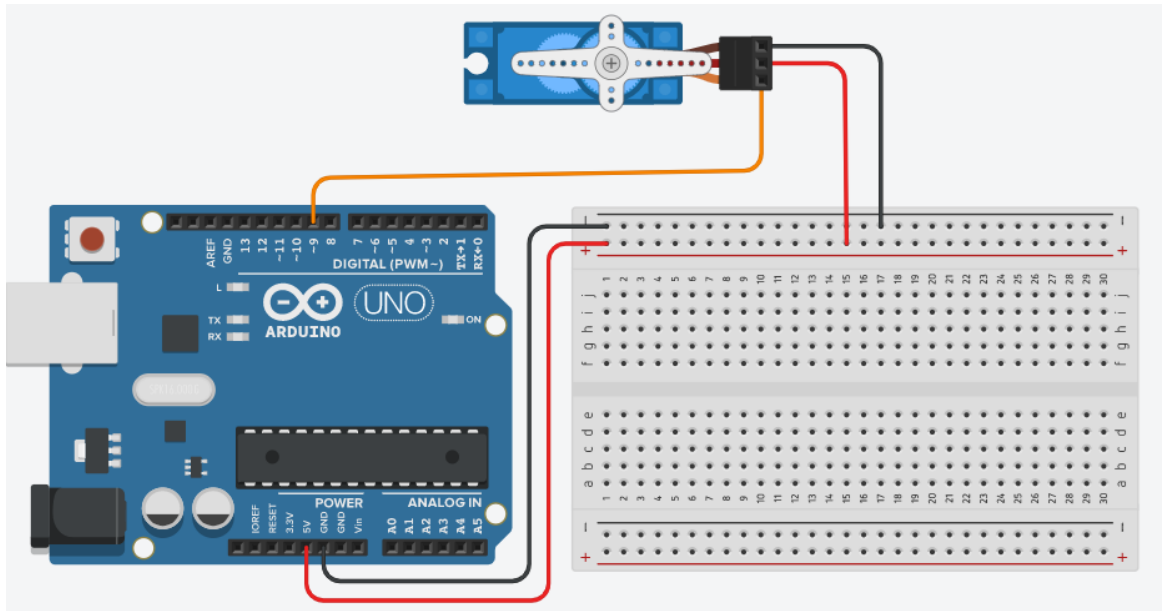
PROGRAMA :

```
int LED_pin=9;
int LDR_pin=A0;
int Valor_LDR = 0;
void setup() {
  Serial.begin(9600);
  pinMode(LDR_pin, INPUT);
  pinMode(LED_pin, OUTPUT); //debe ser una salida PWM
}
void loop() {
  Valor_LDR = analogRead(LDR_pin); // lee el valor del sensor
  Serial.println(Valor_LDR); // imprime la lectura del sensor permitiendo conocer su rango
  // mapea la lectura del sensor(0 a 1023) a un rango para el Led (0 a 255)
  analogWrite(LED_pin, !map(Valor_LDR, 0, 1023, 0, 255));
  delay(100); // Espera 100 milisegundos
}
```

Ilustración 10 Práctica 2 del manual

Práctica 3:

Ahora pasamos a la programación de los servomotores tanto el continuo (ascensor) como el posicional (barrera del garaje). Internamente un servo está constituido por un motor de corriente continua, acoplado a un reductor para reducir la velocidad de giro, junto con la electrónica necesaria para controlar su posición (servos de 180º) o para regular su velocidad (servos de 360º o giro continuo). Los servos admiten una tensión de alimentación entre 4,8V a 7,2V, siendo el valor más adecuado el de 6V. Con tensiones inferiores el motor tiene menos fuerza y velocidad. Con tensiones superiores a 6,5V los servos empiezan a vibrar demasiado, lo cual los hace poco útiles.



Color de los cables:

- Señal (a cualquier pin digital de Arduino): puede ser amarillo, naranja o blanco
- Vcc: es rojo
- GND: puede ser negro o marrón

b) PROGRAMAS:

-Programa 1: Mueve el servo una única vez desde 0 a 180 ° para volver enseguida a 0°. NOTA: debéis modificar int time

```
#include <Servo.h>
Servo myservo; // crea el objeto servo
int time = 500; //variable tiempo
void setup() {
  myservo.attach(9);
  myservo.write(0);
  delay(100); //le damos un tiempo pequeño para que vaya a 0°
  myservo.write(180); //va a la posición 180°
  delay(time); //tiempo para llegar a la posición 180°
  myservo.write(0); //vuelve a la posición 0°
  delay(time); //tiempo para llegar a la posición 0°
}
void loop() { }
```

Responde:

a) El eje del servo necesitará un tiempo para llegar de 0 a 180° (delay).

El tiempo dado de 500 ms no será suficiente para ello. Debéis averiguar que valor en ms es el adecuado.

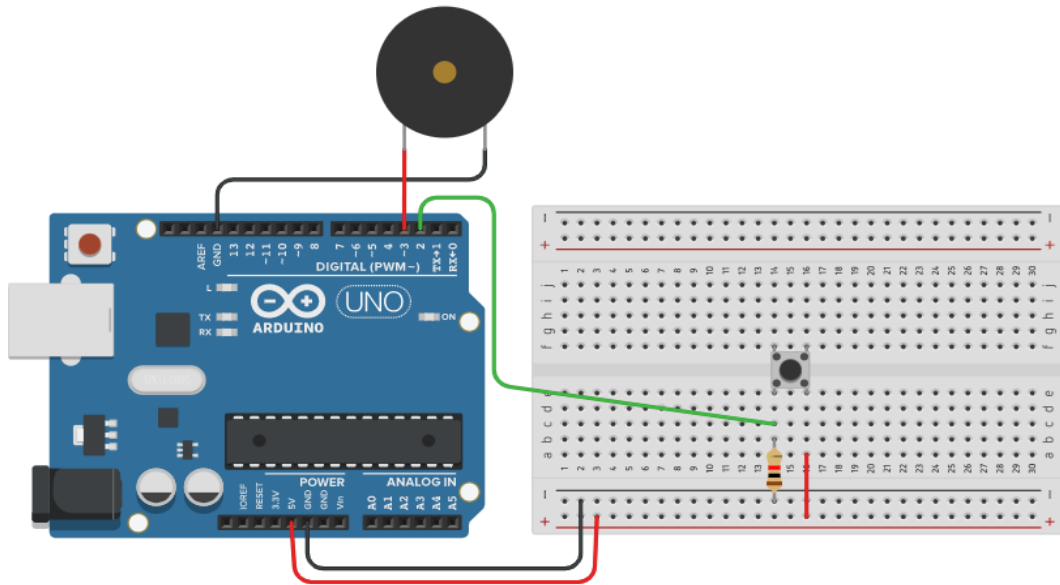
Tiempo: ms

b) Se ha puesto el servo en un pin PWM. Prueba a ponerlo en un pin digital (no PWM), cambia el pin en myservo.attach(); y si funciona:

Ilustración 11 Práctica 3 del manual

Prácticas 4 y 5:

Ahora pasamos al aprendizaje de como programar el pulsador y la alarma (buzzer), son interruptores momentáneos que cierran un circuito cuando son presionados. Los dos modelos de la derecha, con 4 pines, se colocan con facilidad sobre la placa de pruebas. Se utilizan para abrir o cerrar el paso a una señal. Respecto al buzzer es un dispositivo que permite convertir una señal eléctrica en sonido. Emite un sonido o zumbido en un mismo tono. Está polarizado, lo cual quiere decir que hay una patilla positiva (cable rojo, señalada con + o Vcc) y otra negativa (cable negro, sin señal o Gnd). Si se conecta mal no sonará y habrá que cambiarlo de posición.



b) PROGRAMA:

```
int BOTON_pin = 2; // pin del botón
int BUZZER_pin = 3; // pin del buzzer
int Valor_BOTON;
void setup() {
  pinMode(BOTON_pin, INPUT); // pin del botón como entrada
  pinMode(BUZZER_pin, OUTPUT); // pin del buzzer como salida
}
void loop() {
  Valor_BOTON = digitalRead (BOTON_pin)
  if (Valor_BOTON == HIGH) { // si es HIGH, el botón esta oprimido
    for(int i=0 ; i<5 ; i++)
      digitalWrite (BUZZER_pin, HIGH);
      delay(500);
      digitalWrite (BUZZER_pin, LOW);
      delay(500);
  }
}
```

Ilustración 12 Prácticas 4 y 5 del manual

Práctica 6:

Y por último y más importante el mando infrarrojo el cual funciona mediante código sexagesimal (0x) comunicándose con el sensor IR, de esta forma le podemos dar las órdenes a la casa domótica.

Nuestros controles para la casa son:

0: LDR

1,2,3,4,5: LEDs

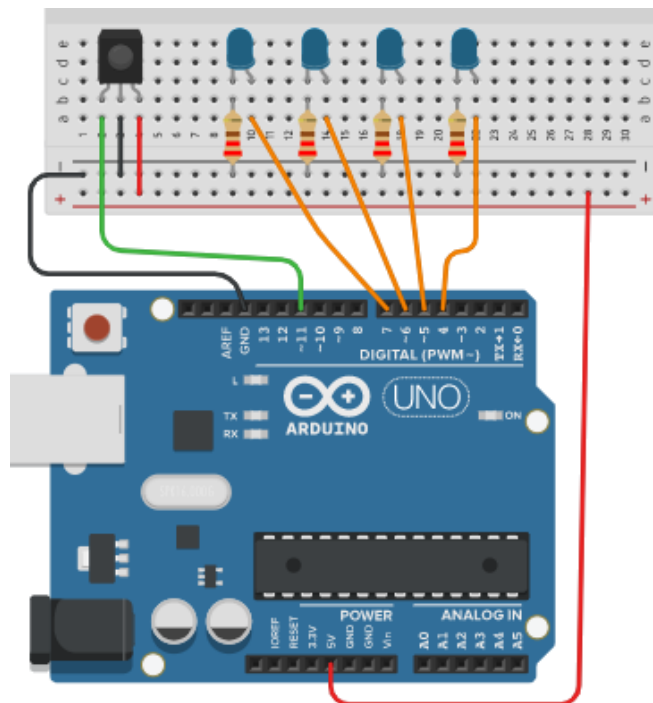
6: Puerta (con password 1234)

7: Bloquear puerta

8: Activar/desactivar alarma

9: Subir/parar/bajar ascensor

OK: Códigos cosas extra



b) PROGRAMA:

/* Control de cada pin con la tecla del mando con el número del pin. Al darle a la tecla se cambia el estado de dicho pin, de modo que se puede encender y apagar cada led al apretar sucesivamente la misma tecla */

#include <IRremote.h> //incluir librería

int LED_pin4=4;

int LED_pin5=5;

int LED_pin6=6;

int LED_pin7=7;

int IR_PIN = 11; //pin de señal del receptor IR

IRrecv sensorIR(IR_PIN); //crea el objeto para el receptor IR, en el pin especificado

decode_results codigoIR; // crea un lugar donde guardar los datos recibidos por el sensor

void setup()

{

Serial.begin(9600);

```

sensorIR.enableIRIn(); //inicializa la recepción de datos por IR
pinMode(LED_pin4, OUTPUT);
pinMode(LED_pin5, OUTPUT);
pinMode(LED_pin6, OUTPUT);
pinMode(LED_pin7, OUTPUT);
}
void loop()
{
  if (sensorIR.decode(&codigoIR)) //permite almacenar el dato recibido en el espacio codigoIR
  {
    switch(codigoIR.value) { //compara el valor recibido con el valor de los casos siguientes
      //si coinciden ejecuta ese caso y sale del switch
      //se debe cambiar el número hexadecimal de cada tecla por el que habéis obtenido anteriormente
      case 0xFD28D7: Serial.println("Tecla: 4"); //0x indica hexadecimal
        digitalWrite(LED_pin4, !digitalRead(LED_pin4));
        break;
      case 0xFDA857: Serial.println("Tecla: 5");
        digitalWrite(LED_pin5, !digitalRead(LED_pin5));
        break;
      case 0xFD6897: Serial.println("Tecla: 6");
        digitalWrite(LED_pin6, !digitalRead(LED_pin6));
        break;
      case 0xFD18E7: Serial.println("Tecla: 7");
        digitalWrite(LED_pin7, !digitalRead(LED_pin7));
        break;
    }
    sensorIR.resume(); //listo para recibir el siguiente dato
    delay(200);
  } }

```

Ilustración 13 Práctica 6 del manual

Práctica (Placa):

Respecto a la placa que utilizamos para el proyecto usaremos una placa Arduino Mega 2560 tiene 54 pines de entrada/salida, de los que 14 pueden ser utilizados como salidas de PWM (Modulación por ancho de pulso), cuenta con 16 entradas analógicas. Tiene mayor memoria que la Arduino Uno y que la Leonardo, es por eso por lo que utilizaremos esta placa.

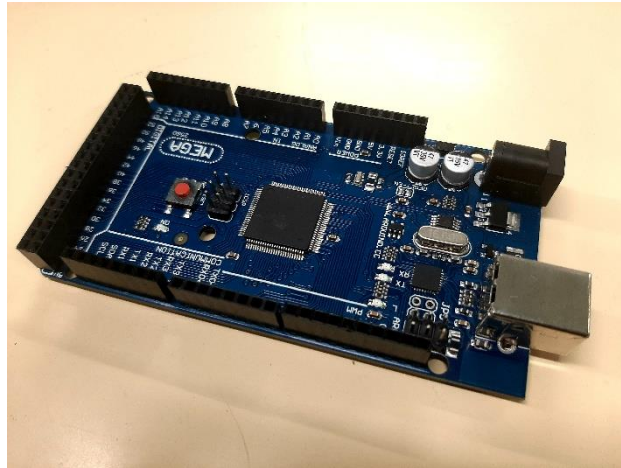


Ilustración 14 Arduino MEGA instalada

Ahora pasaremos a la explicación del siguiente punto de desarrollo del proyecto, el modelado 3D en SketchUp. Cada participante del grupo hizo su modelo de la casa domótica y se eligió el que se quería construir.

<https://youtu.be/6tptsBpuieI>





Ilustración 15 Diseño 3D en SketchUp



Ilustración 16 Vistas del diseño y perspectiva



Ilustración 17 Diseño 3D con cotas

Después pasamos a la simulación en TinkerCad dónde simulamos el circuito de la casa domótica con todas sus funciones. El poder simular el circuito y su programa es de gran ayuda ya que podemos detectar erros y añadir funciones con mayor facilidad que si lo hiciéramos directamente en la placa real

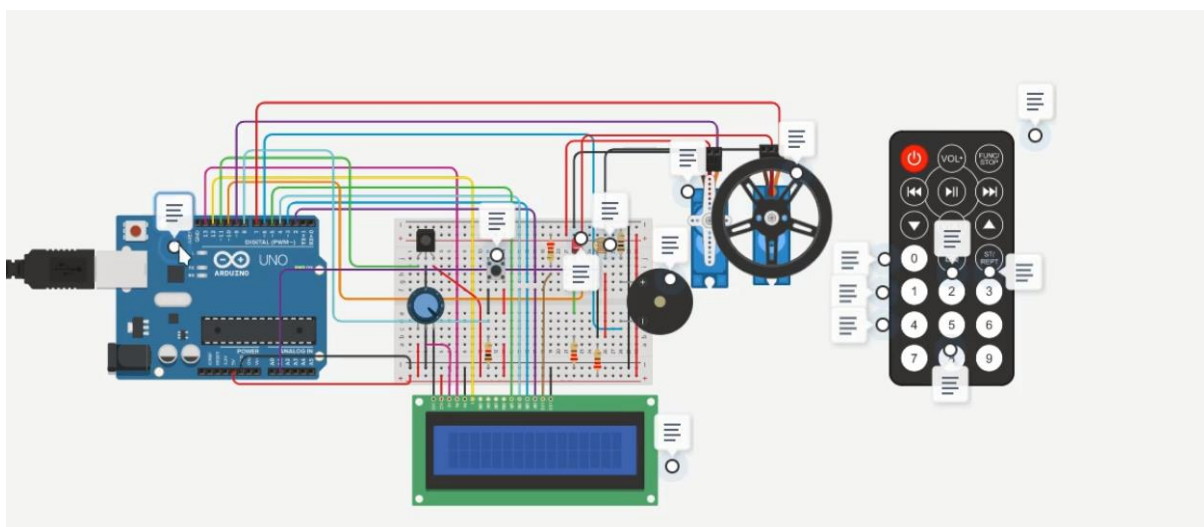


Ilustración 18 Simulación con Tinkercad

<https://youtu.be/xvklD3wc3ck>

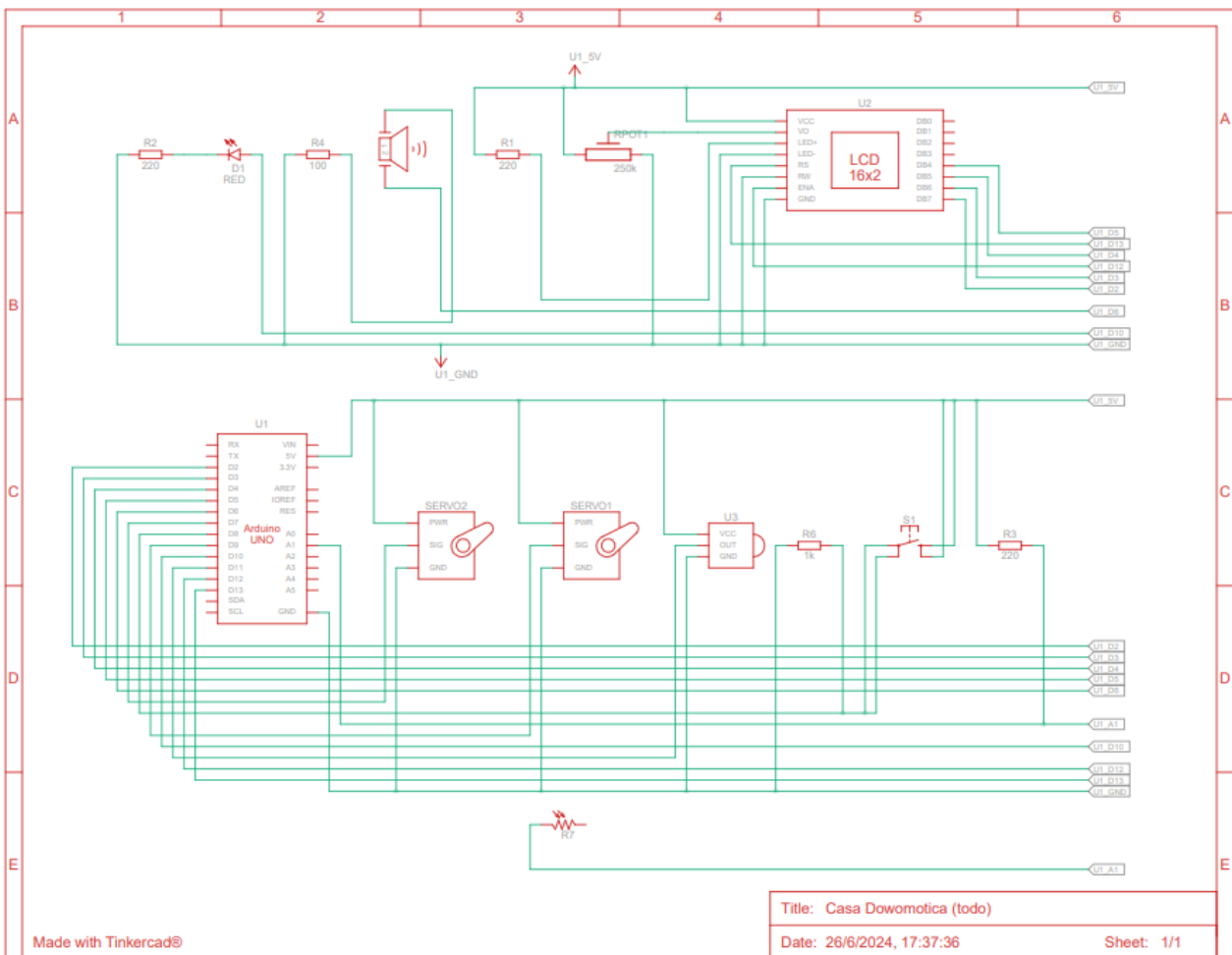
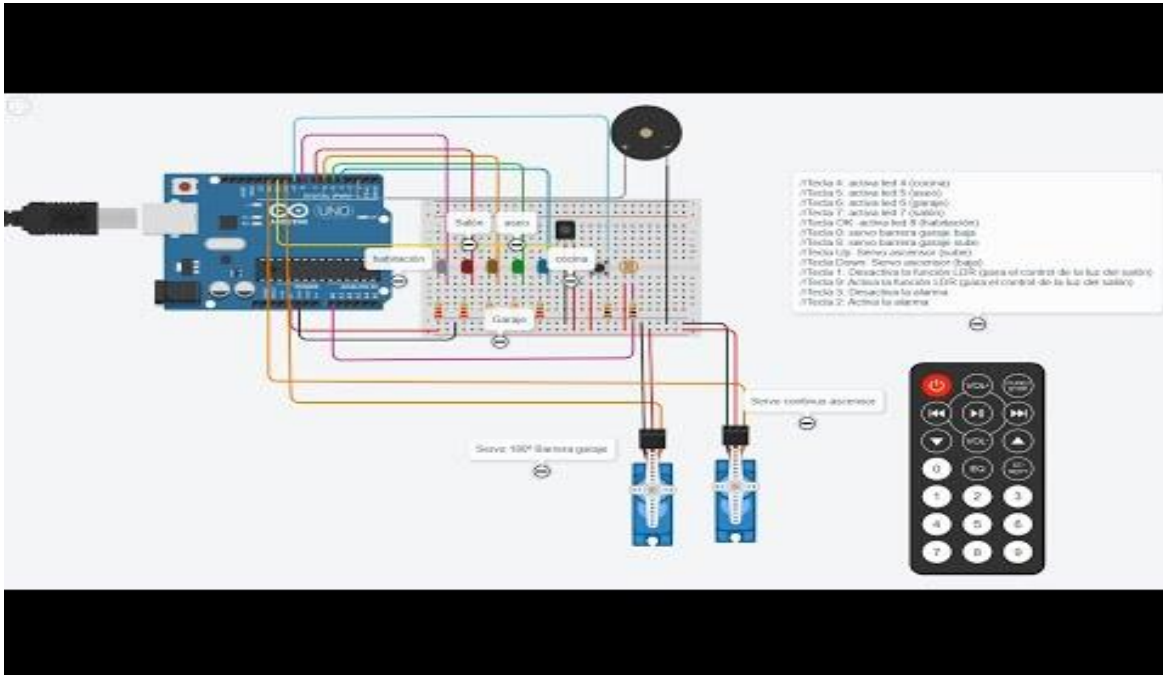


Ilustración 19 Esquema eléctrico

Una vez terminado y probado el código en TinkerCad pasamos a la construcción del proyecto, donde empezamos con la base de madera DM de 40 x 40 cm, en esta base es donde construiremos toda la estructura de la casa con madera contrachapada, y donde incorporaremos todos los componentes electrónicos, la placa Arduino Mega y la protoboard también se situarán en un extremo de la base.

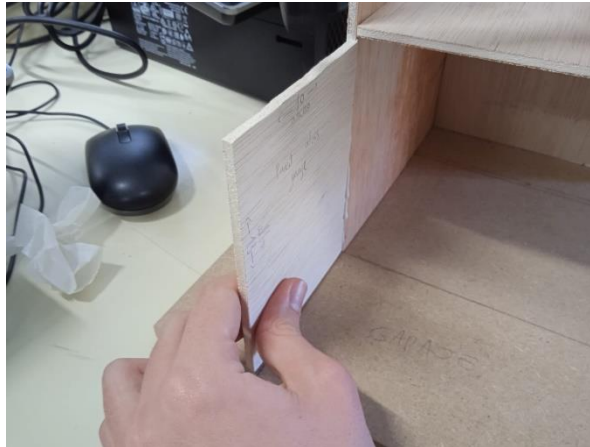


Ilustración 20 Proceso de montaje

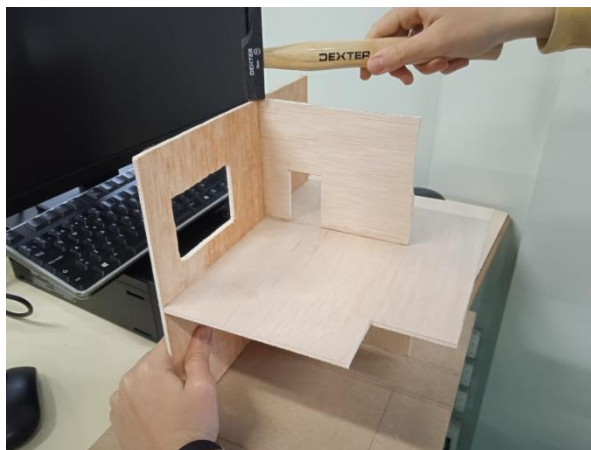


Ilustración 21 Montaje de la segunda planta

Finalmente, tenemos que probar el código realizado en Tinkercad en la placa real con Arduino IDE, y después de prueba y error al final dimos y solucionamos algunos inconvenientes que surgieron.

Código FINAL:

Aquí tenéis parte del programa final de la casa domótica. Si queréis construir y programar este proyecto tan ambicioso tendréis que poner os manos a la obra y trabajar mucho en él, tal y como han hecho nuestros alumnos/as:

```

void loop(){
  if (IR.decode()){

    if(value != 0){
      tecla();
      Serial.println(value, HEX);
      if (value == num0){ // Encender/apagar LDR
        LDR_LED = !LDR_LED;
        if(LDR_LED) lcdline("LDR on");
        else lcdline("LDR off");

      }else if(value == num1){ // Encender/apagar LED
        digitalWrite(pin_led1,!digitalRead(pin_led1));
        lcdline("LED 1");

      }else if(value == num2){
        digitalWrite(pin_led2,!digitalRead(pin_led2)); // Encender/apagar LED
        lcdline("LED 2");

      }else if(value == num3){
        digitalWrite(pin_led3,!digitalRead(pin_led3)); // Encender/apagar LED
        lcdline("LED 3");

      }else if(value == num4){
        digitalWrite(pin_led4,!digitalRead(pin_led4)); // Encender/apagar LED
        lcdline("LED 4");

      }else if(value == num5){
        digitalWrite(pin_led5,!digitalRead(pin_led5)); // Encender/apagar LED
        lcdline("LED 5");
      }else if(value == num6){ // Abrir/cerrar puerta

        if (password_puerta){ // Si la contrasena esta puesta
          lcdline("Puerta");
          delay(500);
          IR.resume();
          password();
        }
        if (!abierta){
          lcdline("Puerta abierta");
          servo1.write(0);
          abierta = 1;
        }else{
          lcdline("Puerta cerrada");
        }
      }
    }
  }
}

```

5) Resultados y análisis

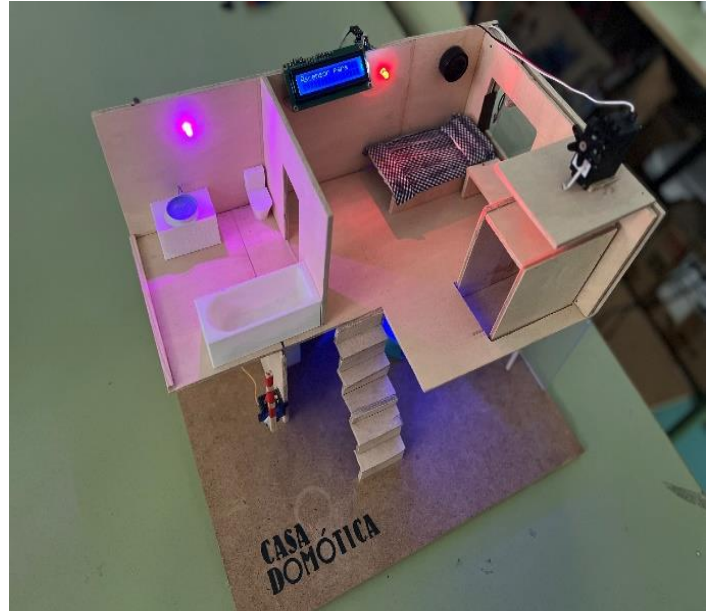


Ilustración 22 Producto final

En esta imagen se pueden ver todas las funciones de la casa activadas y funcionando correctamente.

Al principio usamos una placa Arduino nano, pero al añadir componentes como la LCD no teníamos suficiente espacio ni pines y cambiamos la placa por una Arduino Mega 2560, la cual cuenta con 54 pines digitales y 16 analógicos, además de por supuesto una mayor memoria.

Por supuesto, que también tuvimos problemas a la hora de programar, entre ellos los más destacados son problemas con el Buzzer para reproducir la música, y también con la LCD la cual daba ciertos errores.

El resultado de la resolución de los problemas surgidos tanto por parte del código, como por parte de del hardware o de la construcción, no es otra cosa que nuestro proyecto, la casa domótica, fruto de nuestro trabajo y errores.

6) Conclusiones

En conclusión, tras superar diferentes dificultades en el proceso de producción del proyecto, conseguimos de forma satisfactoria el resultado final deseado. Como hemos comentado anteriormente, al principio tuvimos problemas al montar el ascensor, ya que la cuerda se montaba sobre sí misma, provocando que éste se cayera. Para arreglar esto, implementamos unas guías que mantenían firme el ascensor. A esto hay que sumarle las dificultades para el funcionamiento correcto de la LCD y del buzzer.

En resumen, este proyecto CADI es una recopilación de todos los conocimientos de programación adquiridos durante el curso y los obtenidos a la hora de implementar este proyecto, junto con todas las herramientas utilizadas ya en cursos anteriores.

7) Agradecimientos

Queremos dar las gracias por la colaboración del departamento de tecnología y por supuesto a nuestra profesora Elena Gil, que nos ha proporcionado todas las herramientas necesarias para el desarrollo del proyecto, y a Carolina Caballero, cotutora del proyecto que nos ha apoyado a lo largo de todo el proceso.

También agradecer la colaboración de los alumnos de segundo de bachillerato que nos han ayudado a resolver varios problemas con nuestra casa domótica.

En resumen, estamos muy agradecidos con todos los involucrados en este proyecto ya sea de forma directa o bien indirecta, estamos seguros de que sin la colaboración de todos no habría sido posible finalizar el proyecto.

8) Bibliografía

- Cuaderno de programación y robótica proporcionado por el departamento de tecnología del instituto IES LA PATACONA.
- Como conectar una LCD: <https://programarfacil.com/blog/arduino-blog/texto-en-movimiento-en-un-lcd-con-arduino/>
- Como poner música en Arduino IDE: <https://juegosrobotica.es/musica-con-arduino/>
- <https://www.tinkercad.com/>
- <https://app.sketchup.com/>