

MAQUETA DE PRUEBAS PARA ARDUINO

MICRO-LOG[®]
LOGKIT
4223

1. OBJETIVOS

Construir una sencilla maqueta equipada con diferentes componentes electrónicos y realizar prácticas con Arduino UNO.

2. FOTOGRAFÍA



3. FUNCIONAMIENTO

La maqueta simula la entrada a una vivienda. Los diferentes componentes electrónicos están relacionados entre sí mediante Arduino y programados con Scratch. Además de realizar las prácticas que se proponen a continuación, el alumno podrá crear nuevas ideas.

4. LISTA DE MATERIALES

- | | |
|--|-----------------------------------|
| 1 Mini servo LOG 06 | 1 Diodo LED Rojo 5 mm LOG 724 |
| 1 Engrane módulo 1 LOG 107 | 1 LDR LOG 731 |
| 1 Tira dentada rígida módulo 1 LOG 199 | 1 Resistencia de 220 LOG 748 220 |
| 1 Contrachapado 12x24x0.3 cm LOG 300P | 5 Resistencias de 10K LOG 748 10K |
| 1 Contrachapado 24x30x0.3 cm LOG 300 | 1 Regleta macho LOG 815 |
| 1 Contrachapado 12x24x1 cm LOG 308 | 1 Metro de mazo de 8 cables S 564 |
| 2 Listones de madera 24x1x1 cm LOG 391 | 1 Shield de conexiones S 4031 |
| 2 Tornillos roscachapa LOG 461 | 4 Casquillos de plástico S 220P |
| 2 Pulsadores NA LOG 540 | 1 Zumbador activo LOG 7714 |
| 2 Finales de carrera LOG 549 | 1 Etiqueta con las salidas |
| 1 Diodo LED Verde 5 mm LOG 722 | 1 Hoja Técnica H4223 |

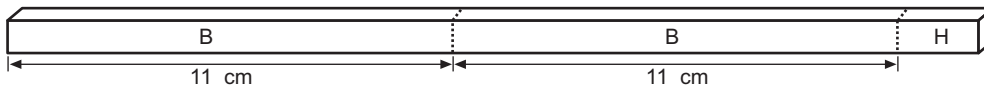
Leer todas las instrucciones y comprobar el listado de materiales antes de empezar el proyecto.

5. CONSTRUCCIÓN

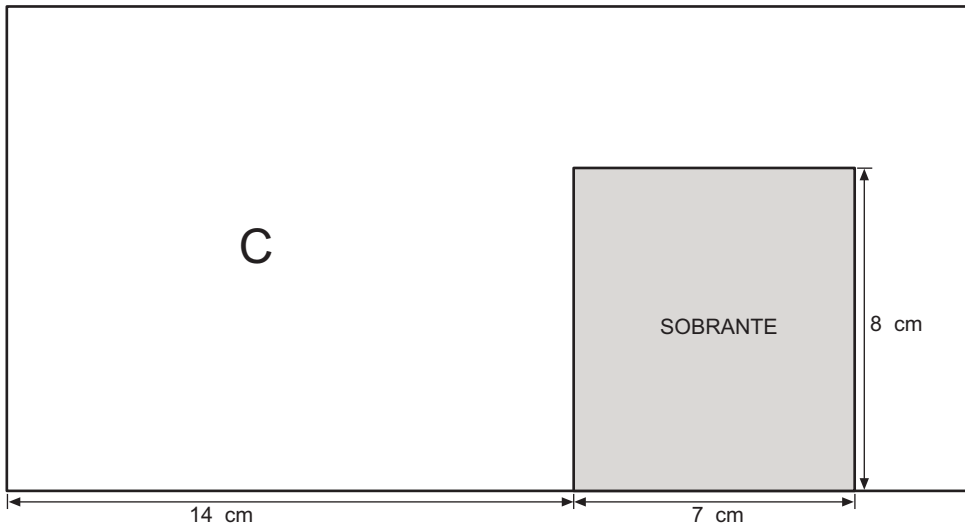
- Listón de 24x1x1 cm **LOG 391** que no requiere cortes.



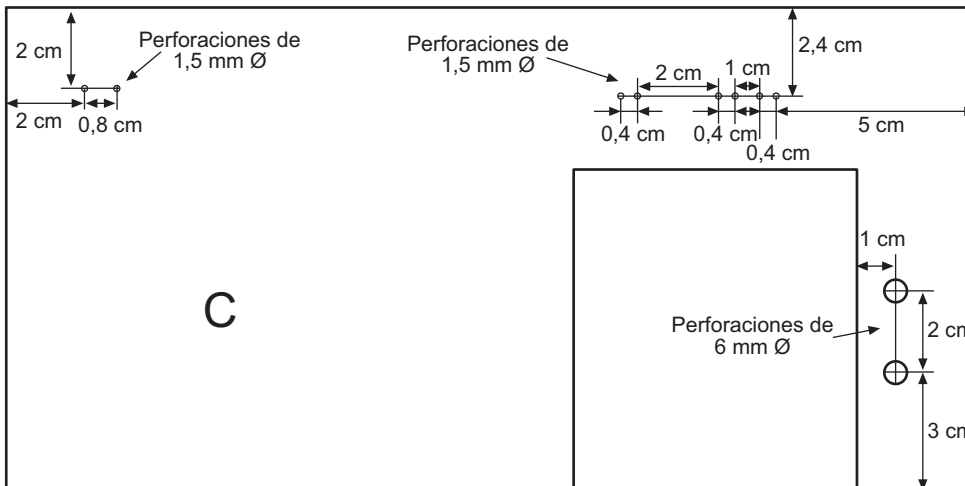
- Cortar un listón de 24x1x1 cm **LOG 391** obteniendo dos trozos de 11 cm.



- Realizar los siguientes cortes en el contrachapado de 24x12x0,3 cm **LOG 300P**.



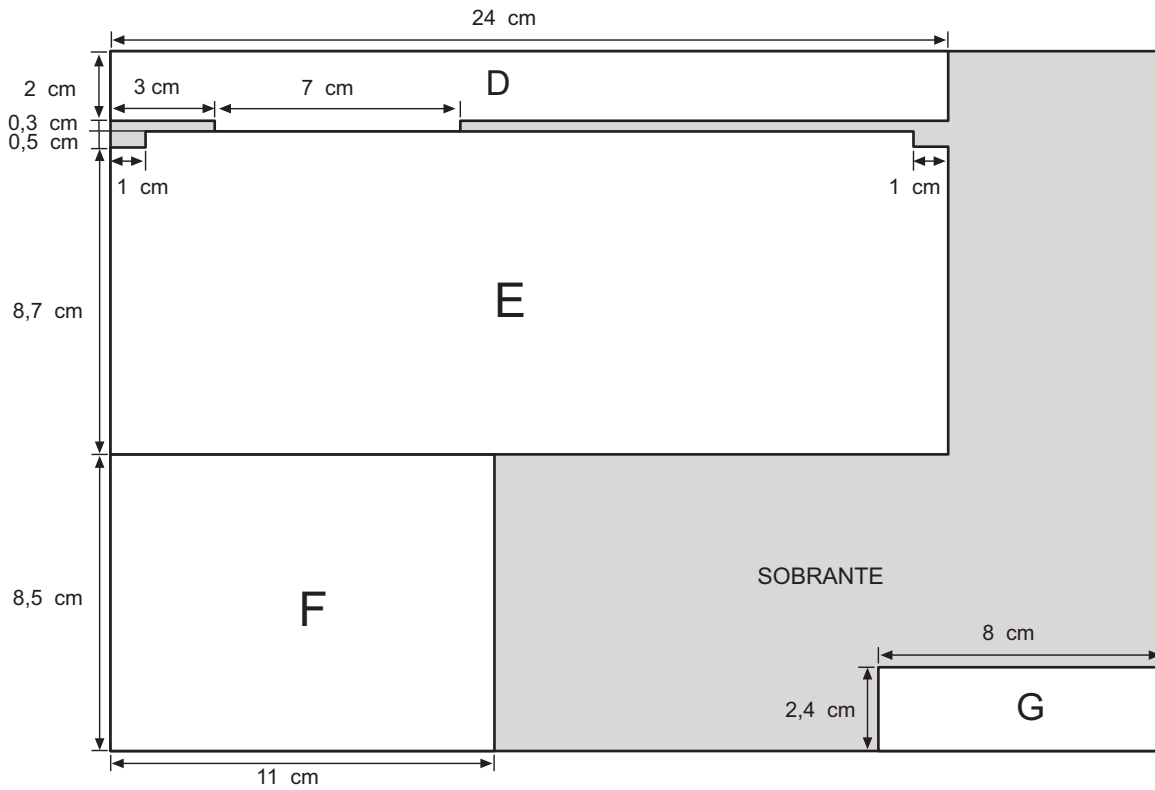
- Realizar las siguientes perforaciones en la pieza C con una barrena o con un taladro de columna.



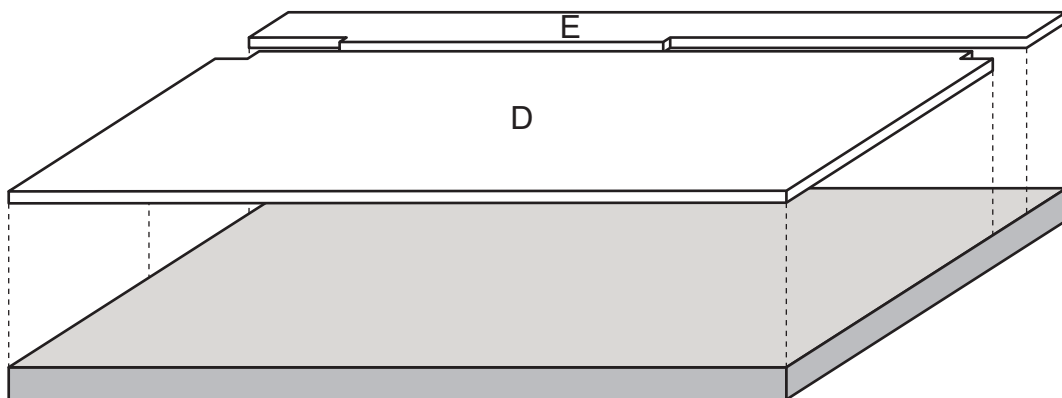
MAQUETA DE PRUEBAS PARA ARDUINO

MICRO-LOG[®]
LOGKIT
4223

- Realizar los siguientes cortes en el contrachapado de 30x20x0,3 cm **LOG 300**.

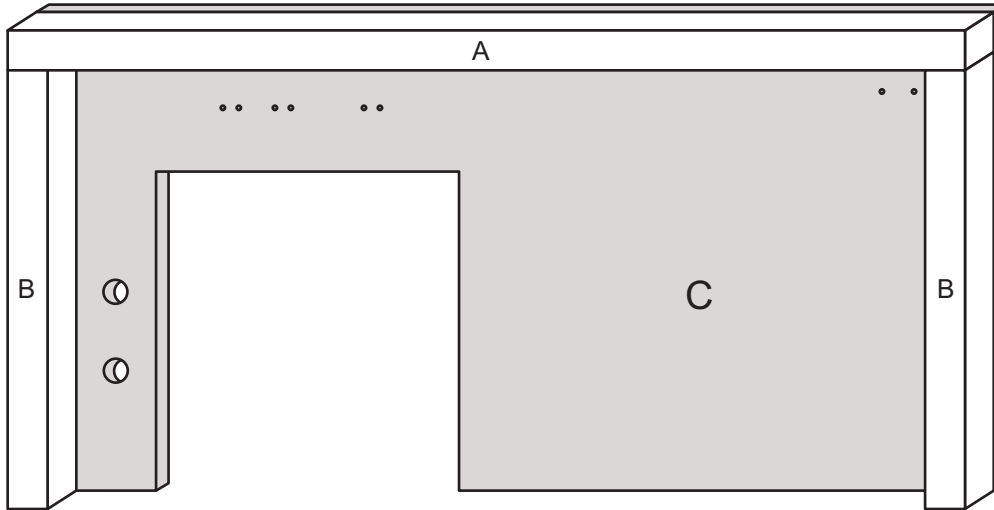


- Pegar las piezas "D" y "E" a la base de contrachapado de 24x12x1 cm **LOG 308**.

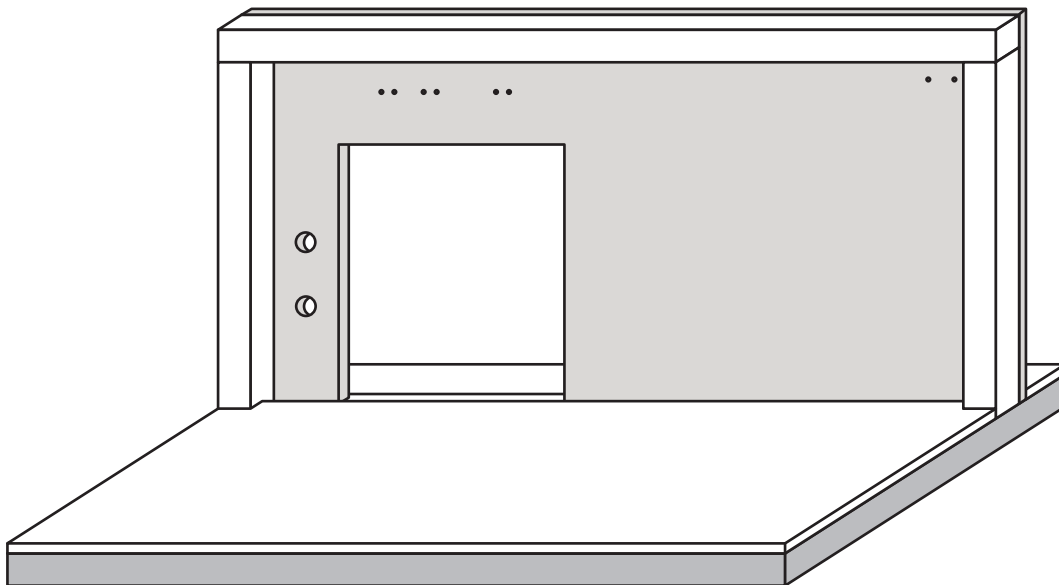


MAQUETA DE PRUEBAS PARA ARDUINO

- Pegar los listones "A" y "B" a la pieza "C". Utilizar cola de carpintero o pegamento termofusible.

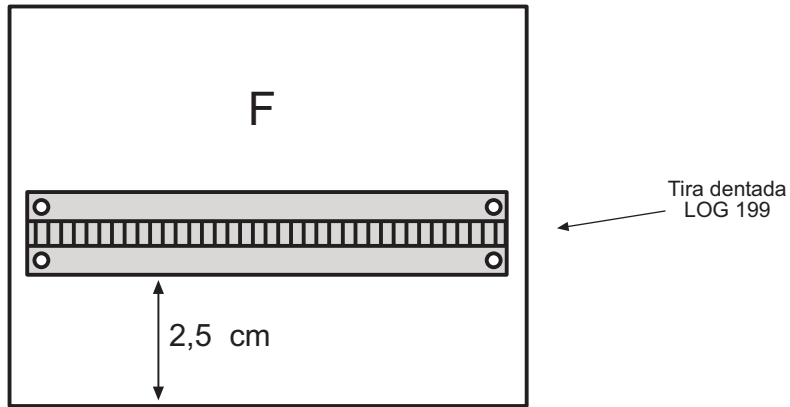


- Pegar los dos conjuntos



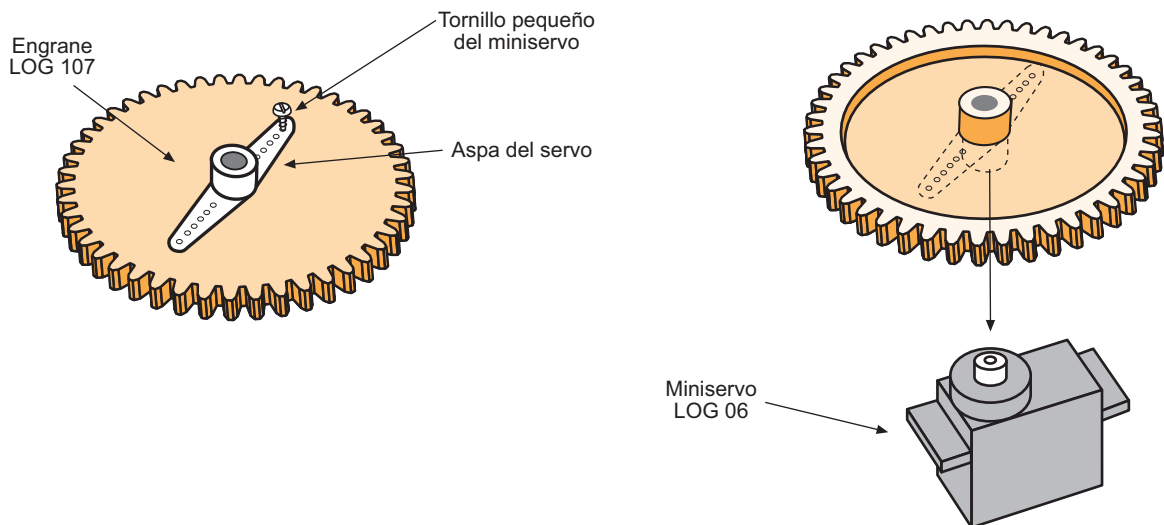
MAQUETA DE PRUEBAS PARA ARDUINO

- Pegar la tira dentada **LOG 199** en la pieza F utilizando pegamento termofusible.



- Pegar con pegamento termofusible el aspa del servo al engrane **LOG 107**. Si se dispone de mini taladro, realizar una perforación en el engrane, utilizando el último agujero del aspa, y atornillar el aspa al engrane con el tornillo pequeño que proporciona el pack del servo, para dar mayor sujeción al conjunto.

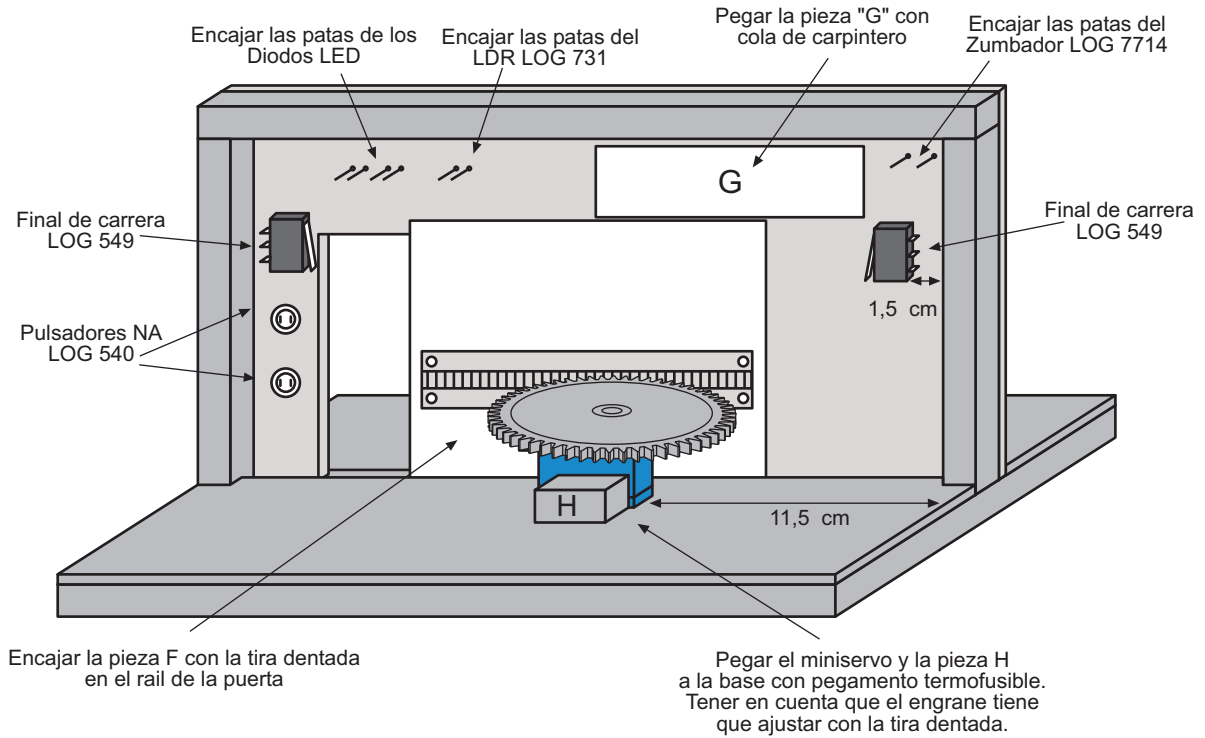
- Colocar el conjunto en el miniservo **LOG 06**.



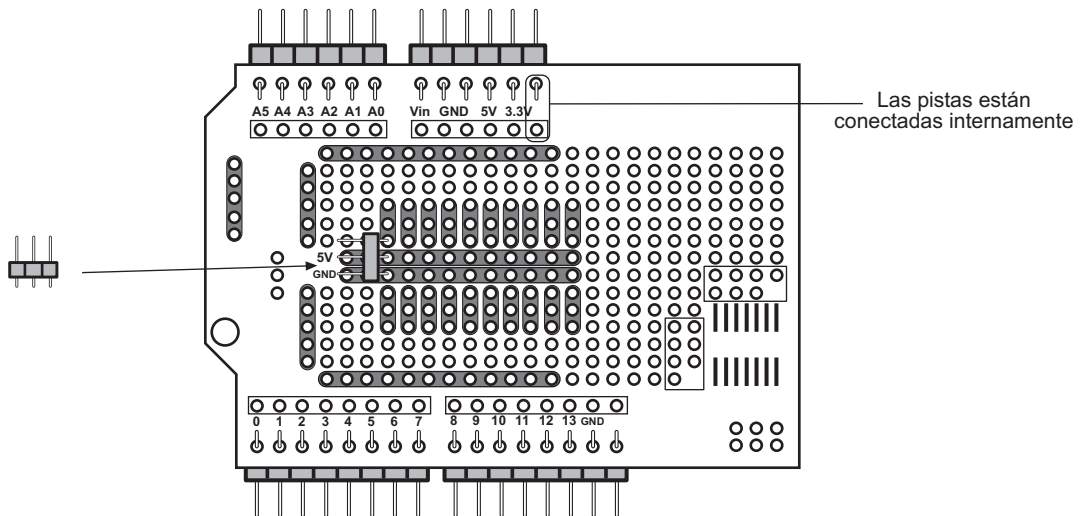
MAQUETA DE PRUEBAS PARA ARDUINO



- Pegar con pegamento termofusible los componentes eléctricos como se muestra en el dibujo y atornillar los pulsadores NA LOG 540.



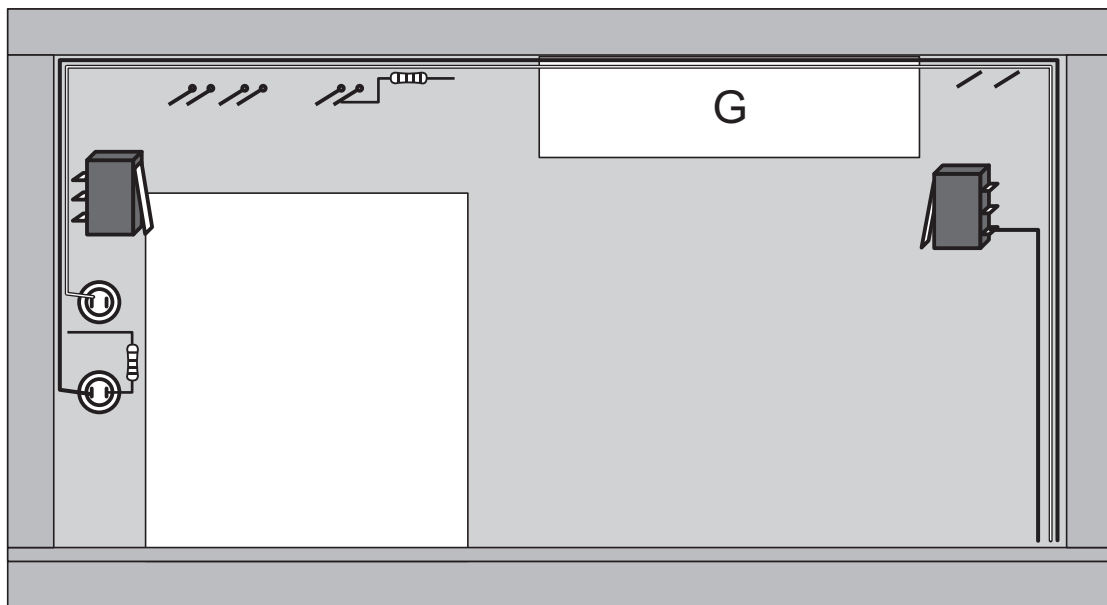
- Recortar la regleta de conexiones LOG 815 en 2 trozos de 6 pines, 2 trozos de 8 pines y 1 trozo de 3 pines, y soldarlos en el shield de conexiones, por la cara que no está serigrafiada.



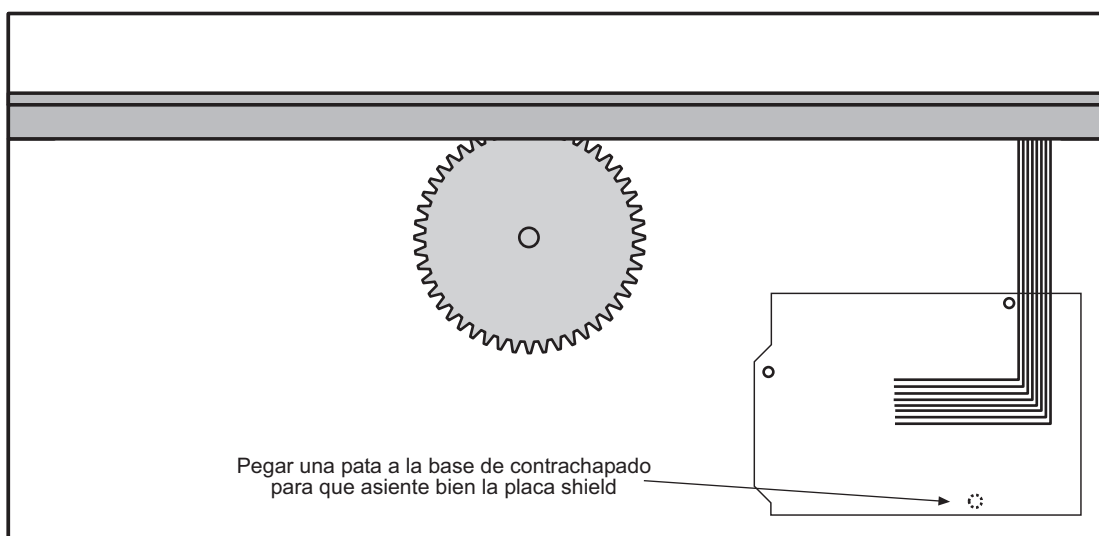
MAQUETA DE PRUEBAS PARA ARDUINO

MICRO-LOG[®]
**LOGKIT
4223**

- Soldar las conexiones, siguiendo el esquema de los puntos 6 y 7. Para mejorar la estética y que no queden los cables sueltos, según vamos realizando las conexiones, pegar los cables al marco de la puerta con pegamento termofusible o cianoacrilato.
- Las resistencias, soldarlas directamente en el componente eléctrico.
- Unificar las conexiones de los negativos y positivos, para que sea más fácil y no tener demasiados cables.



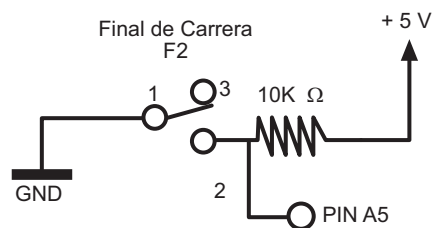
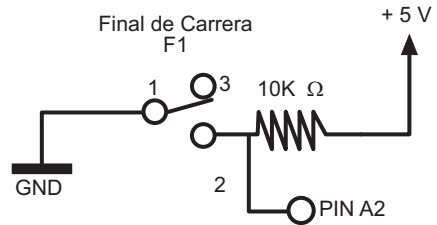
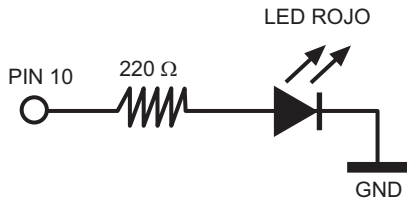
- Dejar un sobrante de unos 10 cm, desde la pared de la puerta, para poder realizar las conexiones en el shield.



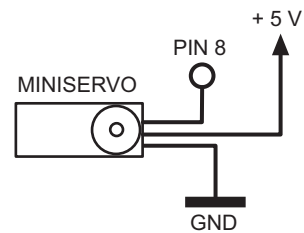
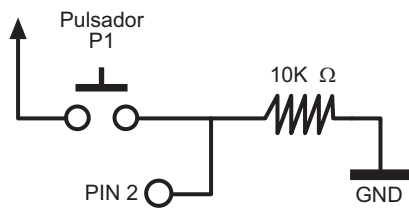
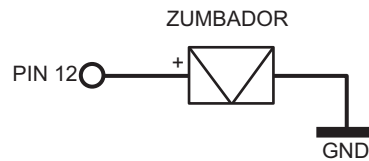
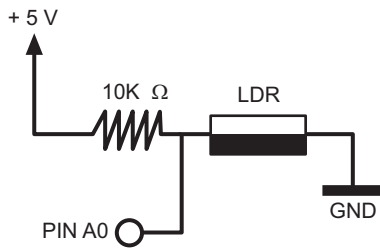
- Cuando estén todas las conexiones soldadas (puntos 6 y 7), atornillar la placa shield a la base. Utilizar los tornillos LOG 461 y colocar los casquillos S 220P para separar un poco la placa de la base de contrachapado.

MAQUETA DE PRUEBAS PARA ARDUINO

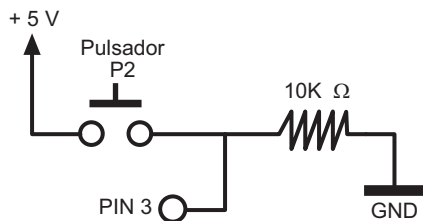
6. ESQUEMA ELÉCTRICO



NOTA: El LED verde no lleva resistencia porque está integrada en la tarjeta Arduino en la salida del pin 13.



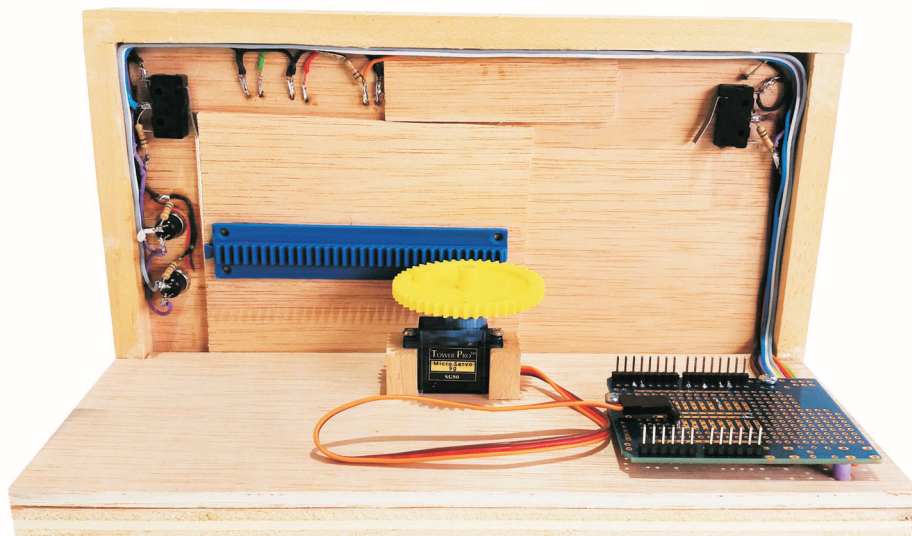
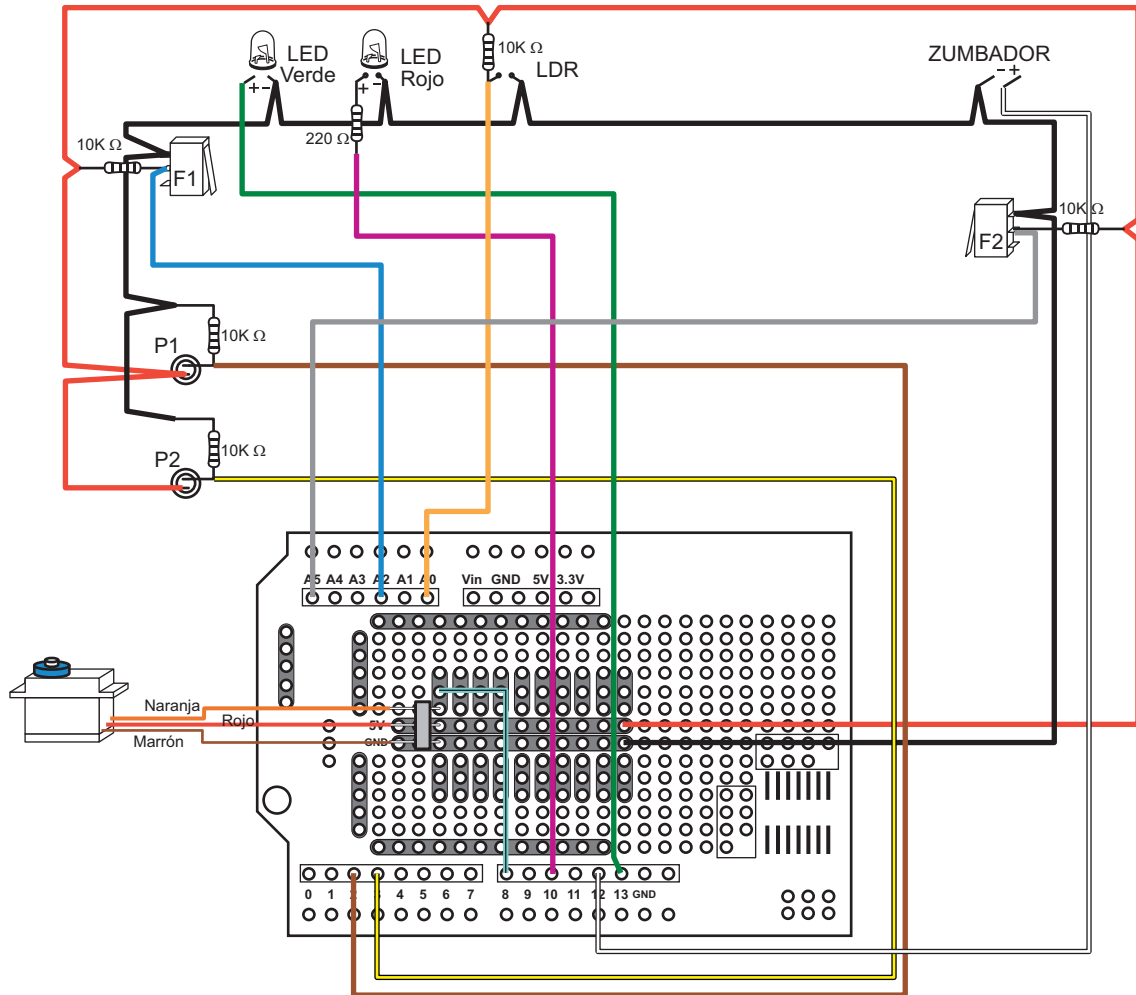
Naranja / Amarillo: Datos Pin 8
Rojo: Positivo Pin 5V
Negro / Marrón: Negativo Pin GND



MAQUETA DE PRUEBAS PARA ARDUINO

7. CONEXIONES

- Soldar los cables a la placa shield por la cara serigráfica.



8. PROGRAMACIÓN EN S4A

LED:

Un LED es una luz que podemos encender y apagar desde la tarjeta controladora. A la hora de conectarlo a la tarjeta arduino, debemos respetar su polaridad. Conectamos la pata negativa del LED a GND y la pata positiva al pin 13 de la tarjeta controladora.

Para encender el LED verde utilizamos la instrucción “digital 13 encendido”.
Para apagar el LED verde utilizamos la instrucción “digital 13 apagado”.

Ejemplo: programamos una luz intermitente.



Para el LED rojo, conectamos la pata positiva al pin 10 y utilizamos la instrucción “digital 10 encendido” o “digital 10 apagado”.

Zumbador:

Es un operador capaz de emitir un sonido. Para conectarlo a la tarjeta arduino debemos respetar su polaridad. La pata negativa la conectamos a GND y la pata positiva se conecta al pin 12 de la tarjeta arduino.

Para que el zumbador emita un sonido utilizamos la instrucción “digital 12 encendido”.
Para apagar el zumbador utilizamos la instrucción “digital 12 apagado”.

Ejemplo: programamos un sonido intermitente.



MAQUETA DE PRUEBAS PARA ARDUINO

LDR:

Operador analógico que devuelve un valor de 0 a 1023 dependiendo de la cantidad de luz que detecte.

Se conecta a la entrada analógica 0.

Ejemplo: El siguiente programa enciende el LED si la LDR determina falta de luz.

```

al presionar
por siempre
si valor del sensor Analog5 > 500
digital 13 encendido
si no
digital 13 apagado
    
```

Pulsadores:

Dispositivo digital que nos devolverá dos estados: "true" si se está presionando y "false" si no lo está.

Tenemos dos pulsadores que se conectan a los pines digitales 2 y 3.

Para programarlos, utilizaremos la instrucción “¿sensor digital2 presionado?” que nos devolverá el valor "true" o "false" en función del estado del pulsador.

Ejemplo: Si presionamos el pulsador, se enciende el LED.

```

al presionar
por siempre
si ¿sensor Digital2 presionado?
digital 13 encendido
si no
digital 13 apagado
    
```

MAQUETA DE PRUEBAS PARA ARDUINO

Ejemplo: Si presionamos un pulsador se enciende el LED, si presionamos el otro pulsador se apaga el LED.

```

al presionar
  por siempre
    si ¿sensor Digital2 presionado?
      digital 13 encendido
    si ¿sensor Digital3 presionado?
      digital 13 apagado
  
```

Miniservo:

El miniservo es el encargado de llevar a cabo el movimiento de la puerta corredera. Sus cables se conectan a la tarjeta arduino del siguiente modo:

- Rojo a 5V
- Marrón a GND
- Naranja al pin 8

Para programarlo utilizamos las siguientes instrucciones.

“motor 8 apagado” – para el motor.

“motor 8 dirección horario/antihorario” – mueve el motor en uno u otro sentido.

Ejemplo: movemos la puerta en ambos sentidos.

```

al presionar
  por siempre
    motor 8 dirección horario
    esperar 1 segundos
    motor 8 dirección antihorario
    esperar 1 segundos
  
```

Finales de carrera:

Funcionan igual que los pulsadores. Debido a la distribución de entradas y salidas de S4A, en lugar de conectar los finales de carrera a pines digitales, los conectaremos a dos pines analógicos. Cuando la entrada devuelva un valor de 1023, significa que no ha sido accionado. Cuando devuelva un valor inferior a 100, nos estará indicando que el final de carrera se ha pulsado.

Utilizamos estos dispositivos para conocer el inicio y el final del recorrido de la puerta corredera.

Conectamos los finales de carrera a los pines A2 y A5.

Para programarlo utilizamos la instrucción “valor del sensor analog0” que nos devolverá un número de 0 a 1023 que estará leyendo la entrada de este operador.

Ejemplo: El pulsador 1 abre la puerta, y el pulsador 2 cierra la puerta. Los finales de carrera paran el motor cuando la puerta esté completamente abierta / cerrada.



MAQUETA DE PRUEBAS PARA ARDUINO

Ejemplo: ampliamos el programa anterior, el zumbador se activará cuando se presione un pulsador zumbador como aviso de movimiento, mientras que dos LEDs señalarán si la puerta se está abriendo o cerrando.



Si queremos ampliar este ejercicio, añadiendo la funcionalidad de encender la luz cuando se hace de noche, basta con tener en el escritorio del programa el ejercicio que acabamos de hacer junto con el de la LDR ya realizado. Así ambos se ejecutarán a la vez de manera paralela.

MAQUETA DE PRUEBAS PARA ARDUINO**9. PROGRAMACIÓN EN ARDUINO IDE****LED:**

```
void setup() {  
    pinMode(13, OUTPUT);  
}  
  
void loop() {  
  
    digitalWrite(13, HIGH);  
    delay(1000);  
    digitalWrite(13, LOW);  
    delay(1000);  
}
```

ZUMBADOR:

```
void setup() {  
    pinMode(12, OUTPUT);  
}  
  
void loop() {  
  
    digitalWrite(12, HIGH);  
    delay(1000);  
    digitalWrite(12, LOW);  
    delay(1000);  
}
```

LDR:

```
void setup() {  
    pinMode(13, OUTPUT);  
}  
  
void loop() {  
  
    if(analogRead(A5)>500){  
        digitalWrite(13, HIGH);  
    }  
    else{  
        digitalWrite(13, LOW);  
    }  
    delay(100);  
}
```

MAQUETA DE PRUEBAS PARA ARDUINO**PULSADORES:**

```
void setup() {
  pinMode(13, OUTPUT);
  pinMode(2, INPUT);
  pinMode(3, INPUT);
}

void loop() {
  if(digitalRead(2)){
    digitalWrite(13, HIGH);
  }
  else{
    digitalWrite(13,LOW);
  }
  delay(100);
}

void setup() {
  pinMode(13, OUTPUT);
  pinMode(2, INPUT);
  pinMode(3, INPUT);
}

void loop() {
  if(digitalRead(2)){
    digitalWrite(13, HIGH);
  }
  if(digitalRead(2)){
    digitalWrite(13,LOW);
  }
  delay(100);
}
```

MINISERVO:

```
#include <Servo.h>
Servo myservo;

void setup() {
  pinMode(12, OUTPUT);
  myservo.attach(8);
}

void loop() {
  myservo.write(0);
  delay(1000);
  myservo.write(180);
  delay(1000);
}
```


FINALES DE CARRERA:

```
#include <Servo.h>
Servo myservo;

void setup() {
  pinMode(2, INPUT);
  pinMode(3, INPUT);
  myservo.attach(8);
  pos==0;
}

void loop() {

  if(digitalRead(2)){
    do{
      myservo.write(pos);
      pos==pos+1;
    } while ((analogRead(A5)>100)|| (pos==180))
  }

  if(digitalRead(3)){
    do{
      myservo.write(pos);
      pos==pos-1;
    } while ((analogRead(A2)>100)|| (pos==0))
  }
}
```

PUERTA COMPLETA:

```
#include <Servo.h>
Servo myservo;
int i=0;
void setup() {
  pinMode(12, OUTPUT);
  pinMode(13, OUTPUT);
  pinMode(2, INPUT);
  pinMode(3, INPUT);
  myservo.attach(8);
  myservo.write(i);
}

void loop() {
  if(digitalRead(2)){
    digitalWrite(12, HIGH);
    delay(20);
    digitalWrite(12, LOW);
    do{
      i=i+1;
      digitalWrite(13, HIGH);
      myservo.write(i);
    }while((analogRead(A5)>100)&&(i<180))
    digitalWrite(13,HIGH);
  }
  if(digitalRead(3)){
    digitalWrite(12, HIGH);
    delay(20);
    digitalWrite(12, LOW);
    do{
      i=i-1;
      digitalWrite(10, HIGH);
      myservo.write(i);
    }while((analogRead(A2)>100)&&(i>0))
    digitalWrite(10,LOW);
  }
}
```

MAQUETA DE PRUEBAS PARA ARDUINO



10. DETALLES DE TIPO PRÁCTICO

- Cargar el firmware en el Arduino para que funcione con S4A.
- Tiempo de construcción: 4 H.
- Nivel de dificultad: Medio

DIODO LED

ZUMBADOR

FINAL DE CARRERA

RESISTENCIAS

220 Ω: Rojo - Rojo - Marrón
10K Ω: Marrón - Negro - Naranja

SERVOMOTOR

- Cable naranja o amarillo: salida 8
- Cable rojo: 5V
- Cable marrón o negro: GND

11. HERRAMIENTA ACONSEJADAS

- Segueta
- Regla y lápiz
- Destornillador plano
- Pelacables
- Alicates
- Pistola de pegamento termofusible

- Barrena o minitaladro
- Cola de carpintero
- Destornillador estrella
- Soldador y estaño
- Pinzas

12. PRUEBAS

- Al conectar Arduino al PC y encender S4A debe aparecer el mensaje "Buscando placa". Si no reinstalar el firmware en la tarjeta.
- Si los programas no funcionan, comprobar que las conexiones de cada componente son correctas.

Otras prácticas a desarrollar por el alumno:

1. Utilizar los finales de carrera como pulsadores que enciendan y apaguen los LEDs.
2. Abrir y cerrar la puerta en función del estado de los LEDs, utilizando los LEDs a modo de semáforo.
3. Utilizando la instrucción "motor 8 ángulo 180° mover la puerta en ambos sentidos.
4. Utilizando la instrucción "motor 8 ángulo 180° mover la puerta 5 grados en uno u otro sentido con cada presión sobre los pulsadores.
5. Cerrar la puerta cuando LED detecte poca luz y mantenerla abierta cuando haya luz.
6. Con un pulsador, abrir la puerta y encender el LED verde, esperar 5 segundos y a continuación encender el LED rojo y cerrar la puerta.