

Celsius

Fahrenheit

80

176

70

158

60

140

50

122

40

104

30

86

20

68

10

50

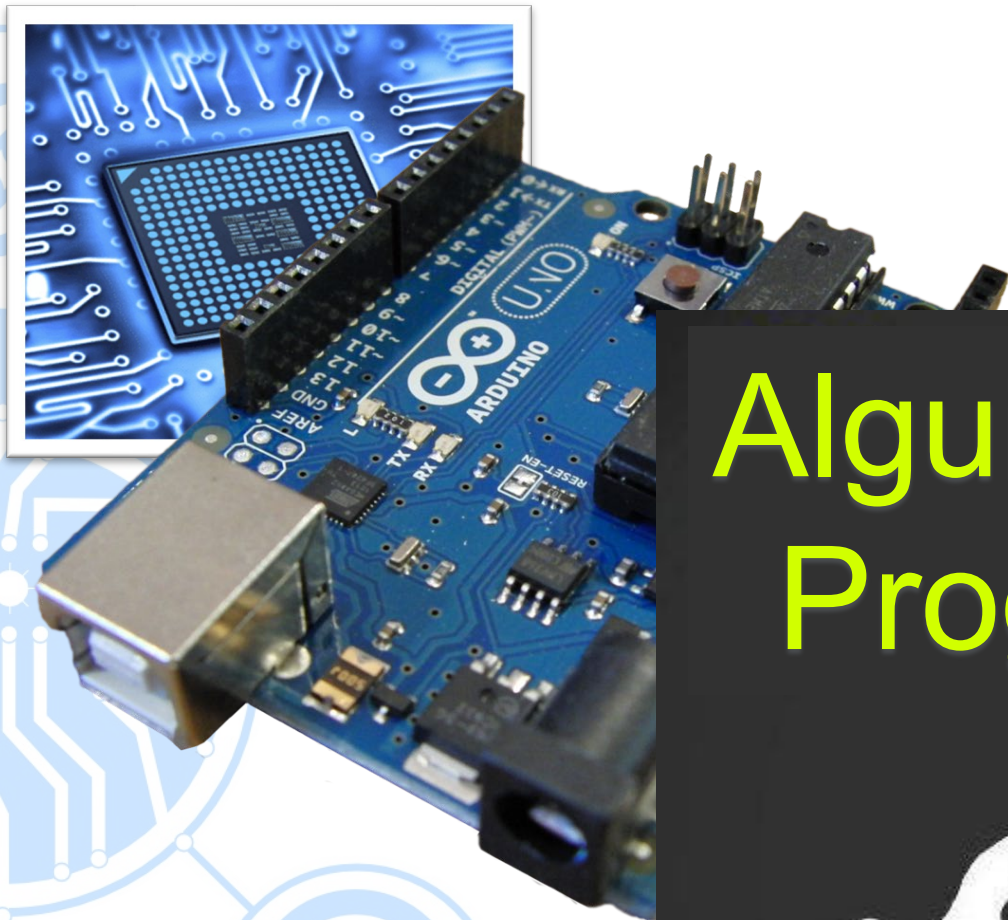
**Controlar:**  
**-Temperatura**  
**-Luz**  
**-Dar avisos:**  
**Sonido y LEDs**

Esta foto de Autor desconocido está bajo licencia [CC BY](https://creativecommons.org/licenses/by/4.0/)

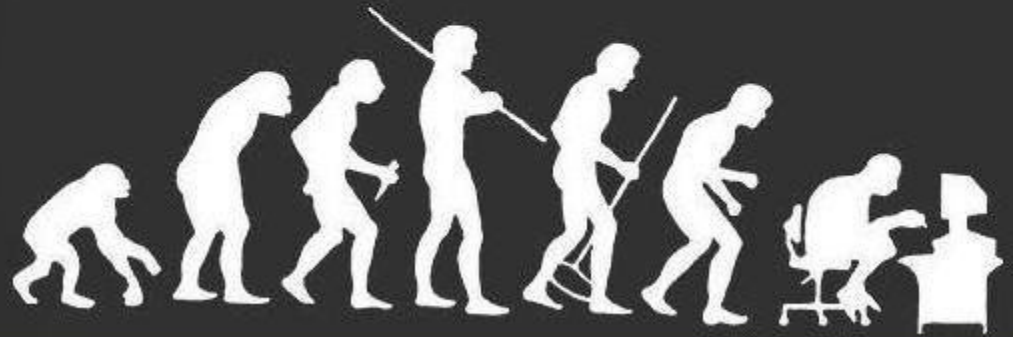


<http://smartopenlab.com>  
[@SolEpcc](https://twitter.com/SolEpcc)





# Alguien tiene que Programarlos!!!



Esta foto de Autor desconocido está bajo licencia [CC BY-SA](#)

Esta foto de Autor desconocido está bajo licencia [CC BY-SA](#)



Smart  
Open  
Lab

<http://smartopenlab.com>  
[@SolEpcc](#)

# Representación de datos

INTRO

- Los computadores trabajan y transmiten bits
- Cada bit tiene 2 estados:
  - ENCENDIDO / APAGADO
  - VERDADERO / FALSO
  - 1 / 0
  - HIGH / LOW
- Arduino puede funcionar siendo:
  - HIGH=1 y LOW=0 o al revés



# Arduino IDE

The image shows the Arduino IDE 2.2.2 interface with several callouts explaining the functions of the icons and buttons:

- Compilar (verificar Código)**: Points to the checkmark icon in the top toolbar.
- Cambiar placa seleccionada**: Points to the dropdown menu showing "Arduino Uno".
- Compilar y subir a la placa**: Points to the upload icon (a right-pointing arrow) in the top toolbar.
- Abrir monitor serie (para ver mensajes)**: Points to the serial monitor icon (a terminal window) in the top right corner.
- Placa seleccionada y ¿conectada?**: Points to the status bar at the bottom which says "Arduino Uno on COM5 [not connected]".

The main code editor shows the following C++ code:

```
1 // sketch_mar10a.ino
2
3 #include <Arduino.h>
4
5 // Definición de pines de salida
6 const int LED1 = 13;
7 const int LED2 = 12;
8
9 // Variables globales
10 int temperatura = 20;
11
12 void setup() {
13   pinMode(LED1, OUTPUT);
14   pinMode(LED2, OUTPUT);
15   Serial.begin(9600);
16 }
17
18 void loop() {
19   if (temperatura < 30) {
20     digitalWrite(LED1, LOW);
21     digitalWrite(LED2, LOW);
22     Serial.println(" bajo");
23   } else if (temperatura < 35) {
24     digitalWrite(LED1, HIGH);
25     digitalWrite(LED2, LOW);
26     Serial.println(" medio");
27   }
28   else {
29     digitalWrite(LED1, HIGH);
30     digitalWrite(LED2, HIGH);
31     Serial.println(" alto");
32   }
33   delay(1000);
34 }
35
36
37
38
39
40
41
42
43
44
45
46
```

The Serial Monitor at the bottom shows the output:

```
25.11 bajo
44
25.39 bajo
42
25.39 bajo
40
```



# Zonas del programa

1er programa: comunicación (serie) con Arduino-PC

```
int valorSensor=20;
```



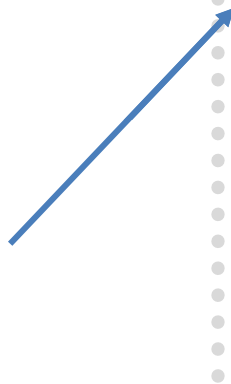
Global Variables

```
void setup() {  
  Serial.begin(9600);  
  Serial.println ("Bienvenidx!");  
}
```



setup()

```
void loop() {  
  Serial.print("El valor es " );  
  Serial.println(valorSensor);  
}
```



loop()

- Variable (caja/almacén)
- Una sola vez
- Continuamente

# Representación de datos

## TIPOS de DATOS

- ¿Y dónde se almacenan estos datos? ¡variables!
  - boolean → *true* o *false*, 0 ó 1 (1 bit → 1 byte)
  - byte, char, unsigned char → *del 0 al 255*
  - int, unsigned int → *del 0 al 65535*
  - long, unsigned long, double, float → 32 bits



8 bits

char



16 bits

int, unsigned int



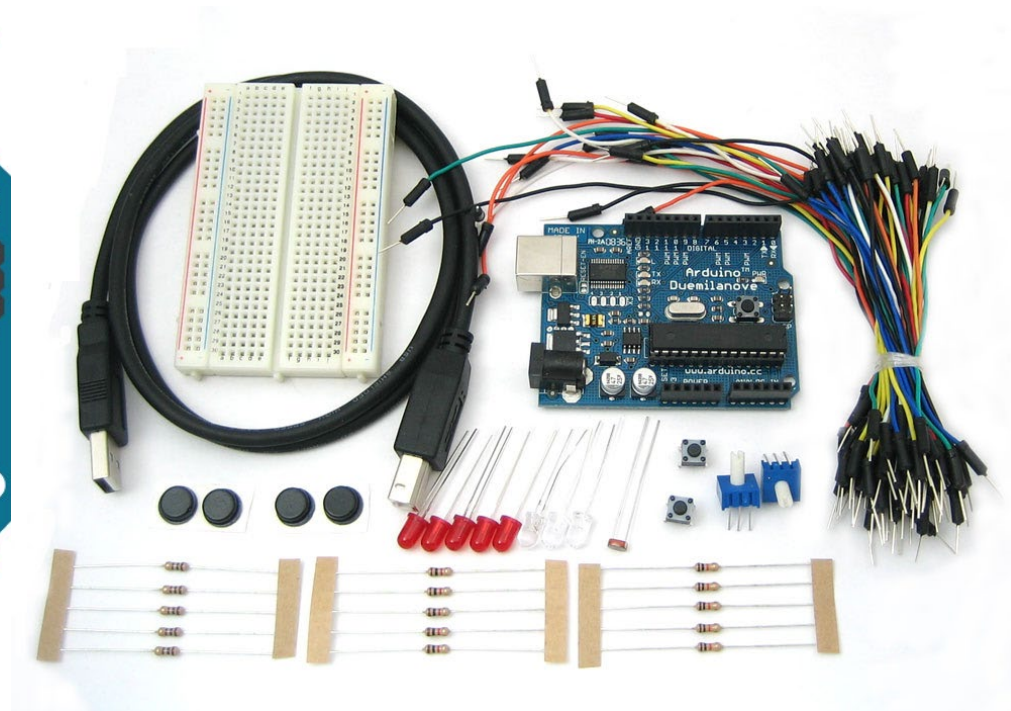
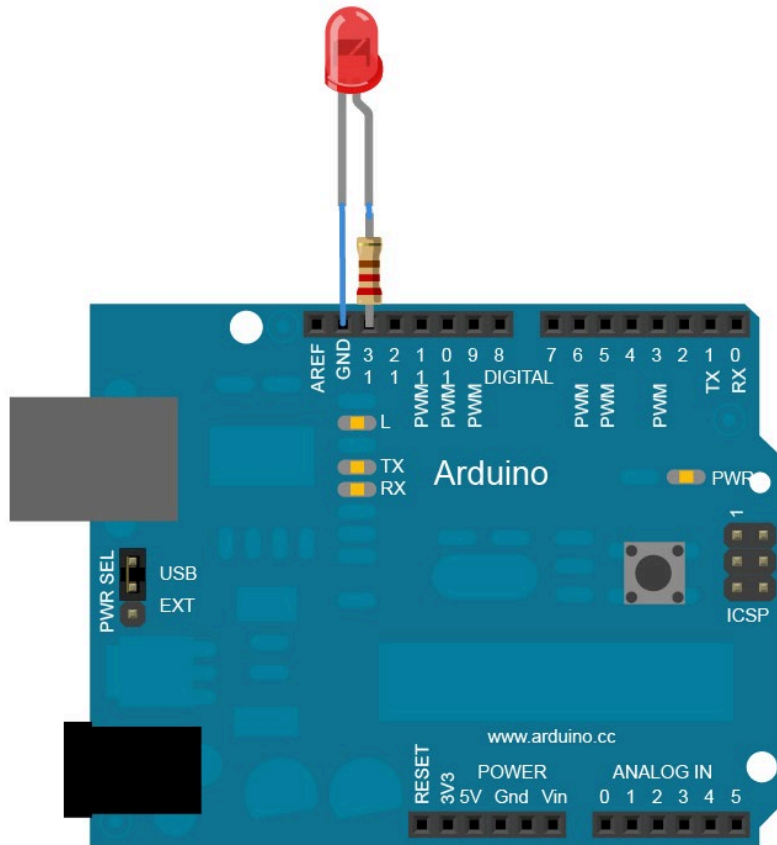
32 bits

long, unsigned long  
float



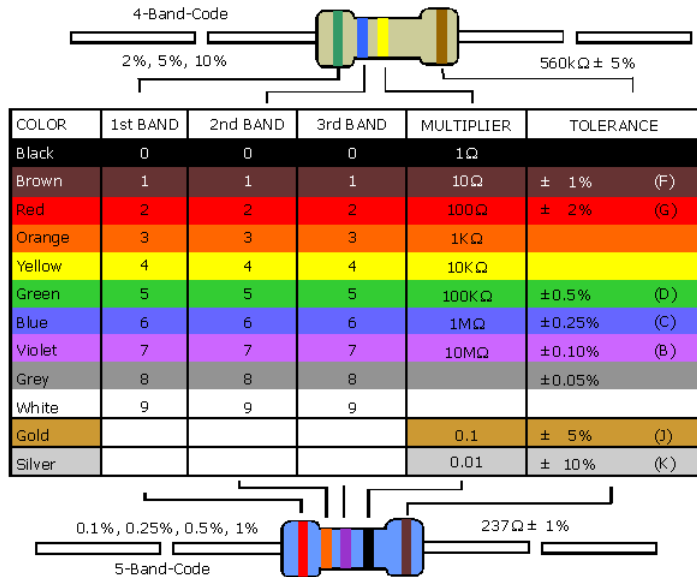
# Formas de prototipado

Directa



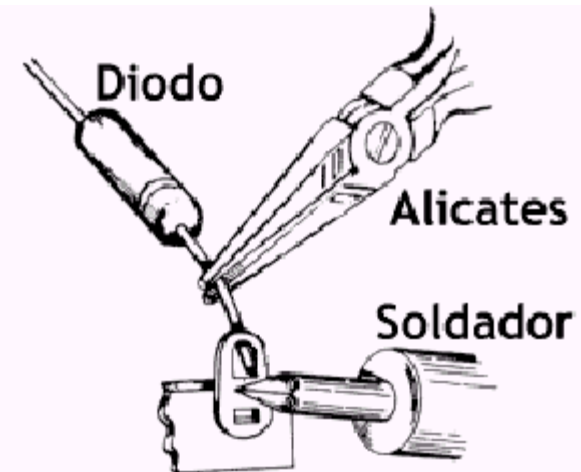
# Formas de prototipado

Problemas: Requiere ciertos conocimientos y habilidades



Electronix Express / RSR  
<http://www.elexp.com>

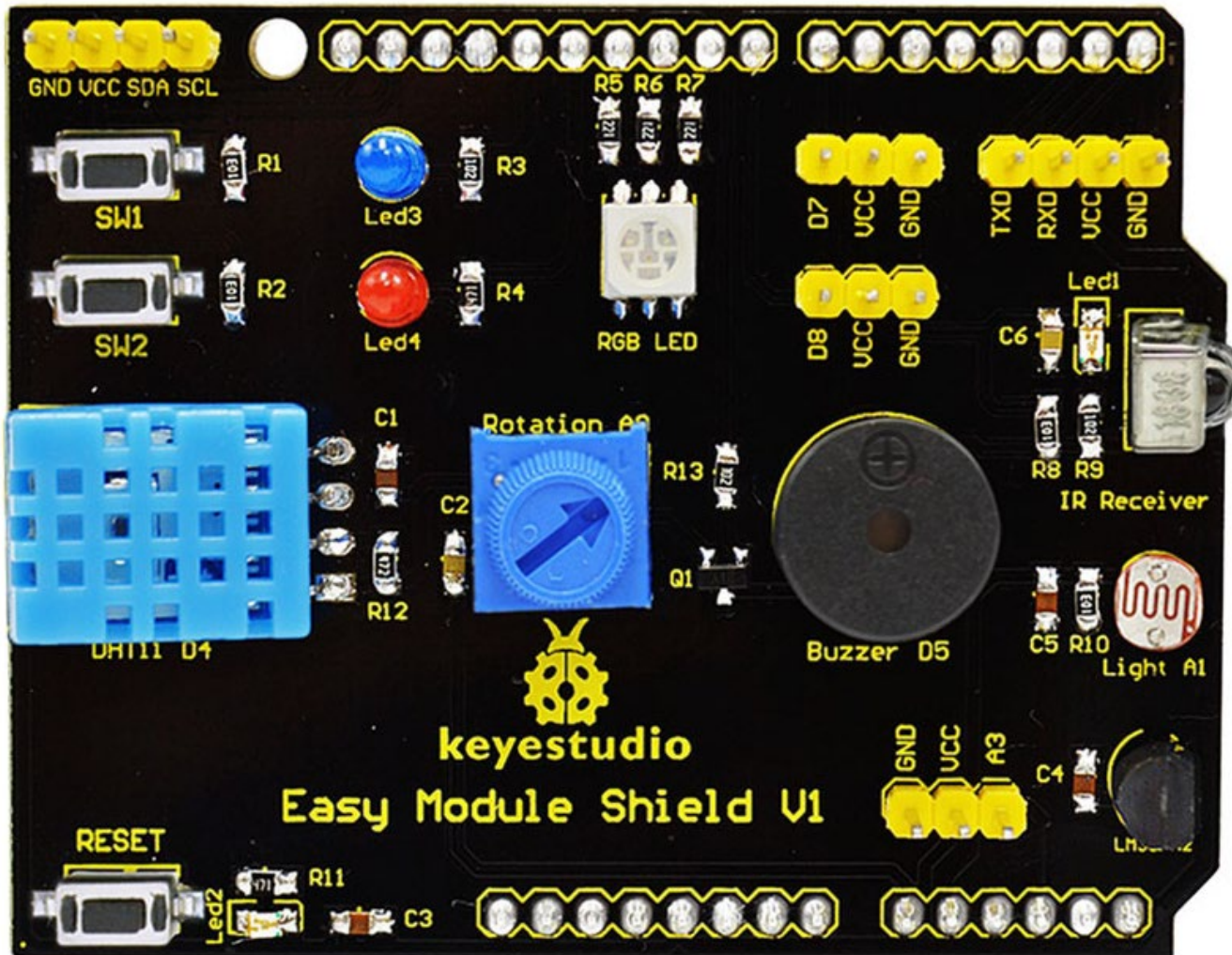
1-800-972-2225  
In NJ 732-381-8020





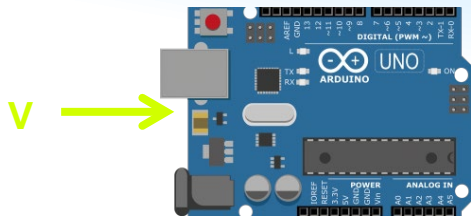
# Formas de prototipado

Escudos de aprendizaje...



Smart  
Open  
Lab

<http://smartopenlab.com>  
@SolEppc



# Entrada: INPUT



**Objetivo: leer estado de un botón**

- Botón
- Sensor puerta compostera



....

`pinMode(#,INPUT);`  
# es el número de pin

```
#define SW1 2 // · Cambiar por 3
```

```
void setup() {  
  Serial.begin (9600);  
  pinMode (SW1,INPUT);  
}
```

```
void loop() {  
  Serial.println(digitalRead(SW1));  
}
```

# entrada y salida

Sentencias de bifurcación: if



**Objetivo: hacer si...**

```
if (BOTON1_ACTIVO)
  hacer_algo1();
else
  hacer_algo2();
```



¡RETO!

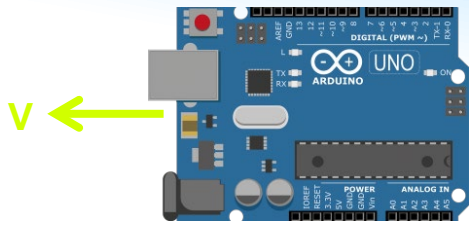


Si está pulsado  
imprimir un msj,  
sino otro

```
#define SW1 2 // · Cambiar por 3
```

```
void setup() {
  Serial.begin (9600);
  pinMode (SW1,INPUT);
}
```

```
void loop() {
  If (!digitalRead(SW1))
    Serial.println ("¡Pulsado!");
  else
    Serial.println ("Púlsame!");
}
```



# Salida: OUTPUT

Escritura digital



## Objetivo: encender LED

– LED



– Relé



....

`pinMode(#, OUTPUT);`

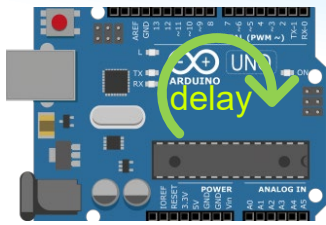
# es el número de pin

```
#define LED1 13
```

```
#define LED2 12
```

```
void setup() {  
    pinMode(LED1, OUTPUT);  
}
```

```
void loop() {  
    digitalWrite(13, HIGH);  
    //Cambiar HIGH por LOW  
}
```

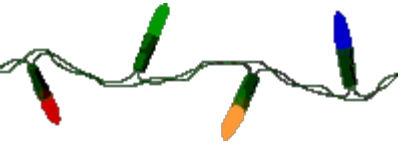


# Esperar

delay



Objetivo: escritura  
LED ¡que parpadea!



delay(x);

x indica los  
milisegundos que va a  
estar esperando el  
procesador

```
#define LED1 13
```

```
#define LED2 12
```

```
void setup() {  
    pinMode(LED1, OUTPUT);  
}
```

```
void loop() {  
    digitalWrite(LED1, LOW);  
    delay(500);  
    digitalWrite(LED1, HIGH);  
    delay(500);  
}
```

[Esta foto](#) de Autor desconocido  
está bajo licencia [CC BY-NC](#)

[Esta foto](#) de Autor desconocido está bajo licencia [CC BY-SA-NC](#) [Esta foto](#) de Autor desconocido está bajo licencia [CC BY-NC-ND](#)

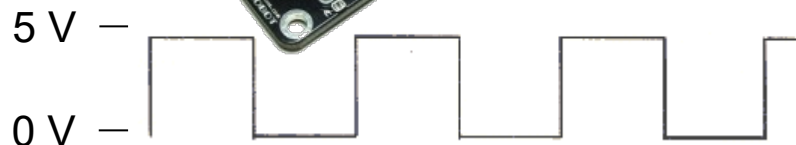


<http://smartopenlab.com>  
[@SolEppc](#)





# Digital vs Analógico



- Toma valores discretos:
  - 0 ó 1, LOW ó HIGH,...
  - Realmente es un voltaje:
    - HIGH (~ 2.7V-5V)
    - LOW (~ 0V-1.2V)



- Toma valores continuos:
  - Teóricamente infinitos
  - Al tratarlos con un micro controlador/procesador
    - 0,1,2,3,4,5,6... (no solo enteros) ← son voltajes

# Programación

Lectura analógica



Objetivo: leer la temperatura del LM35

**analogRead(#)**

# es el número de pin



¡RETO!



QUE NO SE  
ENCIENDAN LOS LED  
SI TEMP<30°C, UN  
LED SI TEMP>30°C y 2  
LED SI TEMP>35°C

```
#define LM35 A2
```

```
float temperatura;
```

```
void setup() {  
  Serial.begin(9600);  
  pinMode(LM35, INPUT);  
}
```

```
void loop() {  
  temperatura = (5.0 * analogRead(LM35)  
    * 100.0) / 1024;  
  Serial.println(temperatura);  
  delay(1000);  
}
```



```

#define LED1 13
#define LED2 12
#define LM35 A2

float temperatura;

void setup() {
    Serial.begin(9600);

    pinMode(LED1, OUTPUT);
    pinMode(LED2, OUTPUT);

    pinMode(LM35, INPUT);
}

void loop() {
    temperatura = (5.0 * analogRead(LM35) * 100.0) / 1024;
    Serial.print(temperatura);

    if (temperatura < 30) {
        digitalWrite(LED1, LOW);
        digitalWrite(LED2, LOW);
        Serial.println(" bajo");
    } else if (temperatura < 35) {
        digitalWrite(LED1, HIGH);
        digitalWrite(LED2, LOW);
        Serial.println(" medio");
    }
    else{
        digitalWrite(LED1, HIGH);
        digitalWrite(LED2, HIGH);
        Serial.println(" alto");
    }

    delay(1000);
}

```



# Programación

Lectura analógica



**Objetivo:** leer el sensor luminosidad (LDR)

**tocar un sonido**

tone(pin, freq, duracion);



¡RETO!



QUE SUENE  
ALERTANDO SI LA  
COMPOSTERA ESTÁ  
ABIERTA

```
#define LDR A1
```

```
int intensidadLuz;
```

```
void setup() {  
  Serial.begin (9600);  
  pinMode(LDR, INPUT);  
}
```

```
void loop() {  
  intensidadLuz = analogRead(LDR);  
  Serial.println ("intensidadLuz");  
  delay(1000);  
}
```

```
//Sensor de Luz, LDR
#define LDR A1

//Buzzer, motor vibrador
#define BUZZ 5

int intensidadLuz;

void setup() {
    Serial.begin(9600);

    pinMode(LDR, INPUT);
    pinMode(BUZZ, OUTPUT);
}

void loop() {
    intensidadLuz = analogRead(LDR);
    Serial.println(intensidadLuz);
    if (intensidadLuz > 100)
        tone(BUZZ, 58, 40);
    delay(1000);
}
```







¡RETO FINAL!



QUE SE ENCIENDAN LUCES  
DEPENDIENDO DE LA  
TEMPERATURA DE LA  
COMPOSTERA

QUE SUENE UNA ALARMA  
SI LA TAPA ESTÁ ABIERTA

```

//LEDs azul y rojo
#define LED1 13
#define LED2 12

//Sensor de temperatura 1
#define LM35 A2 //ACCURATE TEMP

//Sensor de Luz, LDR
#define LDR A1

//Buzzer, motor vibrador
#define BUZZ 5

int intensidadLuz;
float temperatura;

void setup() {
  Serial.begin(9600);

  pinMode(LED1, OUTPUT);
  pinMode(LED2, OUTPUT);

  pinMode(LM35, INPUT);

  pinMode(LDR, INPUT);
  pinMode(BUZZ, OUTPUT);
}

```

```

void loop() {
  temperatura = (5.0 * analogRead(LM35) * 100.0) / 1024;
  Serial.print(temperatura);

  if (temperatura < 30) {
    digitalWrite(LED1, LOW);
    digitalWrite(LED2, LOW);
  } else if (temperatura < 35) {
    digitalWrite(LED1, HIGH);
    digitalWrite(LED2, LOW);
  }
  else{
    digitalWrite(LED1, HIGH);
    digitalWrite(LED2, HIGH);
  }

  intensidadLuz = analogRead(LDR);
  Serial.println(intensidadLuz);
  if (intensidadLuz > 100)
    tone(BUZZ, 58, 40);

  delay(1000);
}

```

