

# GUIA SOBRE ARDUINO



<b>1. Introducción.....</b>	<b>3</b>
<b>2. Hardware de Arduino.....</b>	<b>3</b>
Modelos de Arduino.....	3
Componentes del Arduino Uno.....	3
<b>3. Software de Arduino.....</b>	<b>3</b>
Instalación del IDE de Arduino.....	3
Primer programa: "Blink".....	3
<b>4. Conceptos Básicos de Programación.....</b>	<b>4</b>
Estructura del Código.....	4
Funciones principales (setup y loop).....	4
<b>5. Conexiones Básicas.....</b>	<b>4</b>
Entradas y Salidas Digitales.....	4
Entradas y Salidas Analógicas.....	4
<b>6. Proyectos Iniciales.....</b>	<b>5</b>
Control de un LED.....	5
Lectura de un Sensor de Temperatura.....	5
Control de un Motor DC.....	6
<b>7. Recursos Adicionales.....</b>	<b>6</b>

## 1. Introducción

Arduino es una plataforma de hardware libre que facilita la creación de proyectos de electrónica. Combina una placa microcontroladora con un entorno de desarrollo (IDE) que permite escribir y subir código fácilmente.

## 2. Hardware de Arduino

### Modelos de Arduino

Existen varios modelos de Arduino, cada uno con diferentes características. Algunos de los más comunes son:

- **Arduino Uno:** Ideal para principiantes.
- **Arduino Mega:** Tiene más pines y memoria, útil para proyectos más complejos.
- **Arduino Nano:** Más pequeño, perfecto para proyectos compactos.

### Componentes del Arduino Uno

- **Microcontrolador:** ATmega328P.
- **Pines Digitales:** 14 (6 de ellos pueden ser usados como salidas PWM).
- **Pines Analógicos:** 6.
- **Voltaje de Operación:** 5V.
- **Memoria Flash:** 32 KB.
- **Cristal Oscilador:** 16 MHz.
- **Conector USB:** Para programar y alimentar la placa.
- **Conector de Alimentación:** Entrada de voltaje externo (7-12V).

## 3. Software de Arduino

### Instalación del IDE de Arduino

1. Ve a la página oficial de [Arduino](https://www.arduino.cc).
2. Descarga la versión correspondiente a tu sistema operativo (Windows, macOS, Linux).
3. Sigue las instrucciones de instalación.

### Primer programa: "Blink"

1. Conecta tu Arduino al computador usando un cable USB.
2. Abre el IDE de Arduino.
3. Ve a **Archivo > Ejemplos > 01.Basics > Blink**.
4. Selecciona tu placa y puerto en **Herramientas > Placa y Herramientas > Puerto**.
5. Haz clic en el botón **Subir** (flecha hacia la derecha).

Este programa hará parpadear un LED conectado al pin 13 del Arduino Uno.

## 4. Conceptos Básicos de Programación

### Estructura del Código

El código de Arduino se compone de dos funciones principales:

```
void setup() {  
    // Inicialización de variables, pines, etc.  
}  
  
void loop() {  
    // Código a ejecutar continuamente.  
}
```

### Funciones principales (**setup** y **loop**)

- **setup()**: Se ejecuta una vez cuando la placa se enciende o se reinicia. Aquí se configuran los pines y se inicializan las variables.
- **loop()**: Se ejecuta repetidamente. Aquí va el código que queremos que se repita continuamente.

## 5. Conexiones Básicas

### Entradas y Salidas Digitales

**Salida Digital**: Para encender/apagar LEDs, relés, etc.

```
pinMode(LED_BUILTIN, OUTPUT); // Configura el pin como salida  
digitalWrite(LED_BUILTIN, HIGH); // Enciende el LED  
digitalWrite(LED_BUILTIN, LOW); // Apaga el LED
```

- 

**Entrada Digital**: Para leer botones, interruptores, etc.

cpp

Copiar código

```
pinMode(buttonPin, INPUT); // Configura el pin como entrada  
int state = digitalRead(buttonPin); // Lee el estado del pin
```

- 

### Entradas y Salidas Analógicas

**Entrada Analógica**: Para leer sensores de temperatura, potenciómetros, etc.

cpp

Copiar código

```
int sensorValue = analogRead(A0); // Lee el valor del pin analógico  
A0
```

- 

**Salida Analógica (PWM):** Para controlar la intensidad de LEDs, velocidad de motores, etc.

```
analogWrite(ledPin, 128); // Escribe un valor PWM al pin (0-255)
```

- 

## 6. Proyectos Iniciales

### Control de un LED

1. Conecta un LED a un pin digital (por ejemplo, pin 13) con una resistencia de 220 ohmios en serie.

Escribe el siguiente código:

```
void setup() {  
    pinMode(13, OUTPUT);  
}  
  
void loop() {  
    digitalWrite(13, HIGH); // Enciende el LED  
    delay(1000); // Espera 1 segundo  
    digitalWrite(13, LOW); // Apaga el LED  
    delay(1000); // Espera 1 segundo  
}
```

- 2.

### Lectura de un Sensor de Temperatura

1. Conecta un sensor de temperatura LM35 al pin A0.

Escribe el siguiente código:

```
void setup() {  
    Serial.begin(9600);  
}  
  
void loop() {  
    int sensorValue = analogRead(A0);  
    float voltage = sensorValue * (5.0 / 1023.0);  
    float temperatureC = voltage * 100; // Convertir voltaje a temperatura  
    Serial.println(temperatureC);  
    delay(1000);  
}
```

2.

## Control de un Motor DC

1. Conecta un transistor (NPN) con una resistencia de 1k ohm al pin digital 9 y al motor DC.

Escribe el siguiente código:

```
void setup() {  
    pinMode(9, OUTPUT);  
}  
  
void loop() {  
    analogWrite(9, 128); // Controla la velocidad del motor (0-255)  
    delay(5000);  
    analogWrite(9, 0); // Apaga el motor  
    delay(5000);  
}
```

2.

## 7. Recursos Adicionales

- Documentación Oficial de Arduino
- Tutoriales en Arduino Project Hub
- Foro de Arduino