

Arduino: Entradas / Salidas



- Estructuras Propias de Arduino
- LED RGB con Arduino
- Pines digitales de Arduino pinMode, digitalWrite y digitalRead

Vamos a analizar el funcionamiento de los puertos disponibles en la placa Arduino Nano.

Digitales

Trabajan únicamente con 1 y 0: Señales digitales binarias.

Pueden programarse tanto como entradas como salidas, de forma independiente unas de otras.

Por defecto, al iniciar el Arduino, están configuradas como entradas. Esto es así para protegerlas frente a eventuales cortocircuitos:

- Si algún pin resulta que va a ser una entrada, entonces se le estará aplicando una tensión externa al mismo. Si ese mismo pin se configura inicialmente como salida se producirá un cortocircuito en dicha situación.

Lo que haremos por programa es programar las que vayan a ser salidas.

1) Estado inicial. Simplemente configurando los puertos digitales como salidas... ¿Cuál es el valor por defecto de las mismas, 1 ó 0?

```
// Declaración de variables
int persianal_sube, persianal_baja;

// Inicialización de variables
persianal_sube=3;
persianal_baja=4;

void setup() {

  // Inicializo persiana 1: pin salida y valor 1
  digitalWrite(persianal_sube, HIGH);
  pinMode(persianal_sube, OUTPUT);
  digitalWrite(persianal_baja, HIGH);
  pinMode(persianal_baja, OUTPUT);

  // Prueba para ver valor por defecto
  pinMode(5, OUTPUT);
}

void loop() {

}
```

- Al ejecutar este programa, se ve que los pines configurados como salidas tienen un valor 0 por defecto.
- Si estos pines van a estar conectados a interfaces que se activen con 0 deberemos previamente ponerlos a 1 para que no lo activen al inicio del programa, tal como se muestra en el ejemplo anterior donde los relés se activan con un nivel bajo y cuando programamos como salidas del Arduino los pines que los van a controlar deberemos estar seguros de que no se activan en ese momento.

Analógicas

Los pines analógicos que se numeran del PC0 al PC5 en el ATmega328P y **del A0 al A5** en la placa de Arduino, son pines analógicos pero también **se pueden utilizar como pines digitales**. Se puede acceder a ellos como si fueran pines digitales de Arduino utilizando la numeración **del 14 al 19**. Siendo el 14 el pin A0, el 15 el A1 y así sucesivamente hasta el 19.

Importante. Sin embargo los pines analógicos adicionales que posee el Arduino Nano **A6 y A7 no pueden funcionar en este modo de pines digitales**, siendo únicamente entradas analógicas. *Estos pines solo tienen capacidades ADC y no se pueden usar como entradas o salidas digitales.*

- Así, el siguiente código funciona en un Arduino Nano, poniendo un 0 y un 1 de forma intermitente en el terminal A0 de la placa

[salida_analogica1.ino](#)

```
byte pin_color = A0;

void setup(void) {
  pinMode(pin_color, OUTPUT);
} // Final setup

void loop(void) {
  digitalWrite(pin_color, 0); // Salida con '0'
  delay(2000);
  digitalWrite(pin_color, 1); // Salida con '1'
  delay(2000);
} // Final loop
```

- Sin embargo si intentamos hacer lo mismo con el pin A6 (byte pin_color = A6;) **no funciona**.

From:
<https://euloxio.myds.me/dokuwiki/> - **Euloxio wiki**

Permanent link:
https://euloxio.myds.me/dokuwiki/doku.php/doc:tec:prg:uc_arduino:arduino_io:inicio

Last update: **2025/11/05 12:32**

