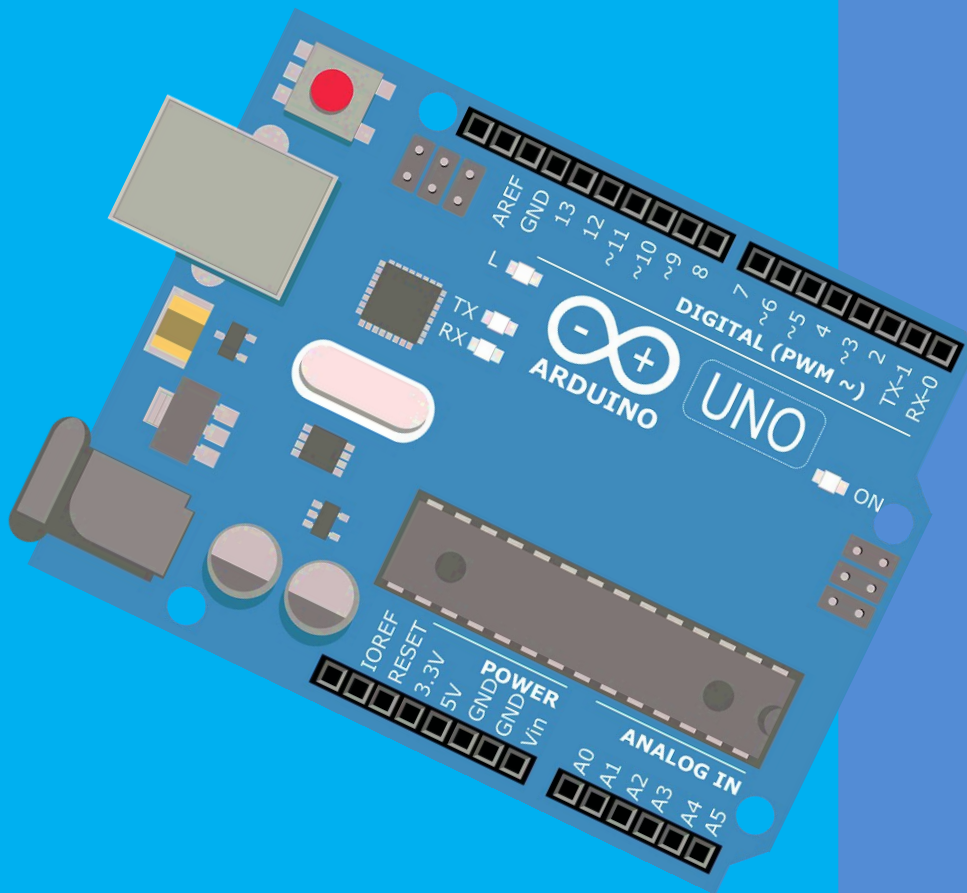


# ARDUINO NIVEL 1

**Hagamos una  
secuencia de luces**

**Instructor: Konrad Peschka**



# Curso de Arduino nivel 1

## Hagamos una secuencia de luces

Para interactuar con los pines digitales de Arduino ya conocemos las funciones que nos ofrece Arduino: `pinMode()`, `delay()`, y `digitalWrite()`.

Recordaremos que una señal digital puede variar únicamente entre dos valores, que denominamos  $-V_{cc}$  y  $+V_{cc}$ . Una salida digital es un dispositivo que permite variar su tensión a uno de estos dos valores mediante programación, y por tanto nos permite realizar acciones con el entorno.

En Arduino, en general, los voltajes  $-V_{cc}$  y  $+V_{cc}$  corresponden con 0V (GND) y 5V.

## Intensidad máxima de una salida digital

La intensidad máxima que puede dar un pin es de 40 mA, si bien el valor recomendado es de 20mA. Además, hay restricciones adicionales en cuanto a la potencia, como que la suma total de todas las salidas debe ser inferior a 300 mA, y a su vez no pueden superar 150 mA por puerto.

Esta potencia es suficiente para encender un led, un pequeño servomotor de 9g, o encender algún sensor, pero no es suficiente para alimentar cargas mayores. Si queremos mover una carga superior, como un motor de corriente continua, un servo, o incluso un relé, tendremos que emplear una etapa de amplificación, como un transistor BJT.

No es conveniente forzar los límites de potencia de forma prolongada, la placa podría calentarse y dañarse. El límite de 20 mA por salida significa que, para un voltaje de 5V, la resistencia del dispositivo que queramos alimentar no debe ser inferior a 200 ohmios.

# Curso de Arduino nivel 1

## Secuencia de luces

Para hacer la secuencia de luces el código es similar al usado en la clase anterior, usando las mismas funciones y alterando el orden podremos realizar el proyecto.

Primero debemos asignar los puertos digitales correspondientes como salida, para ello en el bloque Setup, y utilizando la función "pinMode()", definiremos los puertos como salida (OUTPUT).

```
void setup ()
{
  pinMode (5, OUTPUT);
  pinMode (6, OUTPUT);
  pinMode (7, OUTPUT);
  pinMode (8, OUTPUT);
}
```

Bloque Setup

Una vez definidos los puertos como salida, podremos emplear las funciones digitalWrite();y delay(); en el bloque Loop para que se ejecuten continuamente.

```
void loop ()
{
  digitalWrite (5, HIGH);
  delay(1000);
  digitalWrite (6, HIGH);
}
```

Bloque Loop

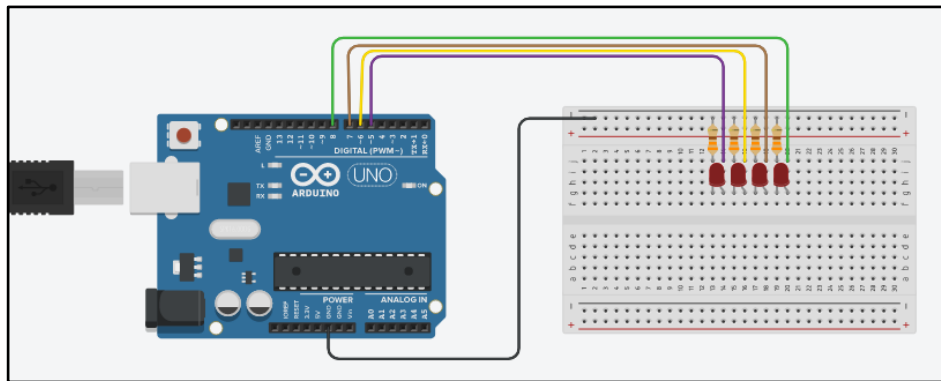
Alternando las funciones y variando sus valores podremos realizar el código necesario.

# Curso de Arduino nivel 1

## Circuito Propuesto

El circuito propuesto consta de un Arduino que hará encender y apagar una secuencia de luces conectadas en los pines: 5,6 7, 8, para ello usaremos:

- 1 Arduino uno
- 1 Protoboard
- 4 Leds rojo
- 4 Resistencias 330  $\Omega$



Circuito: Secuencia de luces

# Curso de Arduino nivel 1

## Solución de la Programación

| Programa: Secuencia de luces   |  |
|--|--|
| <pre>void setup() {   pinMode (5, OUTPUT);   pinMode (6, OUTPUT);   pinMode (7, OUTPUT);   pinMode (8, OUTPUT); }</pre>  | <pre>// Función Setup // llave de apertura del bloque Setup //Asignación de puertos como salida  //llave de cierre del bloque Setup</pre>  |
| <pre>void loop() {   digitalWrite (5, HIGH);   delay(1000);   digitalWrite (6, HIGH);   delay(1000);   digitalWrite (7, HIGH);   delay(1000);   digitalWrite (8, HIGH);   delay(1000);   digitalWrite (8, LOW);   delay(1000);   digitalWrite (7, LOW);   delay(1000);   digitalWrite (6, LOW);   delay(1000);   digitalWrite (5, LOW);   delay(1000); }</pre> | <pre>//Función Loop // llave de apertura del bloque Loop //Puerto 5 en estado alto // retardo de 1000 millisecond(s)  //Puerto 8 en estado bajo  // retardo de 1000 millisecond(s)</pre> |