

**ARDUINO Y PROGRAMACION  
EN EL DESARROLLO DE PROYECTOS DE CIENCIAS  
Y  
ELECTRONICA**

# ¿Qué es ARDUINO?

Ideas?

# Arduino UNO

Arduino es una **plataforma de creación de electrónica de código abierto**, la cual está basada en hardware y software libre, flexible y fácil de utilizar para los creadores y desarrolladores. Esta plataforma **permite crear diferentes tipos de microordenadores de una sola placa** a los que la comunidad de creadores puede darles diferentes tipos de usos.

**Tera**  
Electronics

## ● Comunicación Serial (UART)

RX -Pin 0 (Recepción)  
TX -Pin 1 (Transmisión)

## ● Comunicación I2C

SDA -Pin 18 (Serial DAta)  
SCL -Pin 19 (Serial Clock)

## ● Centrada Analógica

A0 -Pin 14  
A1 -Pin 15  
A2 -Pin 16  
A3 -Pin 17  
A4 -Pin 18  
A5 -Pin 19

## ● Interrupciones Externas

INT0 -Pin 2  
INT0 -Pin 3

## ● Comunicación SPI

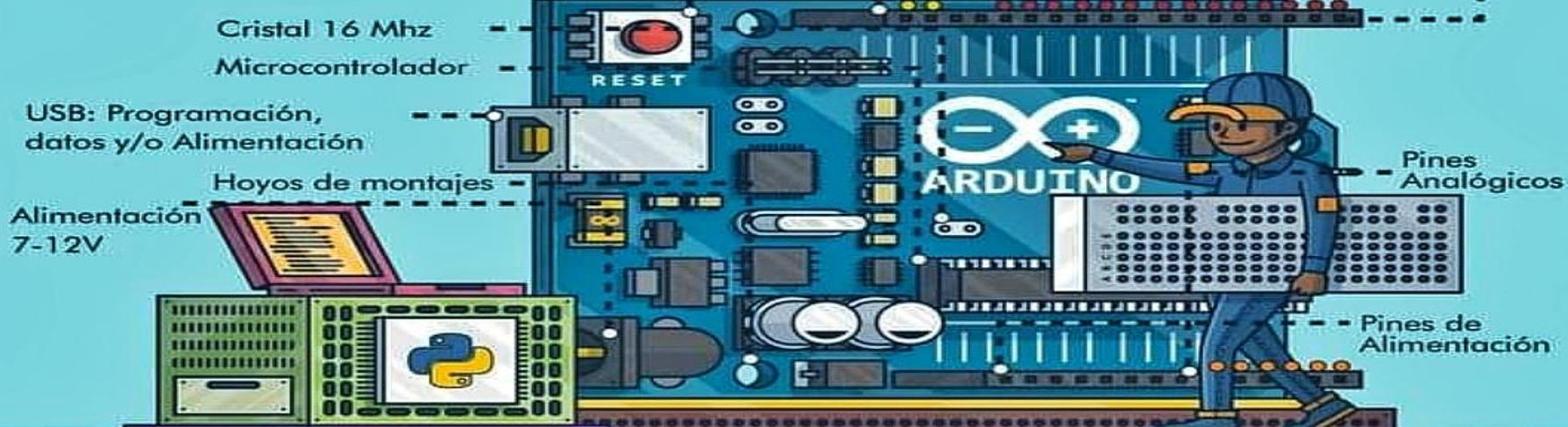
SS -Pin 10  
MOSI -Pin 11  
MISO -Pin 12  
SCK -Pin 13

## ● Entrada / Salida Digital (GPIO)

Pines 0 al 19

## ● Salida PWM~

Pines 3, 9, 10, 11  
Pines 5 y 6





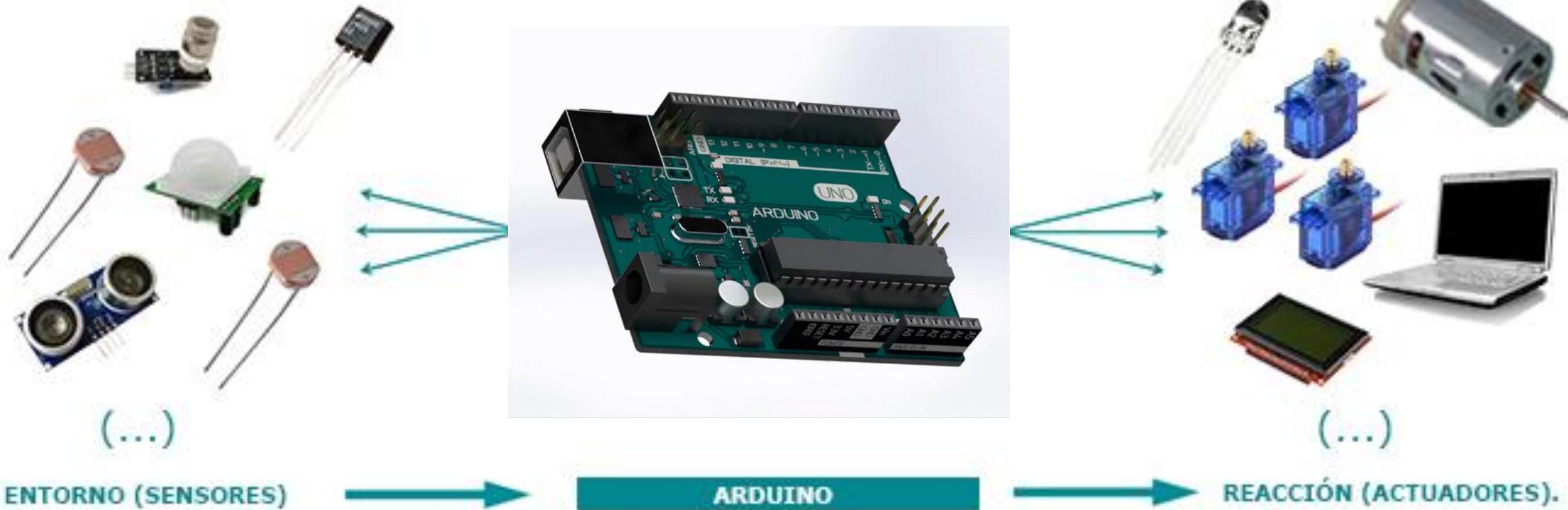
# ¿Qué es Arduino?

- ♾️ Plataforma de desarrollo basado Hardware y software libre.
- ♾️ El microcontrolador en la placa Arduino se programa mediante el lenguaje de programación Arduino (Basado en Wiring).
- ♾️ El entorno de desarrollo Arduino (IDE) está basado en processing.
- ♾️ Multiplataforma (Linux, Mac, Windows).
- ♾️ Facilidad de uso y bajo costo.
- ♾️ Al alcance de todos los usuarios: estudiantes, artistas, publicistas, ingenieros, etc.

# ¿Qué es Arduino?

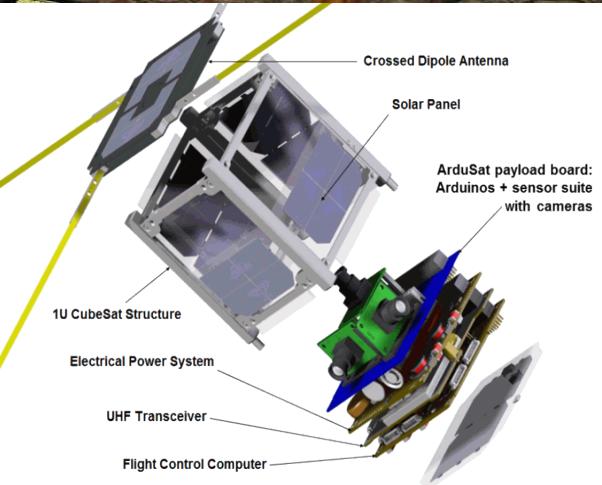
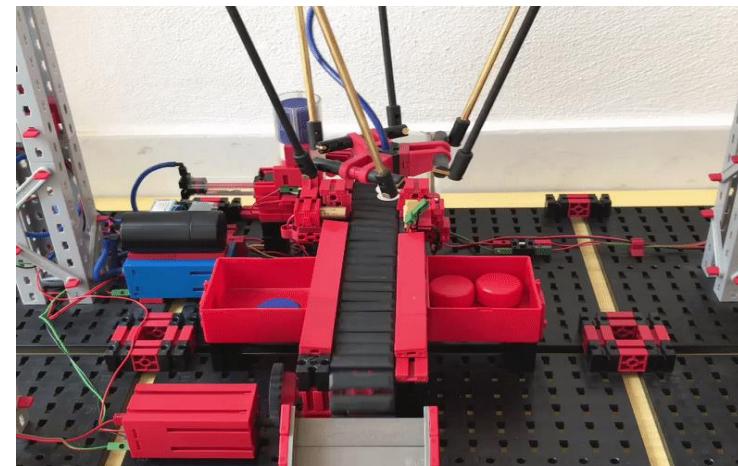
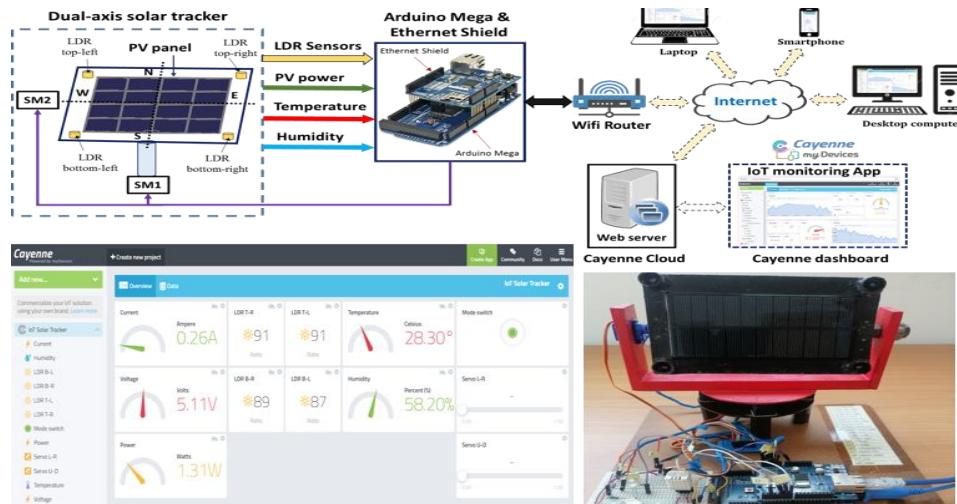
**Arduino = HW + SW + Comunidad**

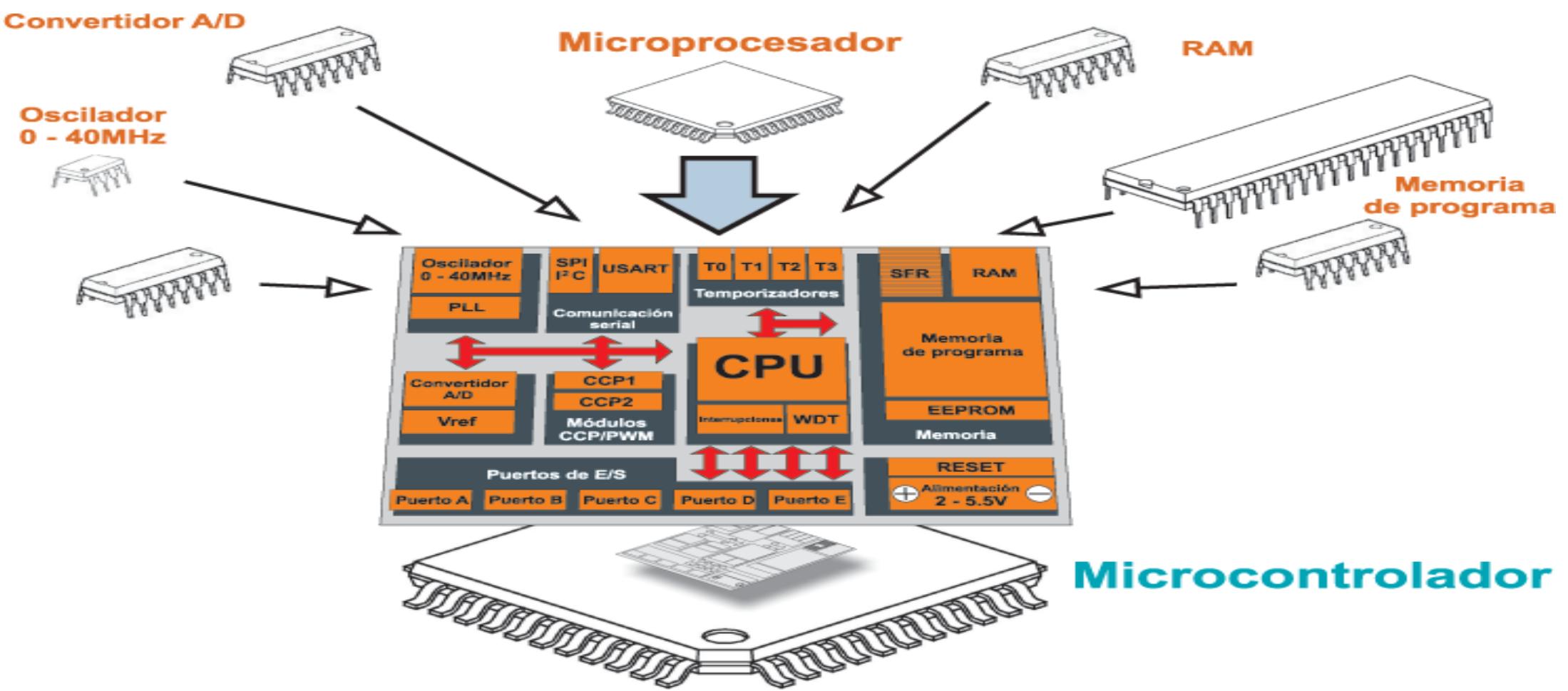
**Filosofía Código Abierto / Open Source**



# ¿PARA QUÉ SIRVE ARDUINO?

La tarjeta Arduino es muy potente y capaz de realizar cualquier tipo de proyectos, **la limitación es la imaginación.**





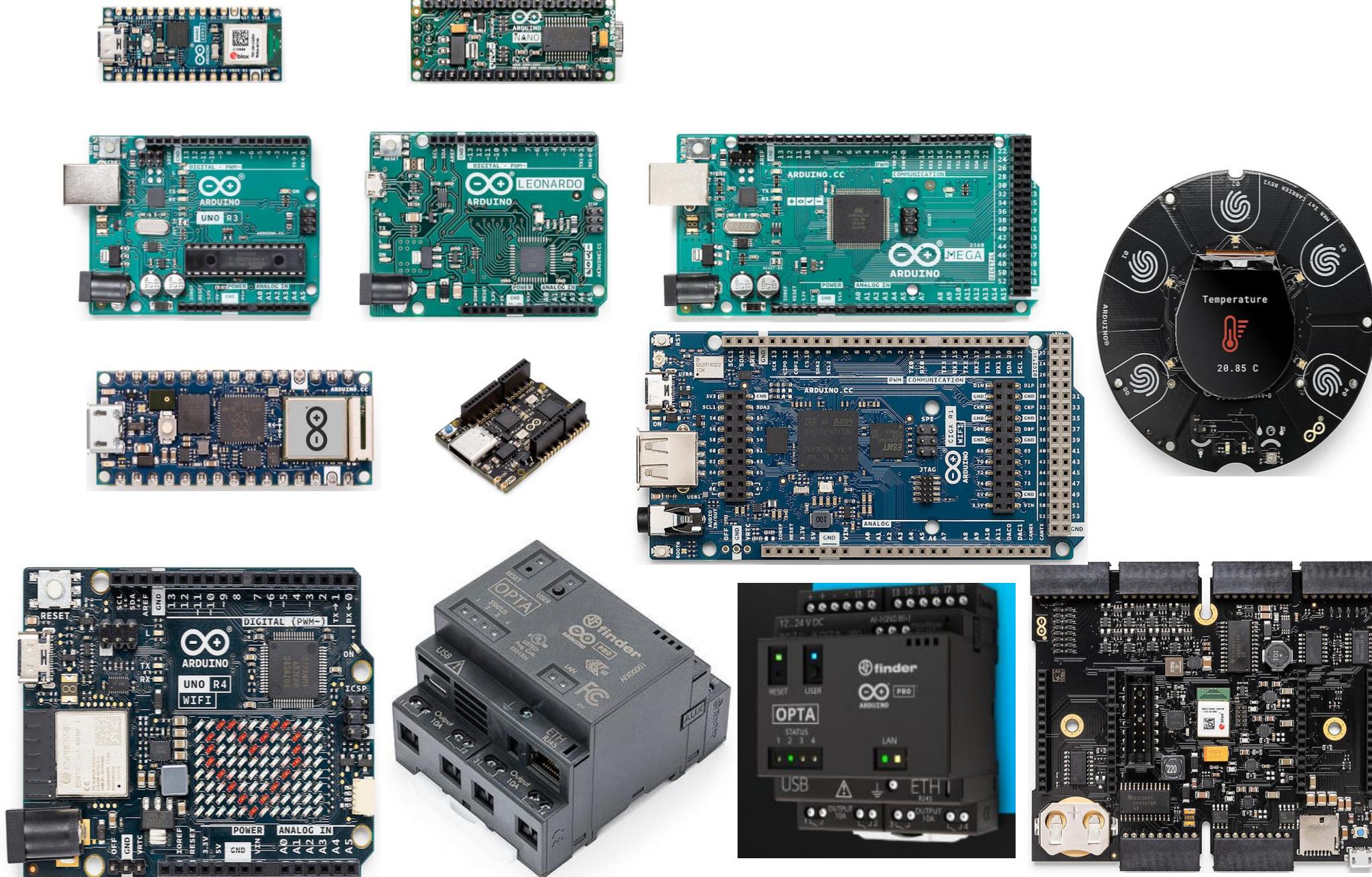
## Se basa en un microcontrolador

En concreto en los de la familia ATMega. Esto es importante para entender su diferencia con Raspberry Pi . Mientras que un procesador convencional a día de hoy es lo suficientemente complejo como para poder ejecutar varios procesos al mismo tiempo en paralelo, un microcontrolador solo puede con uno al mismo tiempo.

**Y todo se basa en la filosofía DIY**

# **PLATAFORMAS ARDUINO**

# ARDUINO TODAY



## Industrial

- Yun/Yun Mini
- Zero
- M0/M0 Pro
- Tian
- 101/Industrial 101



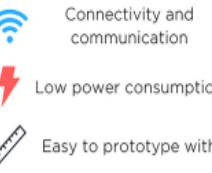
## Educational

- Explora
- Robot



## IoT

- MKR1000
- MKRZero
- MKRFOX1200
- Uno Wi-Fi
- Ethernet
- Primo



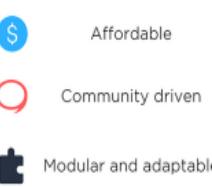
## Wearables

- LilyPad
- LilyPad Simple
- LilyPad Snap
- LilyPad USB
- Primo Core



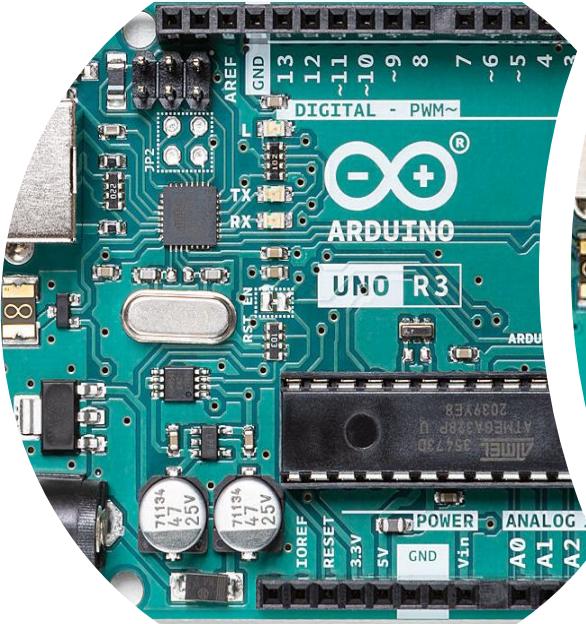
## Maker

- Uno
- Leonardo
- Mini/Pro Mini
- Nano/Micro
- Mega2560/ADK
- Primo
- Due

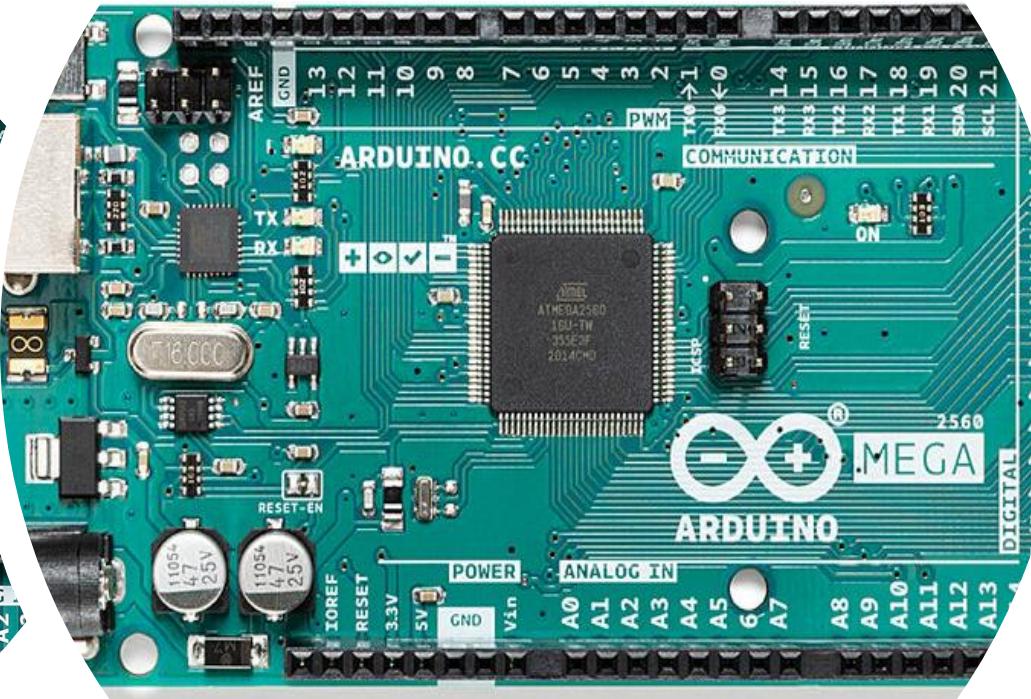


**PLATAFORMAS  
MAS  
USADAS**

# ARDUINO UNO

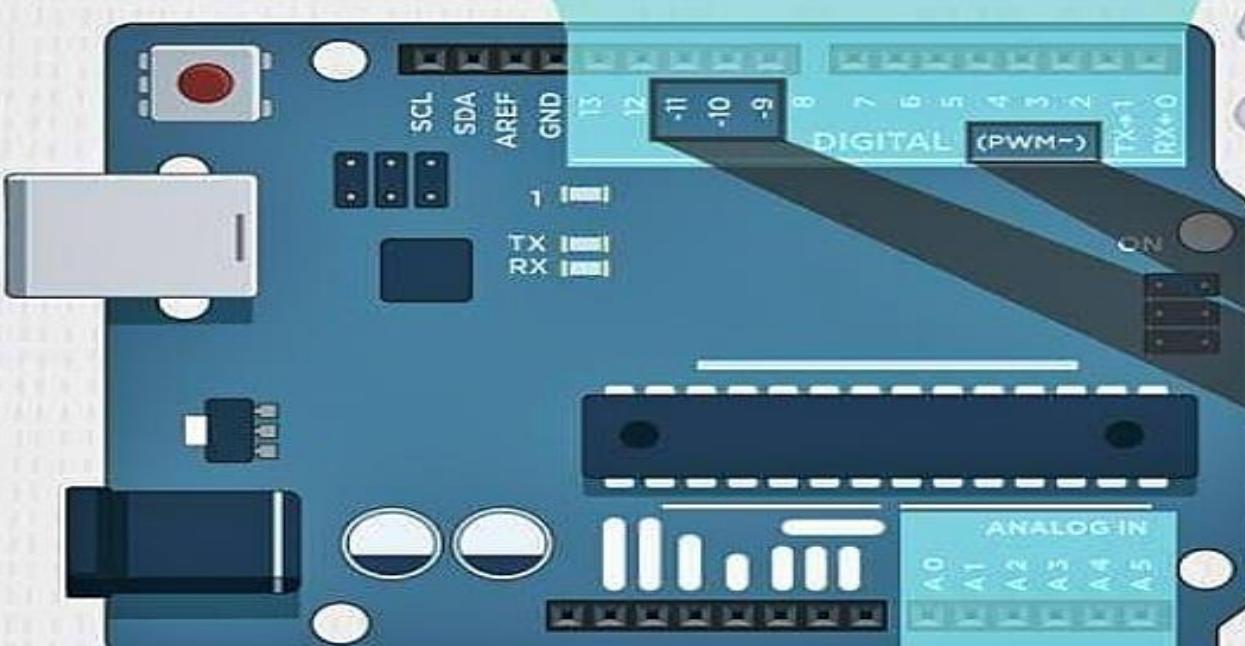


# ARDUINO MEGA



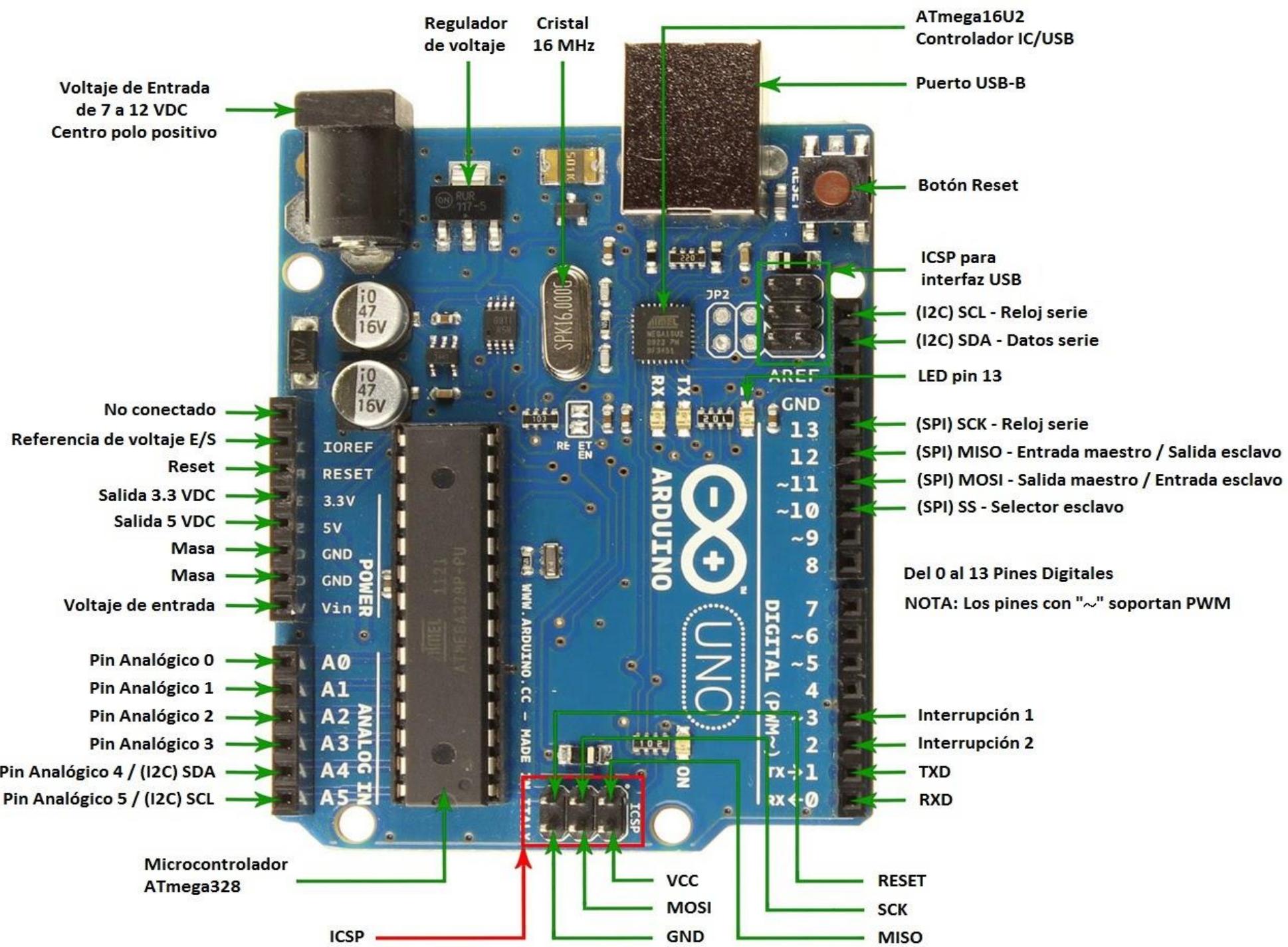
# ARDUINO LENARDO

También es posible encontrar tarjetas de desarrollo que ya tengan etiquetadas las posibles funciones de cada pin en la misma tarjeta

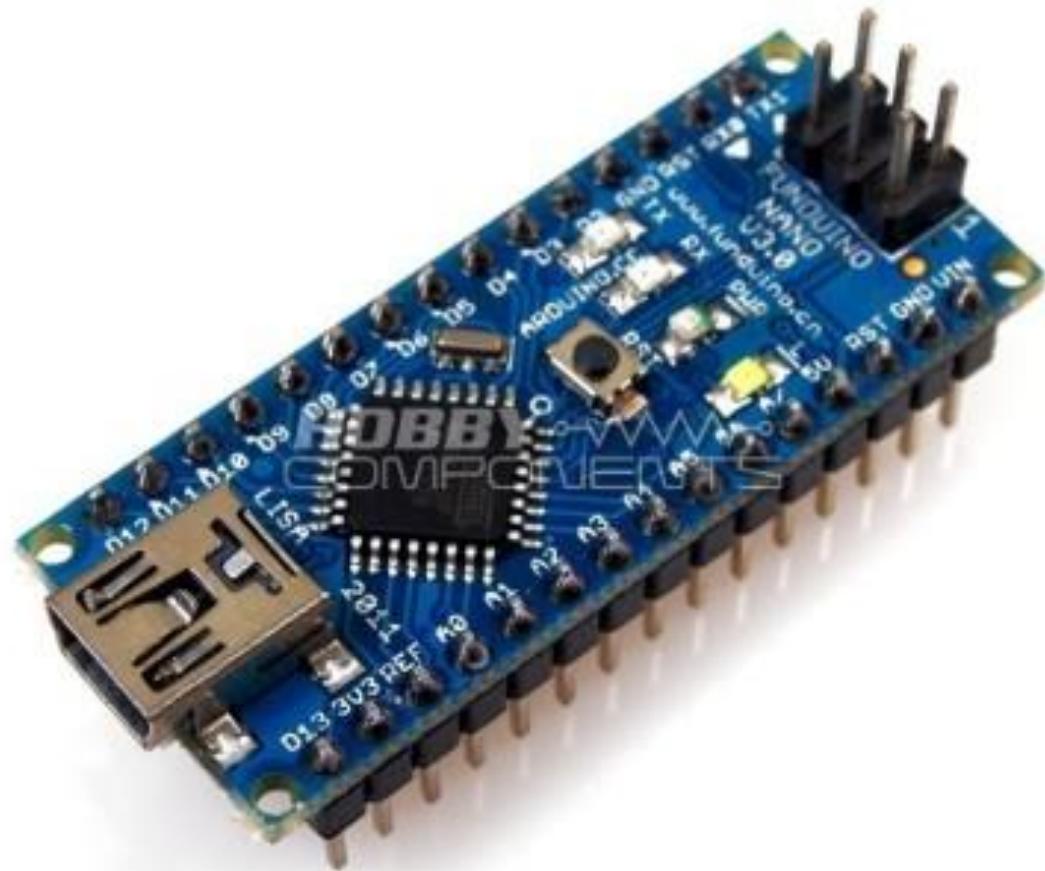


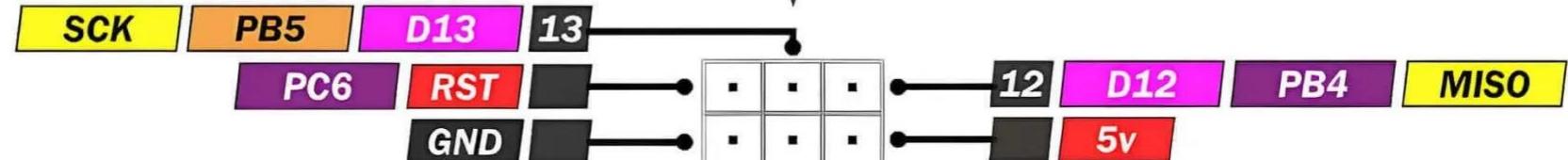
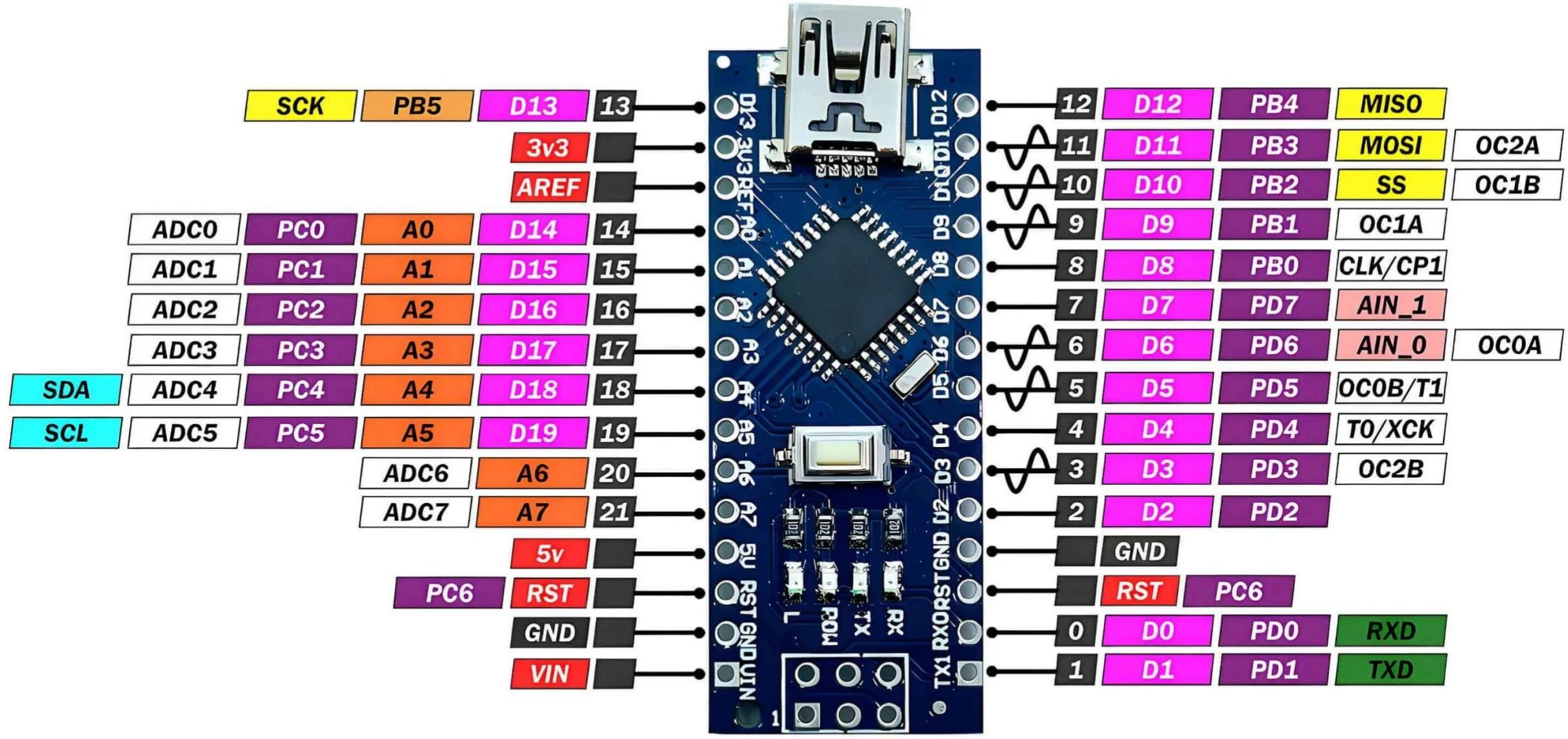
**Gata  
Electronics**

Es importante que en el microcontrolador con el que estés trabajando identifiques que tipos de pines tienes y cuales corresponden a cada uno para que al programarlos no obtengas errores y tus programas funcionen correctamente.



# ARDUINO NANO



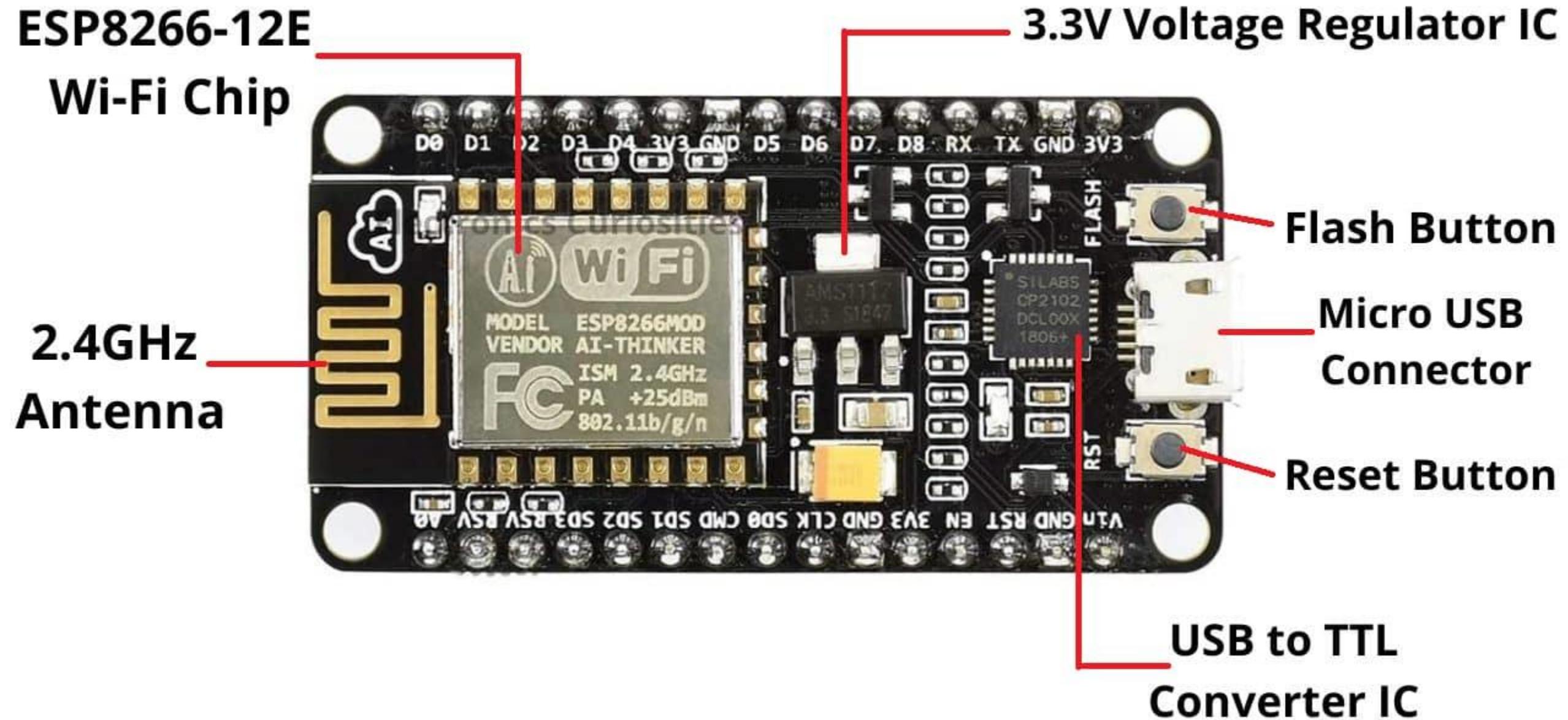


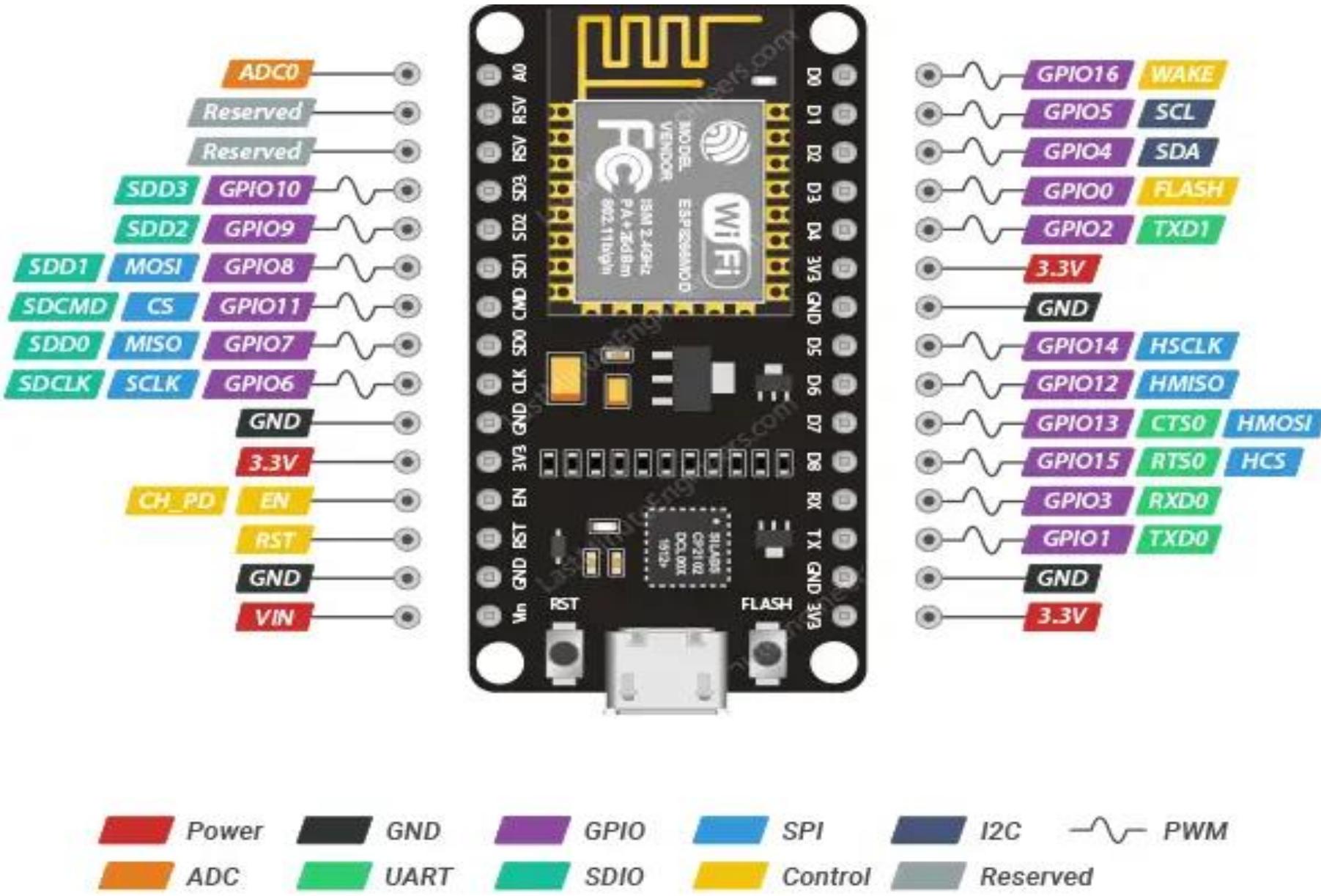
# PLATAFORMAS

## IOT

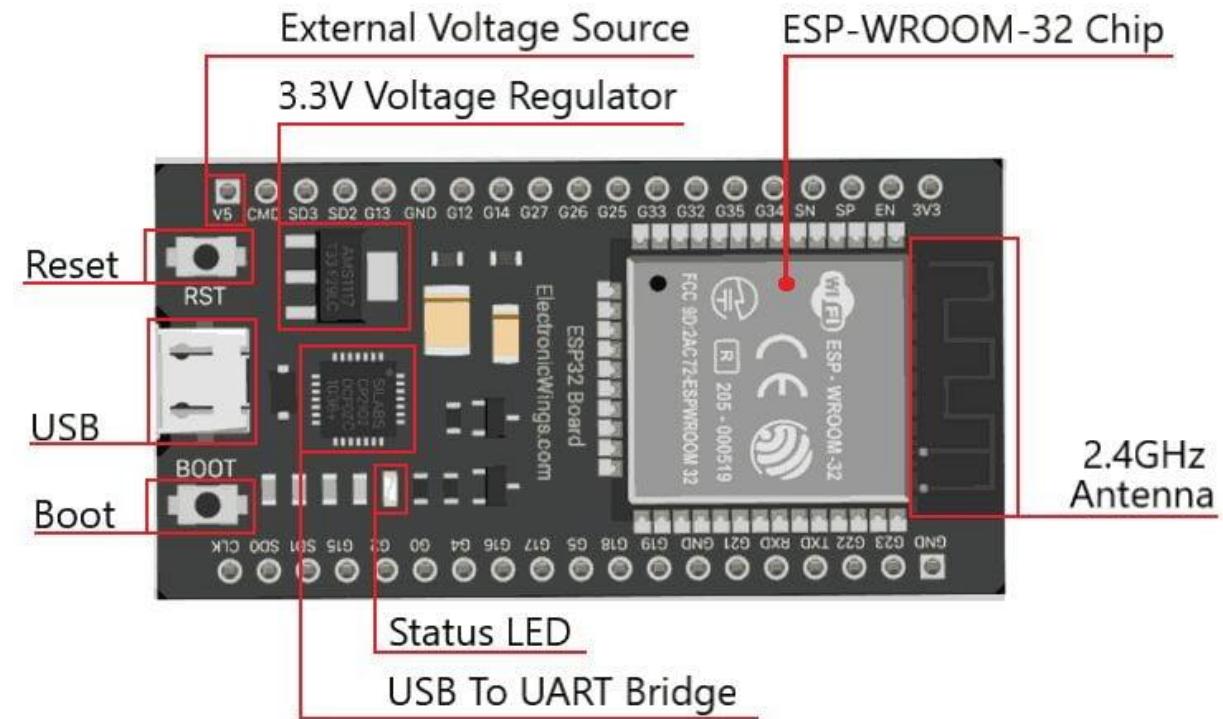
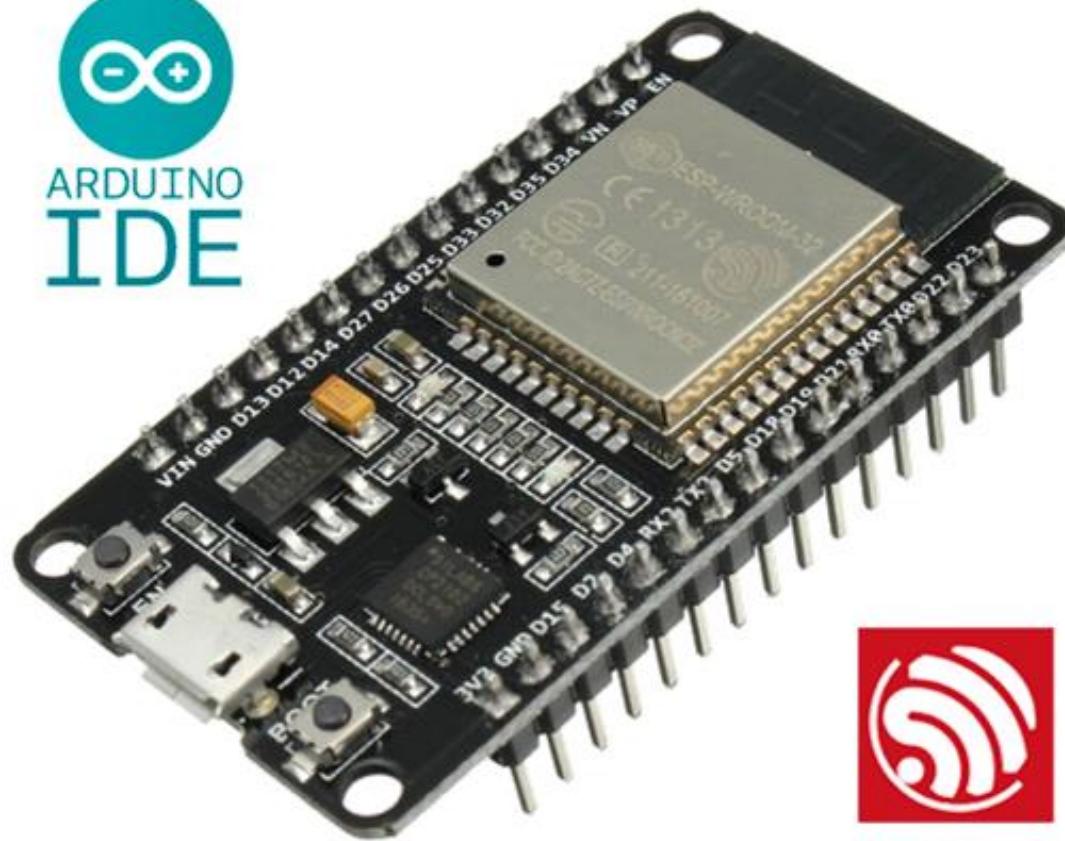
(Internet de las Cosas)

# ESP8266 NODE MCU





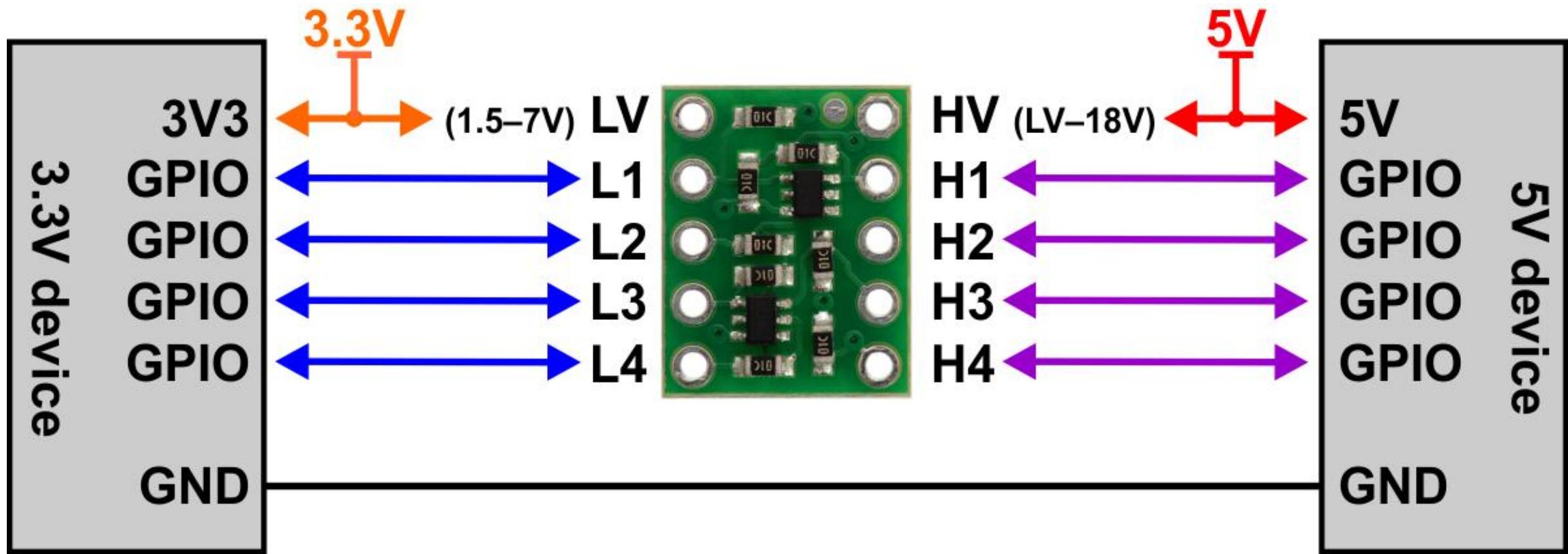
# *ESP32 - CPU 32 bits*



The ESP32 Chip explained: Advantages and Applications

## OBSERVACION:

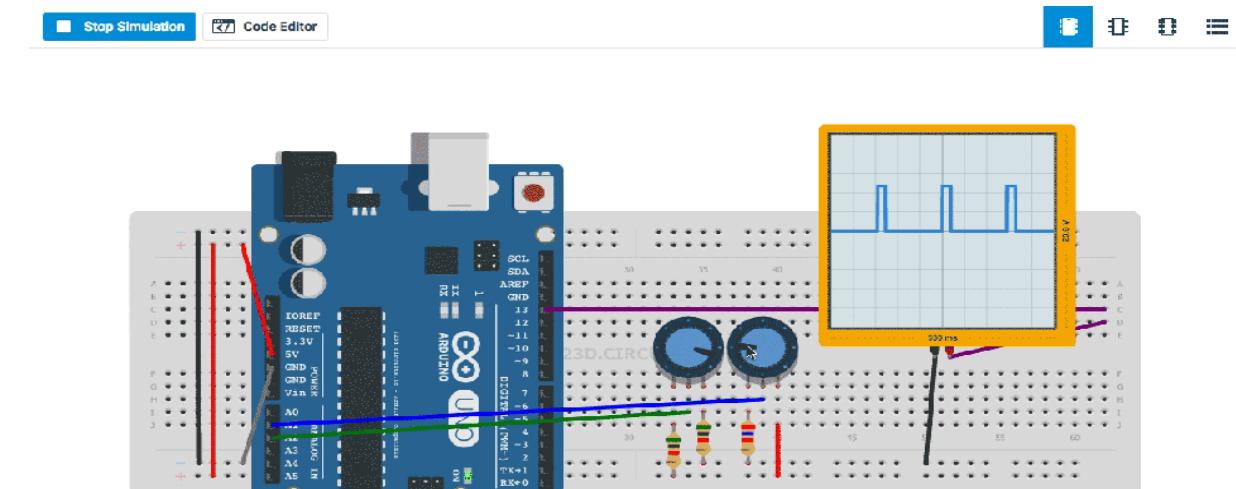
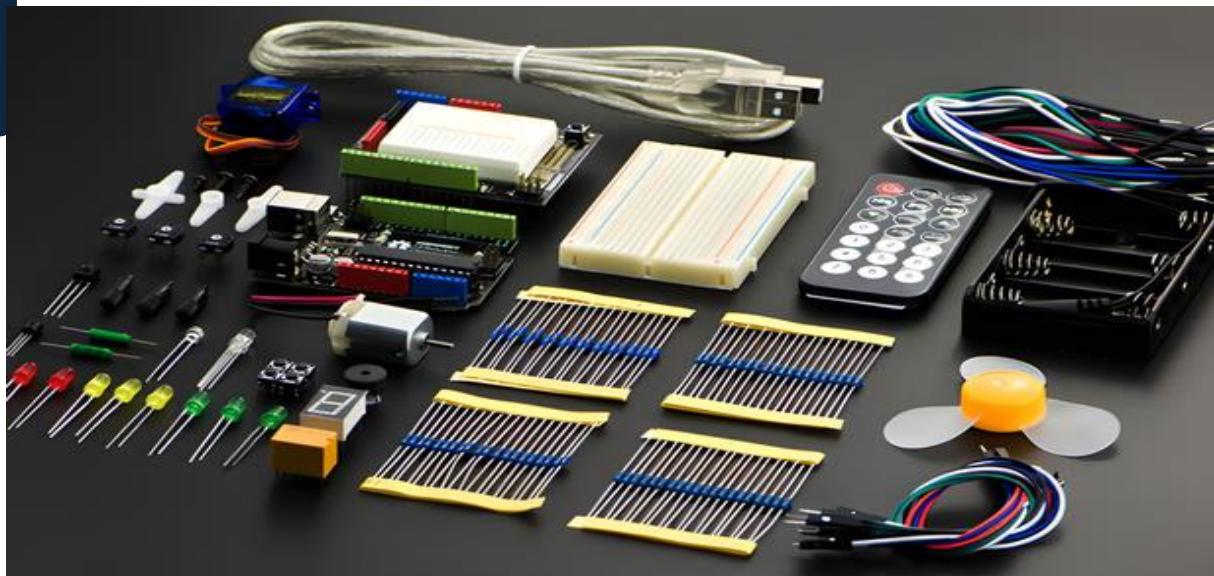
- Las plataformas IOT trabajan a 3.3Voltios de niveles lógicos y de conversor digital a analógico.
- Las plataformas genéricas(verde) trabajan a 5.0Volts.
- Es valido el uso de circuitos fuente a fin de alimentar y/o adaptar tensión de alimentación. Suelen venir incorporados en las plataformas de prototipado(12V)
- Cualquier interacción entre niveles lógicos distintos, debe hacerse por medio de conversores bidireccionales de tensión



# ¿Cómo usar Arduino?

- Ver Aquí: [Página Oficial](#)

# ¿Como usar Arduino?



sketch\_dec07a | Arduino 1.8.3

File Edit Sketch Tools Help

sketch\_dec07a

```
void setup() {
  // put your setup code here, to run once:
}

void loop() {
  // put your main code here, to run repeatedly:
}
```

2

Arduino/Genuino Uno on COM3

A screenshot of the Arduino IDE software. The title bar says "sketch\_dec07a | Arduino 1.8.3". The menu bar includes File, Edit, Sketch, Tools, and Help. The toolbar has icons for file operations. The code editor shows a sketch named "sketch\_dec07a" with the following code:

```
void setup() {
  // put your setup code here, to run once:
}

void loop() {
  // put your main code here, to run repeatedly:
}
```

The status bar at the bottom right shows "Arduino/Genuino Uno on COM3".

# Arduino - Home

The screenshot shows the Arduino website's main landing page. At the top, there's a navigation bar with links for PROFESSIONAL, EDUCATION, STORE, HARDWARE, SOFTWARE, CLOUD, DOCUMENTATION, and COMMUNITY. Below the navigation is a large image of an Arduino Uno board with the text "WHAT IS ARDUINO?". To the right of the board is a video thumbnail showing hands working with a blue spherical IoT device. Below the video is a call-to-action button labeled "BUY AN ARDUINO" with a shopping cart icon.

Nombre

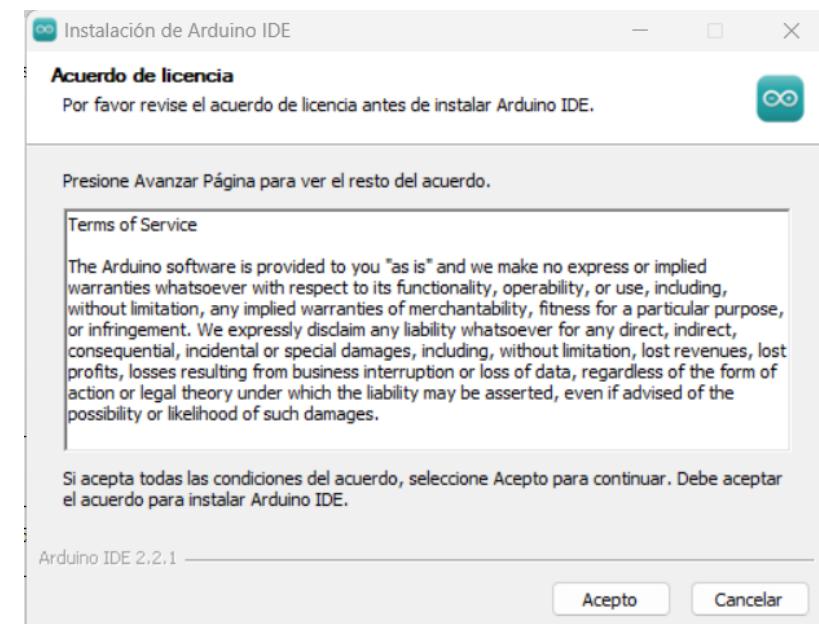
✓ hoy

arduino-ide\_2.2.1\_Windows\_64bit.exe

Fecha de modificación

2/11/2023 23:33

The screenshot shows the Arduino website's "Downloads" section. At the top, there's a navigation bar with links for HARDWARE, SOFTWARE (which is highlighted), CLOUD, DOCUMENTATION, COMMUNITY, BLOG, and ABOUT. Below the navigation is a section titled "Arduino Web Editor" with a brief description and two buttons: "CODE ONLINE" and "GETTING STARTED". To the right of the editor description is a screenshot of the Arduino Web Editor interface showing a list of sketches. Below this is a section for "Arduino IDE 2.2.1" with download links for various operating systems. A "Release Notes" link is also present.



# SINTAXIS DE REFERENCIA Y FUNCIONES

- Ver Aquí: [Guía de Referencia de Arduino](#)
- Funciones básicas: *pinMode()*, *digitalWrite()*,  
*digitalRead()*, *Serial.begin()*, *millis()*, *micros()*

# PARADIGMA DE LA POO (Programación Orientada Objeto)





# 01

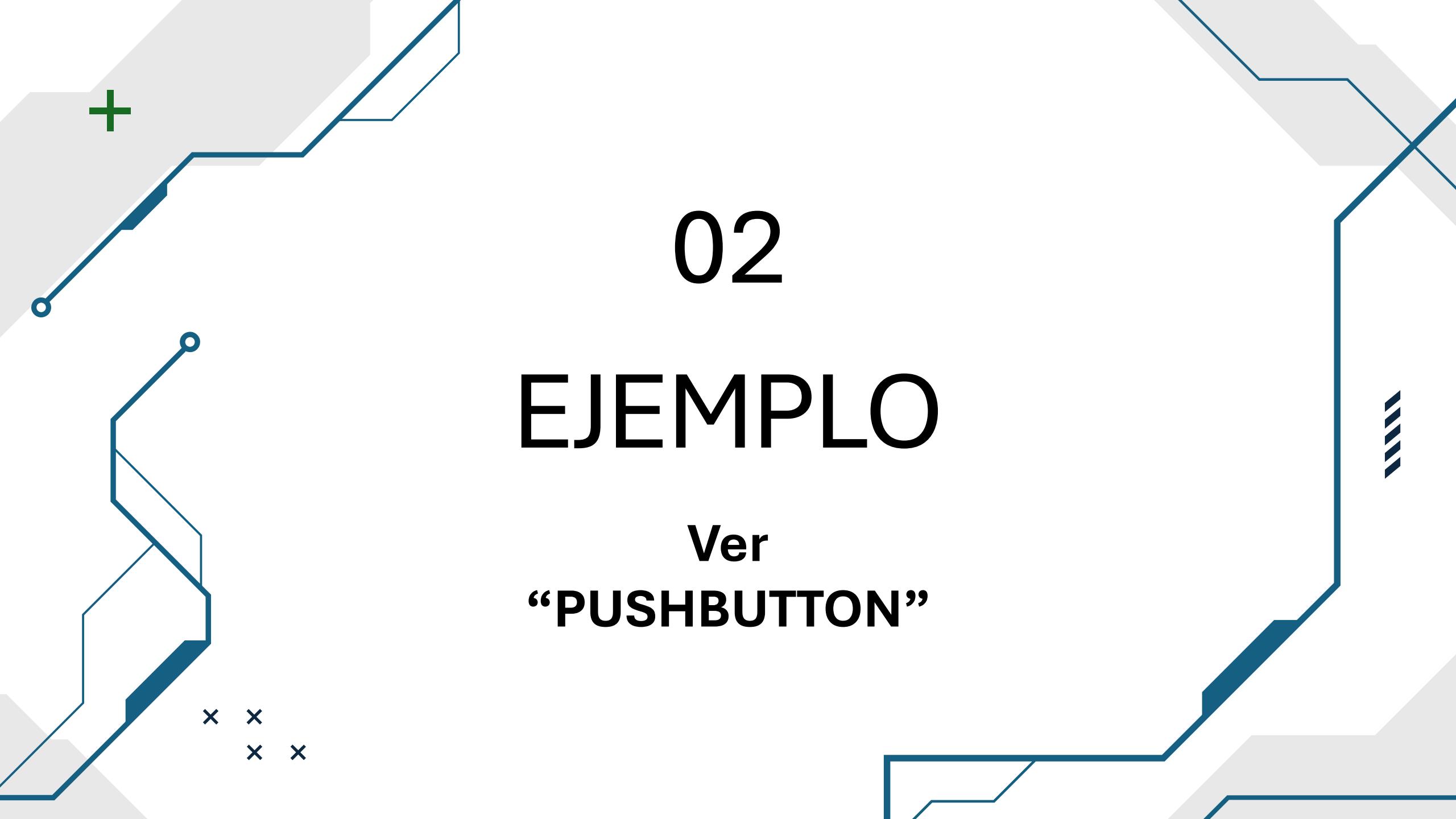
# EJEMPLO

Ver “Blink”

# Actividad de aprendizaje

El diseño ahora es hacer la temporización determinista de 2 actividades, mientras el diodo LED incorporado en placa puede parpadear cada 1 segundo o mas, el puerto de consola serial puede enviar datos muy rápido a unos 100 milisegundos con algún mensaje.

Nota: Revise a función millis()



# 02

## EJEMPLO

Ver  
**“PUSHBUTTON”**

# Actividad de aprendizaje

El diseño ahora es hacer hacer la lectura determinista de al menos 1 pulsador cuyo efecto al ser presionado sea el de realizar 1 acción por puerto serie por única vez. El programa no debe quedarse bloqueado ni la acción repetirse de forma indefinida.

Considerar el efecto rebote

Nota: Revise a función millis()

# PROYECTOS

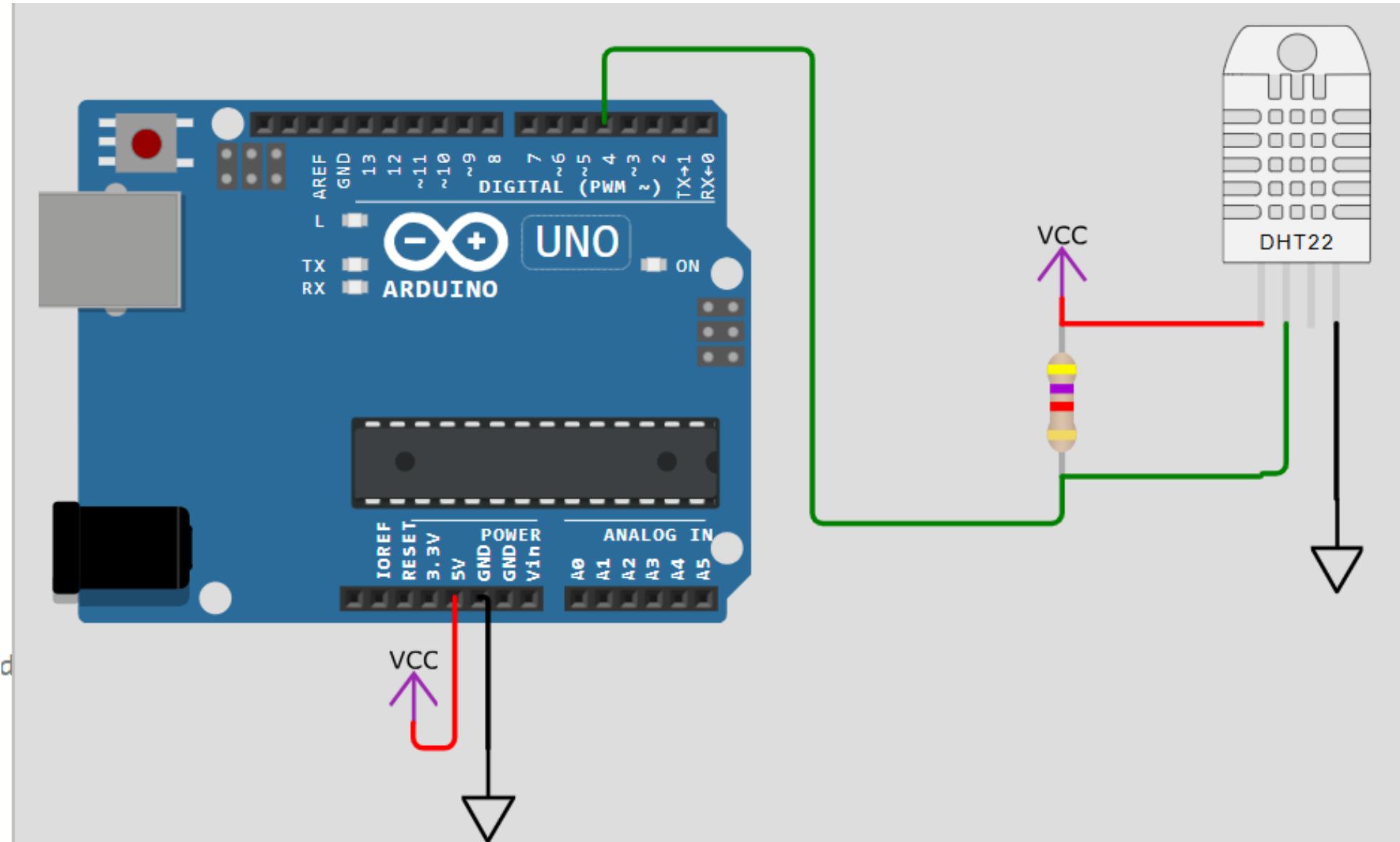
Leer  
Temperatura y  
Humedad

# Actividad de análisis

El diseño ahora es hacer hacer la lectura del sensor de temperatura y humedad(de ambiente) DHT11/DHT22 usando tanto el simulador WOKWI como implementado en protoboard.

- Integrar en las lecturas un **promediado móvil** mediante funciones ya vistas en lenguaje C
- Considerar tanto la Hoja de datos como la librería ya existente para Arduino

# CONEXIONES CIRCUITO SENSOR DHT11/DHT22 – uC – ARDUINO UNO



# Actividad de análisis

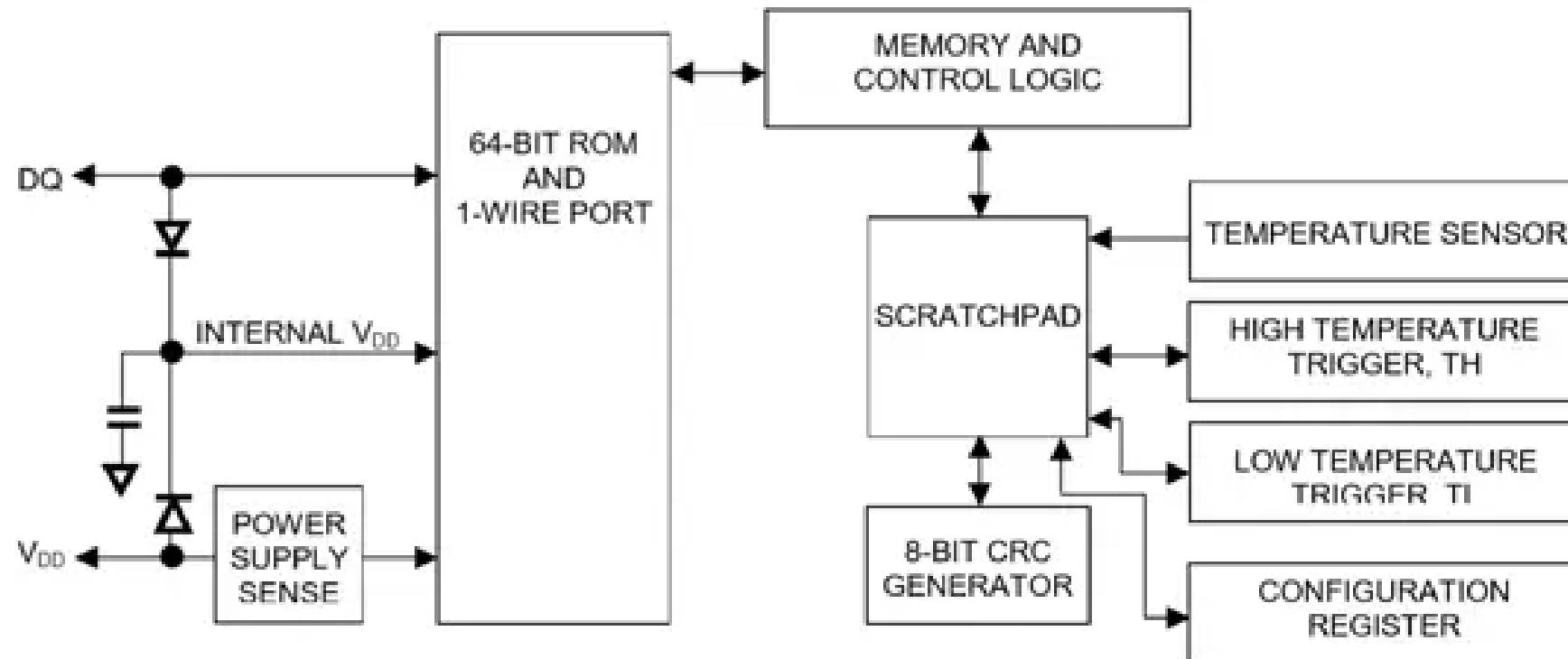
El diseño ahora es hacer hacer la lectura del sensor de temperatura DS18B20 (de líquidos/suelos/solidos)

- Integrar en las lecturas un promediado móvil mediante funciones ya vistas en lenguaje C
- Considerar la *Hoja de datos* como la *Librería* ya existente para Arduino. Considere también la librería *one wire*

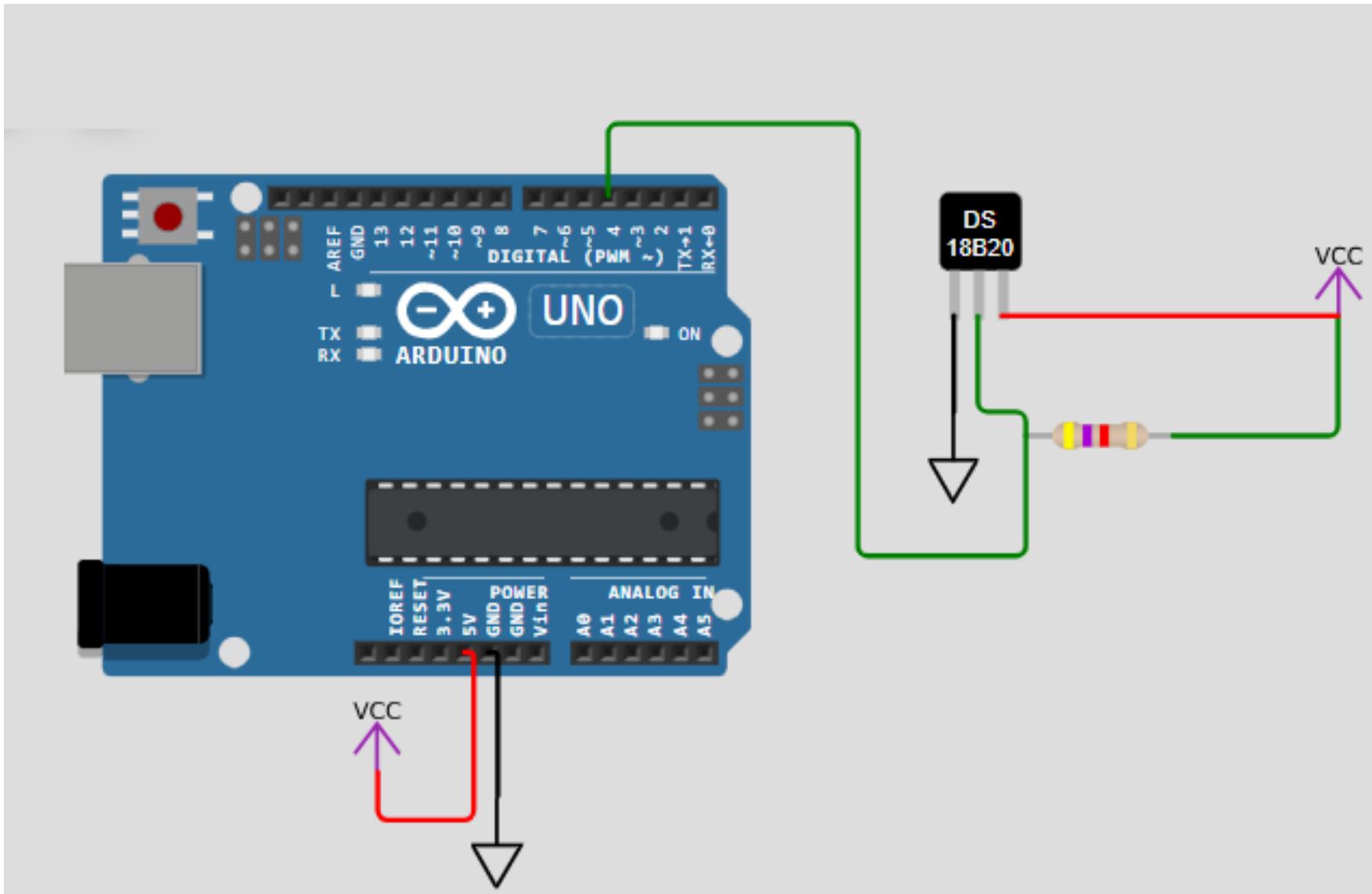
# PRESENTACIONES SENSOR DS18B20



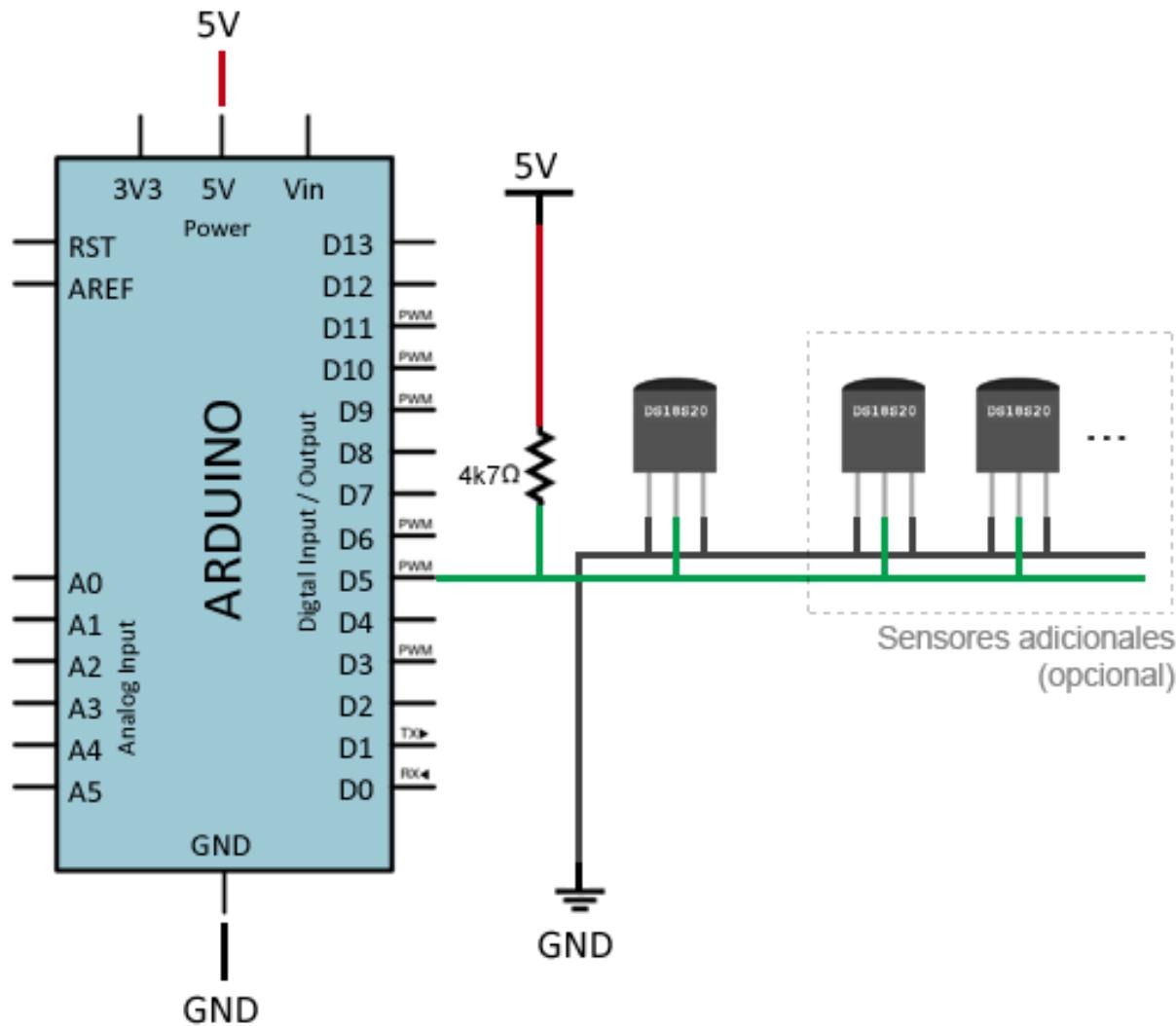
# ESTRUCTURA INTERNA DS18B20 (integrado)



# CONEXIONES CIRCUITO SENSOR DS18B20 – ARDUINO UNO



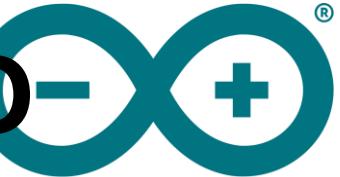
# LECTURA MULTI-SENSOR DS18B20



**Nota: Se puede leer mas de un sensor sobre un mismo BUS**

# **ANEXOS**

# Descripción de la placa Arduino UNO



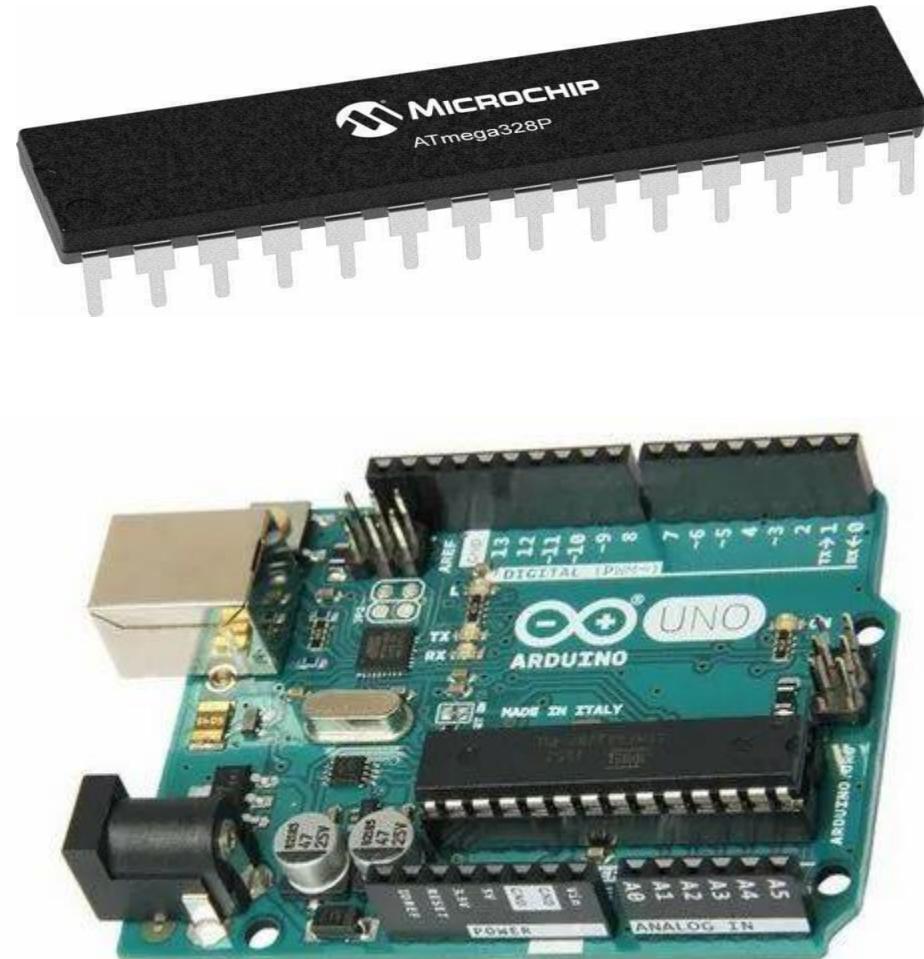
## Arduino function

reset	(PCINT14/RESET) PC6	1	28	PC5 (ADC5/SCL/PCINT13)
digital pin 0 (RX)	(PCINT16/RXD) PD0	2	27	PC4 (ADC4/SDA/PCINT12)
digital pin 1 (TX)	(PCINT17/TXD) PD1	3	26	PC3 (ADC3/PCINT11)
digital pin 2	(PCINT18/INT0) PD2	4	25	PC2 (ADC2/PCINT10)
digital pin 3 (PWM)	(PCINT19/OC2B/INT1) PD3	5	24	PC1 (ADC1/PCINT9)
digital pin 4	(PCINT20/XCK/T0) PD4	6	23	PC0 (ADC0/PCINT8)
VCC	VCC	7	22	GND
GND	GND	8	21	AREF
crystal	(PCINT6/XTAL1/TOSC1) PB6	9	20	AVCC
crystal	(PCINT7/XTAL2/TOSC2) PB7	10	19	PB5 (SCK/PCINT5)
digital pin 5 (PWM)	(PCINT21/OC0B/T1) PD5	11	18	PB4 (MISO/PCINT4)
digital pin 6 (PWM)	(PCINT22/OC0A/AIN0) PD6	12	17	PB3 (MOSI/OC2A/PCINT3)
digital pin 7	(PCINT23/AIN1) PD7	13	16	PB2 (SS/OC1B/PCINT2)
digital pin 8	(PCINT0/CLK0/ICP1) PB0	14	15	PB1 (OC1A/PCINT1)

## Arduino function

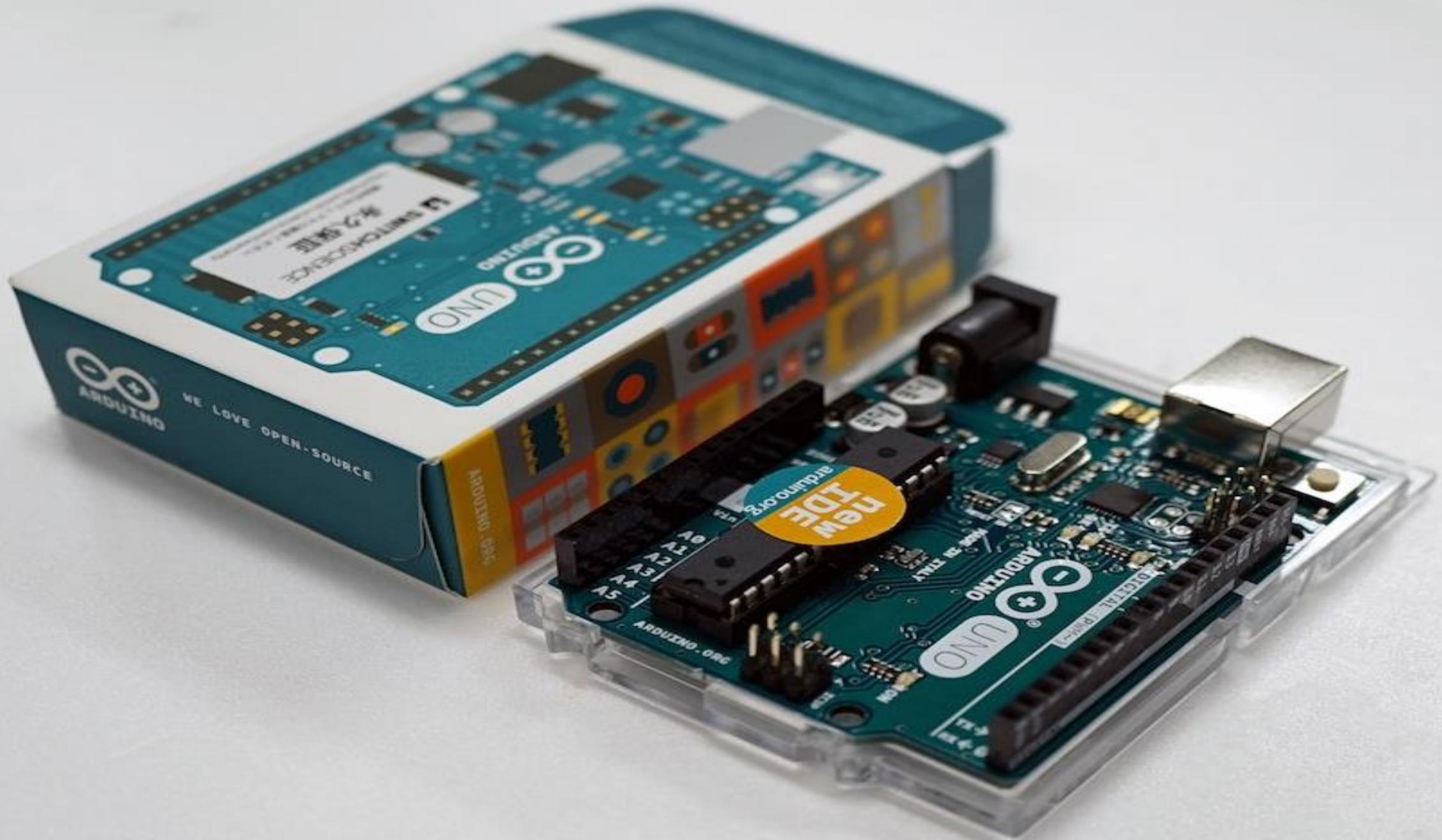
analog input 5
analog input 4
analog input 3
analog input 2
analog input 1
analog input 0
GND
analog reference
VCC
digital pin 13
digital pin 12
digital pin 11(PWM)
digital pin 10 (PWM)
digital pin 9 (PWM)

Digital Pins 11,12 & 13 are used by the ICSP header for MOSI, MISO, SCK connections (Atmega168 pins 17,18 & 19). Avoid low-impedance loads on these pins when using the ICSP header.



**ARDUINO UNO**

**EMPAQUE  
ORIGINAL**



# ESP32

## CARACTERISTICAS

### Espressif's ESP32-S3 Wi-Fi + Bluetooth LE SoC

