

# ***Sesión I***

INTRODUCCIÓN





# ***Temas de avance***

- Electrónica básica.
- Node Mcu.
- Tecnologías a usar (Resumen).
- Preguntas



# ***Introducción a electrónica básica***

¿Qué es la Electrónica?

- Es la Ciencia que se encarga del uso de los electrones para crear circuitos que resuelven y facilitan la vida de las personas.

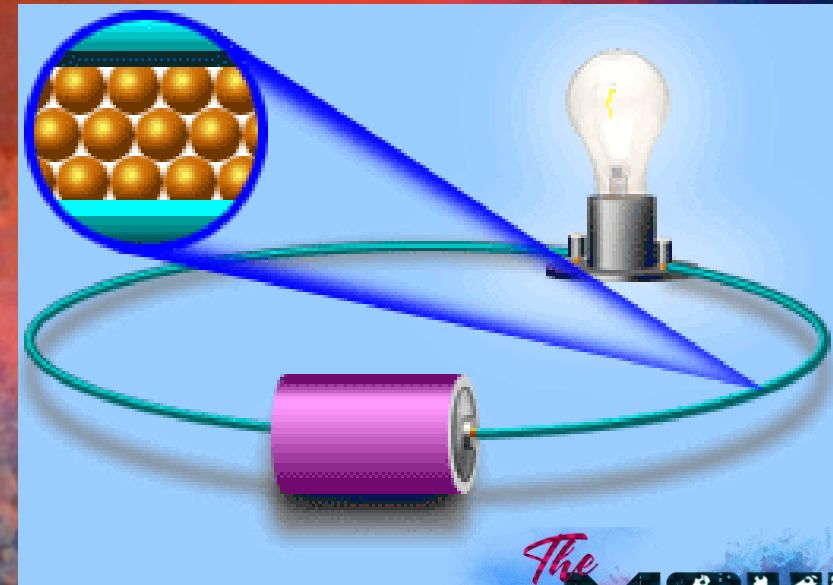
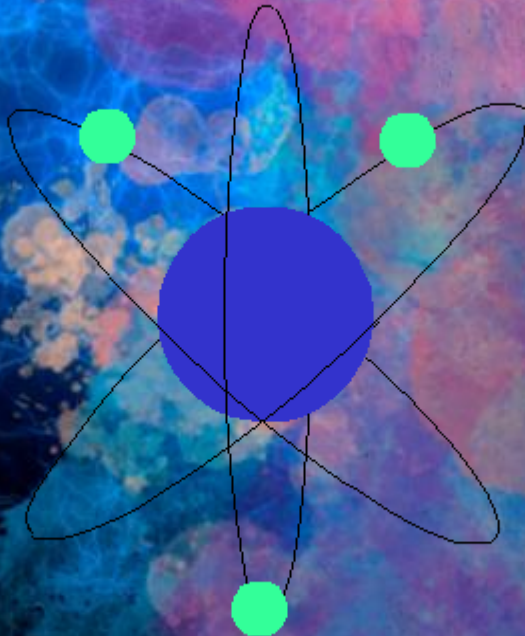




# ***Introducción a electrónica básica***

¿Qué son los Electrones?

- Son partículas pequeñas cargadas negativamente y en el Circuito Eléctrico se desplazan las cargas eléctricas haciendo que funcionen los equipos electrónicos.

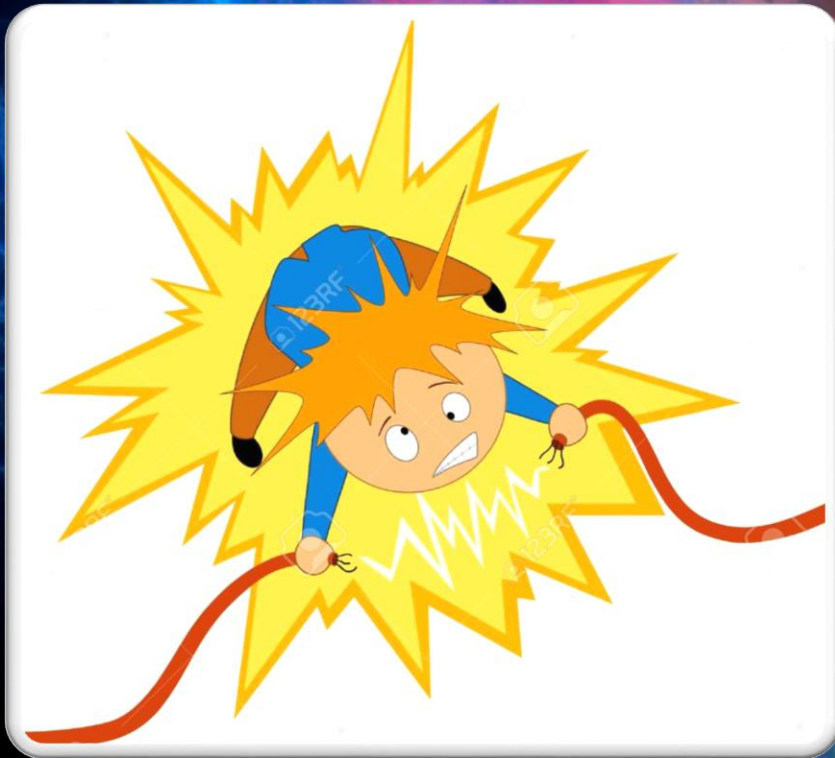




# ***Introducción a electrónica básica***

¿Qué es el Voltaje?

- Es la Fuerza que empuja a los electrones por el circuito eléctrico de un extremo a otro y se mide en Voltios.





# ***El Voltaje***

- Esta fuerza puede ser entregada por:

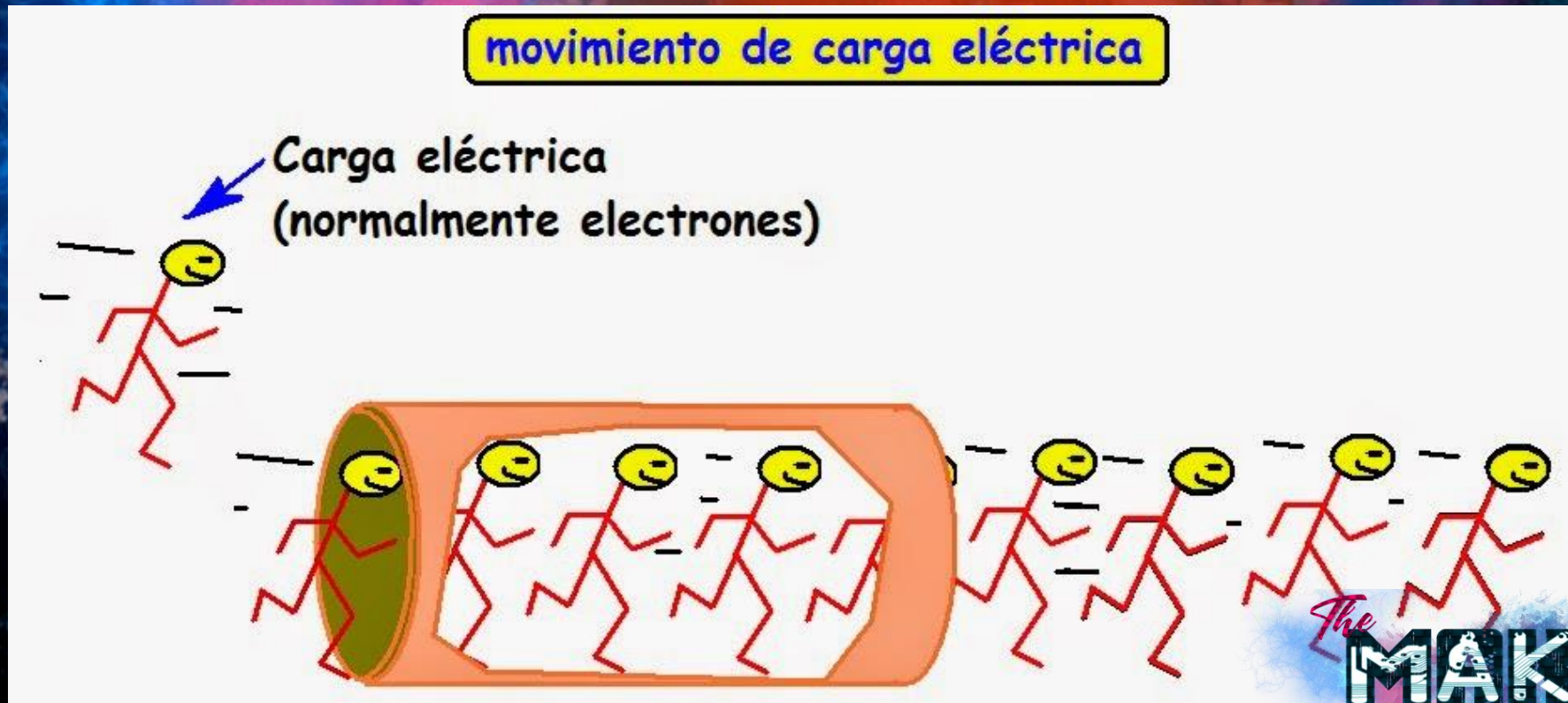




# Corriente

Qué es la Corriente?

- Es la cantidad de Electrones que circula por un circuito en un segundo y se mide en Amperios

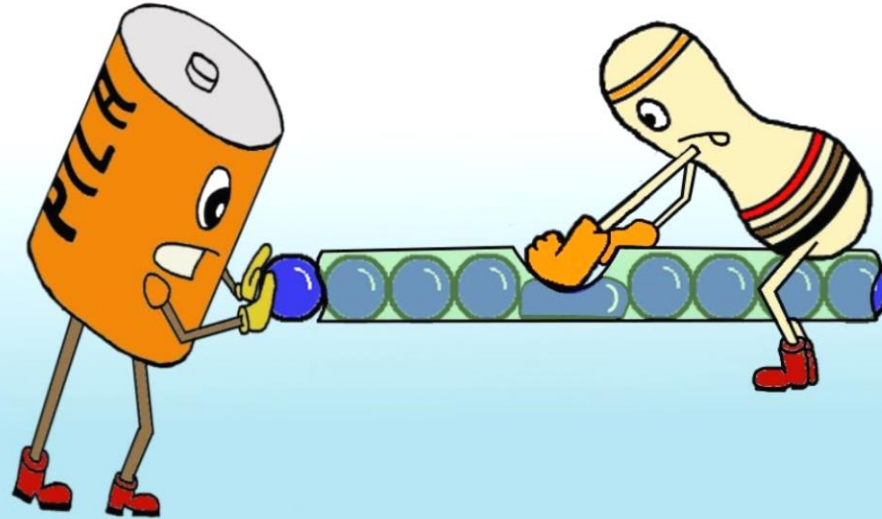




# ***Resistencia***

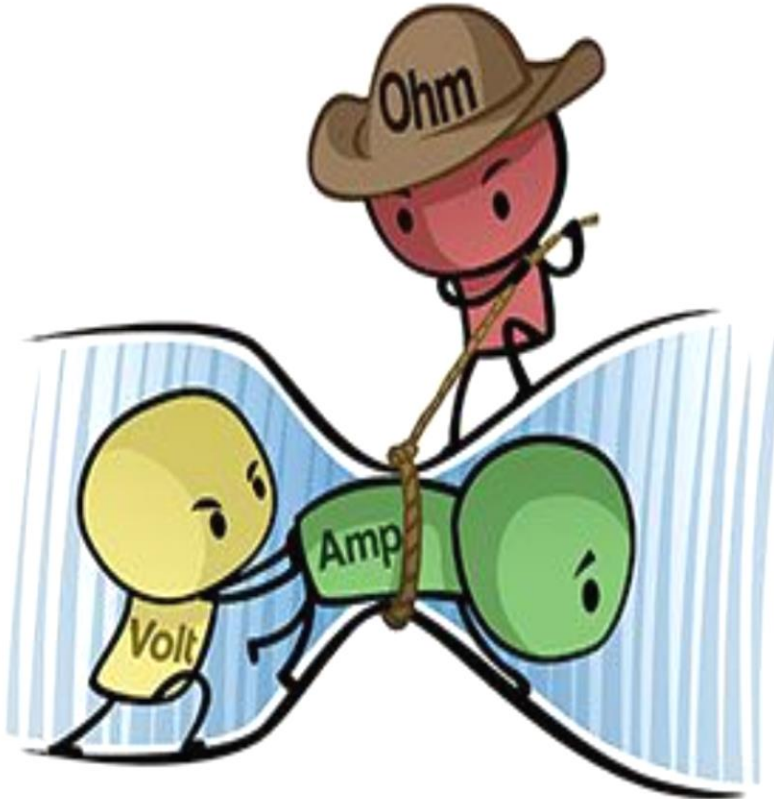
¿Qué es la Resistencia?

- Es aquella fuerza que se opone a la circulación de electrones en un circuito y se mide en Ohmios

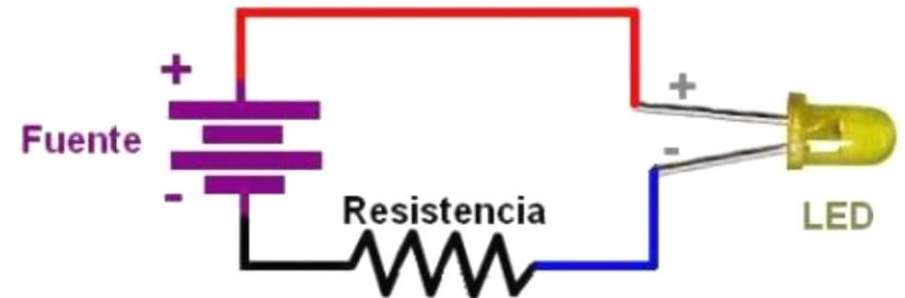




# Ley de Ohm



Como identificamos esto en nuestro circuito y como funciona?





# ***La Resistencia***

- Lo podemos encontrar en:

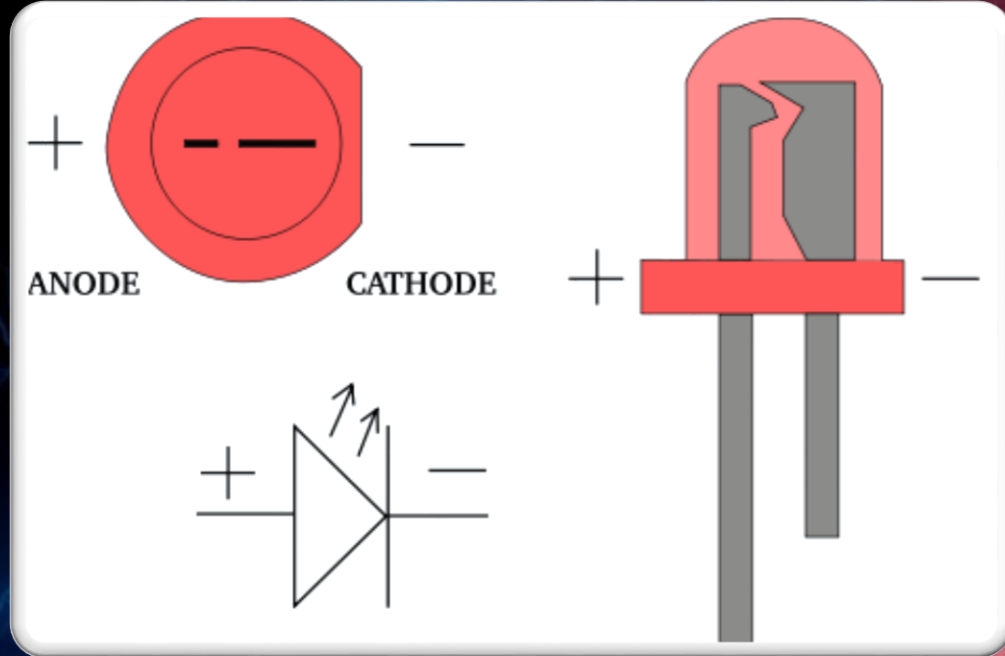




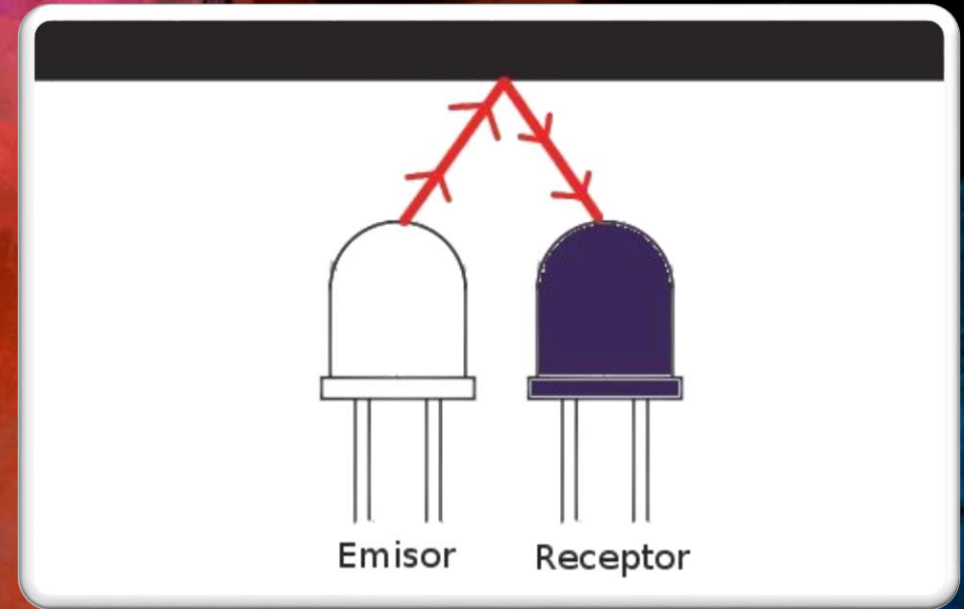
# *¿Qué es el Ánodo y Cátodo?*

- **ÁNODO.-** es el electrodo que pierde electrones en una reacción oxidación.
- Normalmente se vincula al polo positivo del transito de la corriente eléctrica.
- **CÁTODO.-** es el electrodo con carga negativa, que en la reacción química sufre una reacción de reducción.
- Normalmente se vincula al polo negativo del transito de la corriente eléctrica.





Un led es un componente optoelectrónico pasivo concretamente, un diodo que emite luz.

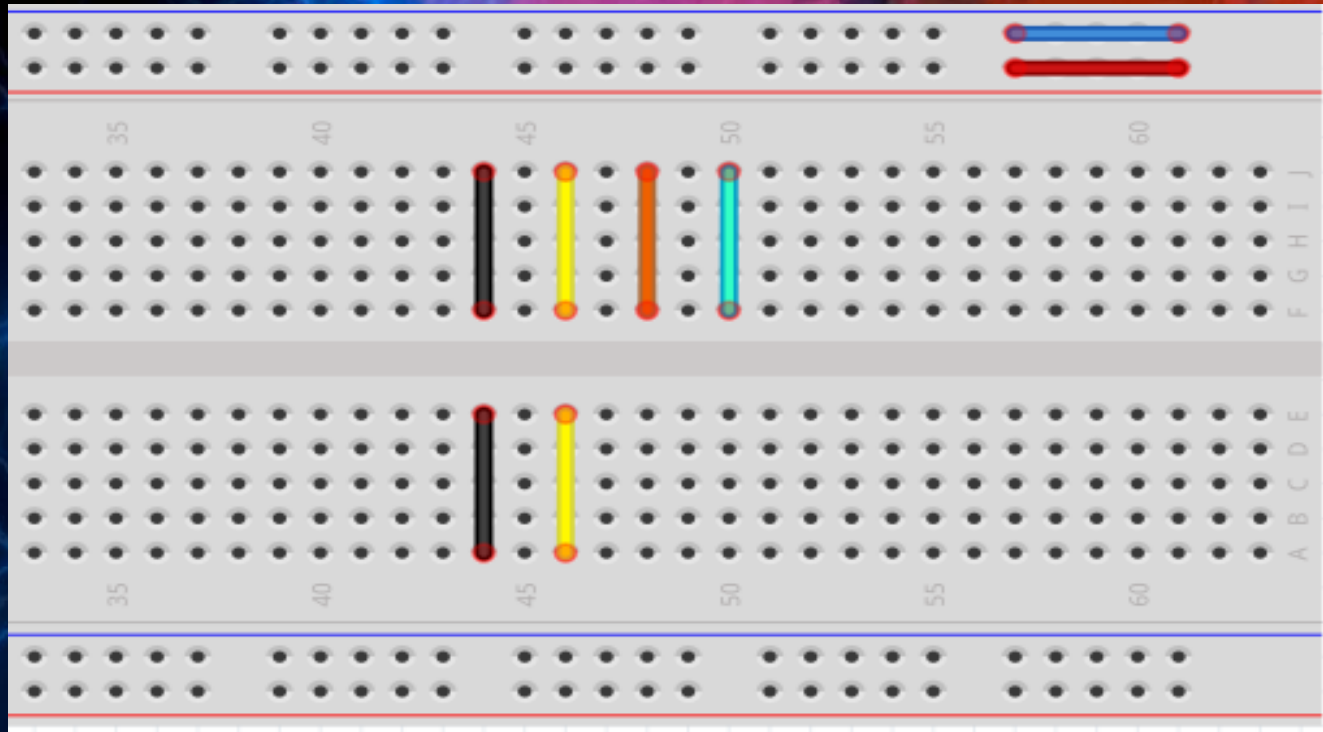


También tenemos LEDs infrarrojos emisores y receptores muy usados para hacer seguidores de línea o medidores de velocidad de giro de ruedas.



# *Armado Circuitos*

- Mi Protoboard



Veamos como esta conformado un protoboard.

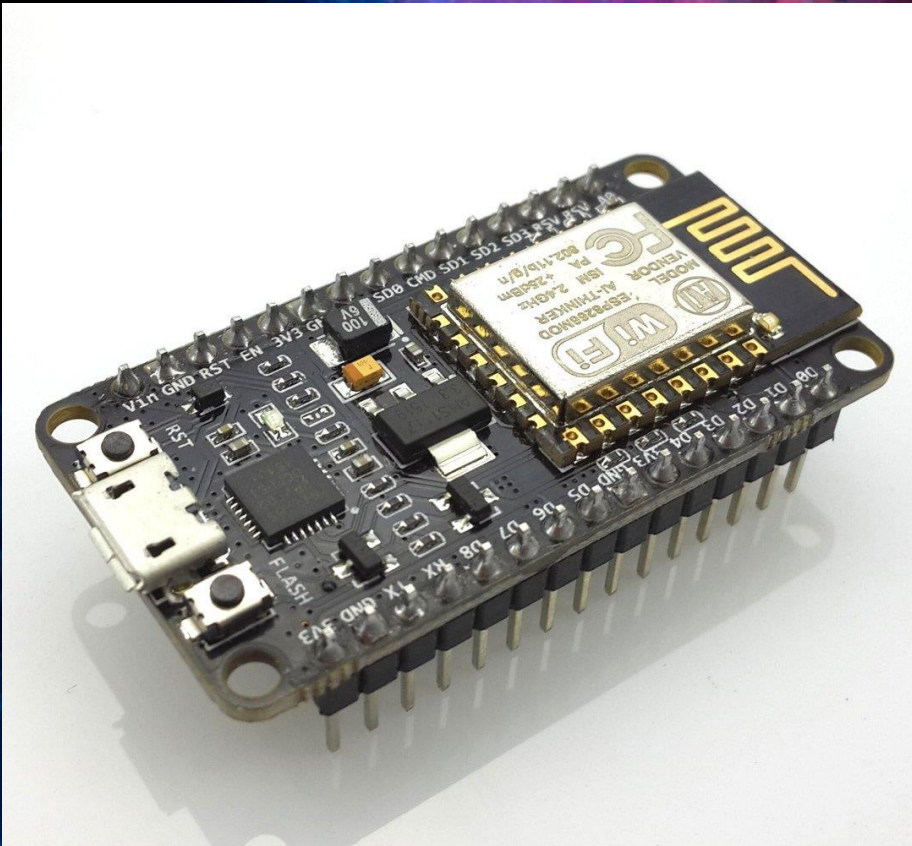


# *Terminología Común*

- 5v, 3.3v, Vcc, Xv, (+) nos indica el voltaje de salida de un determinado circuito o placa de control.
- GND, G, (-) nos indica el pin o conexión a tierra de un determinado circuito o placa de control.
- Vin, Vn (+) nos indica el pin de Entrada de Voltaje de una placa de control.
- mA, medida de consumo de corriente en mili amperios.
- mAh, medida de consumo de corriente en mili amperios por hora.



# ¿Qué es el Node Mcu?



**NodeMCU** fue una de las primeras placas de desarrollo con el microcontrolador **ESP8266**. Hasta entonces este chip solamente estaba disponible como placas ESP-xx como **ESP01** o **ESP12**.

**NodeMCU** se popularizó rápidamente porque permitía programar este microcontrolador de una manera mucho más sencilla que utilizando el **kit de desarrollo de Espressif**. Su diseño integra la electrónica necesaria. Para utilizarla no hace falta nada más que un **cable USB** y un **ordenador**. Este concepto enchufar y listo es el mismo que usan las placas de Arduino.



# Node Mcu



Cuando fue presentada, no existía la integración de ESP8266 con el entorno de Arduino. No utilizaba un lenguaje compilado sino uno interpretado llamado **LUA**.

El nombre de NodeMCU representa por tanto la unión de la placa de desarrollo junto con el firmware. Éste permite escribir código con el lenguaje de programación LUA.

Con la aparición de la integración con el IDE de Arduino, este firmware ha caído poco a poco en desuso. Aunque aún tiene sus incondicionales, ya no es una de las opciones más utilizadas para programar NodeMCU.



# ***Node Mcu***

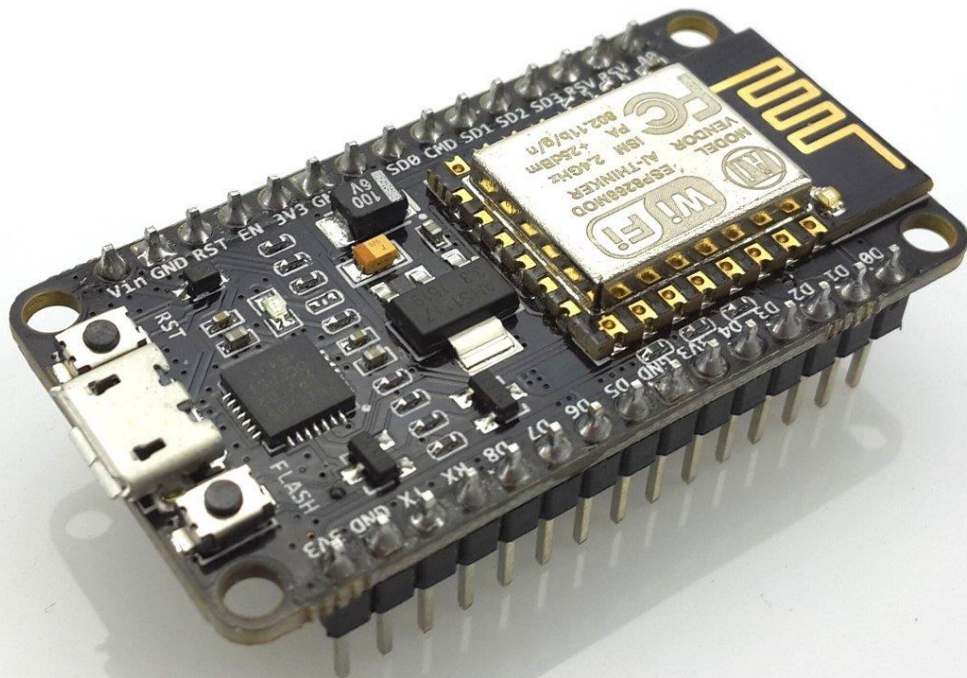
Como todas las placas que utilizan ESP8266, se les puede cargar cualquier firmware. Puede usarse desde el propio de NodeMCU con lenguaje LUA, a MicroPython.

NodeMCU es, desde sus inicios, un proyecto de código abierto. El código del firmware y el diseño del hardware están disponibles en GitHub.





# Node Mcu



- Voltaje de entrada (USB): 5V
- Voltaje de salida en los pines: 3.3V
- Voltaje de referencia en el ADC: 3.3V
- Corriente nominal por pin: 12mA
- Frecuencia de procesador: 80MHz (160MHz max.)
- 4MB Flash
- Consumo de corriente en stand-by @80MHz: 80mA



# Node Mcu

- Consumo de corriente al recibir una petición (librería WebServer en modo de punto de acceso) @ 80MHz: 90mA
- Consumo de corriente al utilizar HTTPClient.get() @ 80 MHz: 100-110mA
- Consumo de corriente en stand-by @160MHz: 90mA
- Consumo de corriente al recibir una petición (librería WebServer en modo de punto de acceso) @ 160MHz: 90-100mA
- Consumo de corriente al utilizar HTTPClient.get() @ 160 MHz: 100-110mA





# ***Lenguajes de programación***

Para los ejemplos de este curso, utilizaremos el lenguaje **C++/Processing** y el Arduino IDE. Como sabemos, el lenguaje Arduino es una plataforma con la que puedes compartir código entre diferentes placas y posee una gran cantidad de de librerías. Las otras plataformas (**Micropython y Lua**) tienen sus ventajas y desventajas. Una de sus grandes deficiencias que podemos mencionar es que no son estables (especialmente Lua) y que son lenguajes interpretados. Esto quiere decir que procesan el código un poco más lento que las contrapartes en **C++/Processing**.



# *¿Por qué Node Mcu?*

La tendencia actual a la programación de microcontroladores se esta moviendo hacia el área del Internet of Things (IoT). Esto quiere decir que nuestros dispositivos deben de ser capaces de conectarse para poder utilizar las últimas librerías y ambientes de trabajo. En este curso nos centraremos en una línea específica de dispositivos capaces de hacer esto: NodeMCU.

Los dispositivos utilizados comúnmente de la línea de Arduino no tienen conectividad a Internet. Sus últimos modelos han sido introducidos con este enfoque en mente, pero es difícil de vencer a un chip de tan **bajo costo y mejores velocidades de procesamiento** como es el ESP8266.



# ***Flexibilidad***

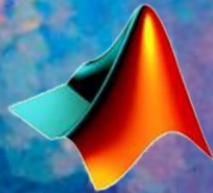
El NODE MCU es capaz de comunicarse con nuestra computadora por medio de una **conexión USB, Comunicación Serial y también mediante conexión WiFi**, esto permite interactuar con aplicaciones escritas en una amplia variedad de lenguajes de programación, tales como Visual Basic, C++, Java y cualquier otro lenguaje capaz de establecer comunicación con el Puerto Serie de la computadora, .





La integración de los proyectos con el node mcu y con lenguajes de programación de Alto Nivel hacen que dicha placa pueda ser considerado en proyectos a nivel científico y en ingeniería.

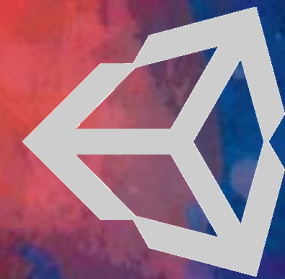
Además de ello nos facilita la comunicación con el internet de una forma mas sencilla y a un menor costo.



MATLAB



Visual Basic





# ***Blynk***



- Blynk es una plataforma compatible con aplicaciones iOS y Android para controlar Placas de desarrollo a través de Internet.
- Es un tablero digital donde puedes construir una interfaz gráfica para tu proyecto simplemente arrastrando y soltando widgets.
- Aplicación recomendada para diseñar proyectos orientados al internet de las cosas.



# *Cayenne*



- Cayenne es una aplicación para teléfonos inteligentes y computadoras que nos permite controlar una diversidad de placas de desarrollo que tengan una conexión a internet, los controla a través del uso de una elegante interfaz gráfica y un sólido protocolo de comunicación agradable.
- Aplicación recomendada para diseñar proyectos orientados al internet de las cosas.



# XAMPP



- XAMPP es un servidor web de plataforma, software libre, que consiste principalmente en el sistema de gestión de bases de datos MySQL, el servidor web Apache y los intérpretes para lenguajes de script PHP y Perl. El nombre es en realidad un acrónimo: X(para cualquiera de los diferentes sistemas operativos), Apache, MariaDB/MySQL, PHP, Perl.



# MySQL



- MySQL es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual: Licencia pública general/Licencia comercial por Oracle Corporation y está considerada como la base datos de código abierto más popular del mundo y una de las más populares en general junto a Oracle y Microsoft SQL Server, sobre todo para entornos de desarrollo web.



# PHP



- **PHP**, sigla recursiva en inglés de **PHP: Hypertext Preprocessor** (procesador de hipertexto), es un lenguaje de programación de propósito general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico. Fue uno de los primeros lenguajes de programación del lado del servidor que se podían incorporar directamente en el documento HTML en lugar de llamar a un archivo externo que procese los datos.



# *Asistente Virtual Jarvis*



- Asistente virtual que te permite interactuar con tu sistema operativo, realiza tareas simples como ejecutar programas y abrir paginas web especificas, también puede conectarse con tus redes sociales, todo por medio de comandos de voz.
- Nos permite interactuar con puertos de comunicación de manera unidireccional (solo envía ordenes). Esta característica nos permite controlar nuestros proyectos de domótica.



--->***PREGUNTAS***<---