

Carrera: Tecnicatura en Telecomunicaciones

Materia: Electrónica Microcontrolada

Docentes: Jorge E. Morales, C. Gonzalo Vera

Fecha: 29/07/22

Lugar: Córdoba, Córdoba, Argentina

Grupo: 10

Integrantes: Lot Brian Llanos Garcia; Alejandro Leyton; William Leyton; Esteban Dario Carrizo; Rafael Cabrera; Mario Alejandro Gonzalez.

SOC ESP8266

El **ESP8266** es un SoC de **bajo costo** y **bajo consumo** de energía fabricado por la empresa Espressif, Se trata de un SoC o Sistema en Chip. Básicamente consiste en un chip que tiene todo integrado (o casi todo) para que pueda funcionar de forma autónoma como si fuera un ordenador incluyendo conexión WiFi y compatible con el protocolo TCP/IP. El ESP8266 lo único que no tiene es una memoria para almacenar los programas. Esto supone un inconveniente ya que parte de los pines de entrada y salida, tendrán que ser utilizados para conectarse a una memoria Flash externa.

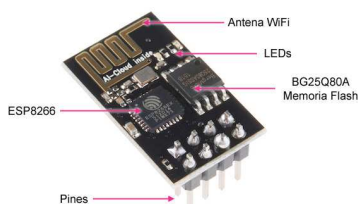
▪ Especificaciones del chip ESP8266

- ❖ Utiliza una CPU Tensilica L106 32-bit
- ❖ Voltaje de Operación entre 3V y 3,6V
- ❖ Según noticias del propio CEO de Espressif, admite tensiones de 5V en los pines GPIO
- ❖ Corriente de operación 80 mA
- ❖ Temperatura de -40°C y 125°C
- ❖ Soporta IPV4 y los protocolos TCP/UDP/HTTP/FTP
- ❖ No soporta HTTPS, pero puede hacerlo mediante software tanto en cliente como en servidor TLS1.2. La primera implementación está en desarrollo.
- ❖ Tiene 17 pines GPIO pero solo se pueden usar 9 o 10. El GPIO16 es especial ya que está conectado al RTC (Real Time Clock).
- ❖ Pueden ser configurados con resistencia Pull-up o Pull-down
- ❖ Soporta los principales buses de comunicación (SPI, I2C, UART)

Modelos de Placas

En el mercado hay varios tipos de modelos de ESP8266 tales como:

ESP-01



Tiene disponible dos pines GPIO digitales para controlar sensores y actuadores. También se puede llegar a utilizar para este uso los pines Rx y Tx si no se utilizan para la comunicación a través del puerto serie. Se puede programar a través de un adaptador serie/USB o con el cableado adecuado, a través de Arduino. Los conectores que vienen por defecto, no permiten conectarlo a la protoboard.

ESP-05



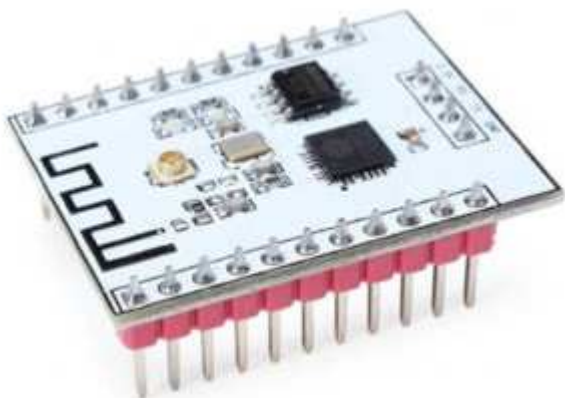
Quizás sea el módulo más simple de toda la gama. Está destinado a ser un Shield WiFi para Arduino. La disposición de los pines nos permiten un fácil conexionado con la protoboard. Por el contrario no dispone de ningún puerto GPIO accesible.

ESP-12



Permite hacer bastantes más cosas que los módulos anteriores. Lo puedes conseguir por un **precio muy reducido**. Tenemos acceso a 11 puertos GPIO de los cuales uno, es analógico con una resolución de 10-bit (1024 valores posibles). La configuración en modo dormido es muy sencilla. Esto nos permitirá ahorrar mucha energía. Por el contrario, la conexión con la protoboard no es muy amigable. Necesitamos soldar los pines o comprar un adaptador, aunque también hay que soldar. Quizás esta sea la mejor opción si queremos hacerlo funcionar de forma autónoma.

ESP-201



En principio solo podemos acceder a 11 puertos GPIO pero tras unas modificaciones, podríamos acceder a un par más de ellos. Lo podemos encajar fácilmente en una protoboard y permite el acople de una antena externa para tener más alcance.

Wemos D1 mini



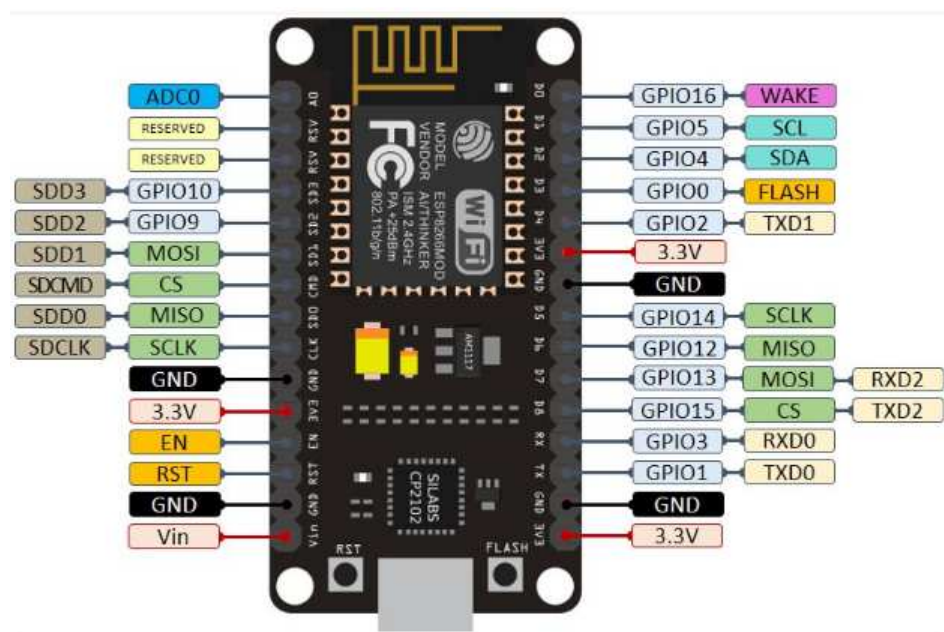
Wemos D1 mini ESP8266 es una plataforma de desarrollo similar a **Arduino** especialmente orientada al Internet de las cosas (IoT). La placa Wemos D1 Mini ESP8266 tiene como núcleo al **SoM ESP-12E** que a su vez está basado en el **SoC Wi-Fi ESP8266**, integra además el convertidor USB-Serial TTL CH340G y conector micro-USB necesario para la programación y comunicación a PC. Wemos D1 mini está diseñado especialmente para trabajar montado en protoboard o soldado sobre una placa. Posee un regulador de voltaje de 3.3V en placa, esto permite alimentar la placa directamente del puerto micro-USB o por los pines 5V y GND. Los pines de entradas/salidas (GPIO) trabajan a 3.3V por lo que para conexión a sistemas de 5V es necesario utilizar convertidores de nivel como: Convertidor de nivel 3.3-5V 4CH o Convertidor de nivel bidireccional 8CH - TXS0108E.

NodeMCU



El **NodeMCU** es el módulo más característico de este tipo. Su **precio ronda los \$1500 pesos**. A diferencia de los otros módulos, viene con todo lo necesario para empezar a trabajar de forma autónoma. Incluye un adaptador serie/USB y se alimenta a través del microusb. Está basado en el ESP-12 y la última versión oficial es la 2. Lo más interesante de este módulo es que puedes descargar un **firmware** que te permite programar en lenguajes como LUA, Python, Basic o JavaScript. Sin duda alguna este módulo es la mejor opción si queremos adentrarnos en el mundo del ESP8266.

Distribución de Pines



Label	GPIO	Input	Output	Notes
D0	GPIO16	sin interrupt	ni PWM ni I2C	ALTO at boot usado para activar desde suspensión profunda
D1	GPIO5	OK	OK	usado como SCL (I2C)
D2	GPIO4	OK	OK	usado como SDA (I2C)
D3	GPIO0	resistor a ALTO	OK	conectado al botón FLASH, falla boot si se pone BAJO
D4	GPIO2	resistor a ALTO	OK	ALTO en boot conectado al LED, falla boot si se pone BAJO
D5	GPIO14	OK	OK	SPI (SCLK)
D6	GPIO12	OK	OK	SPI (MISO)
D7	GPIO13	OK	OK	SPI (MOSI)
D8	GPIO15	resistor a GND	OK	SPI (CS) falla boot si se pone ALTO
RX	GPIO3	OK	RX pin	ALTO at boot
TX	GPIO1	pin TX	OK	ALTO en el boot salida debug en boot, falla boot si se pone a BAJO
A0	ADC0	Entrada Analógica	X	

Pines utilizados durante el arranque

Puede ocurrir que el ESP8266 no arranque si algunos pines se colocan en BAJO o ALTO. La siguiente lista muestra el estado de los pines en BOOT:

GPIO16: el pin está alto en BOOT

GPIO0: error de arranque si pone en BAJO

GPIO2: el pin está alto en BOOT, falla el arranque si se pone a BAJO

GPIO15: error de arranque si se pone a ALTO

GPIO3: el pin está alto en BOOT

GPIO1: el pin está alto en el ARRANQUE, el arranque falla si se pone BAJO

GPIO10: el pin está alto en BOOT

GPIO9: el pin está alto en BOOT

GPIO6 a GPIO11 se conectan generalmente a un chip de memoria flash en las placas ESP8266. Por lo tanto, no se recomienda el uso de estos pines.

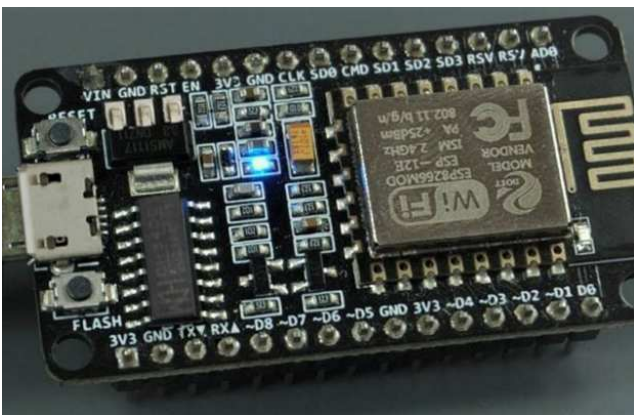
Entrada analógica

El ESP8266 solo admite lectura analógica en un GPIO. Ese GPIO se llama **ADC0** y suele estar marcado en la serigrafía como **A0**.

El voltaje de entrada máximo del pin ADC0 es de 0 a 1 V si se usa el chip básico ESP8266. Usando una placa de desarrollo como el kit ESP8266 12-E NodeMCU, el rango de entrada de voltaje es de 0 a 3,3 V, porque estas placas contienen un divisor de voltaje interno.

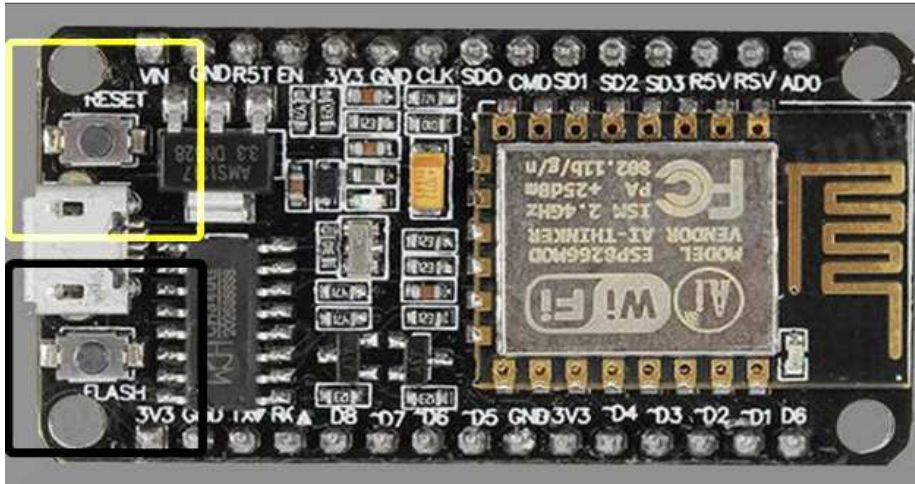
LED integrado

La mayoría de las placas de desarrollo ESP8266 tienen un LED integrado. Este LED suele estar conectado a GPIO2.



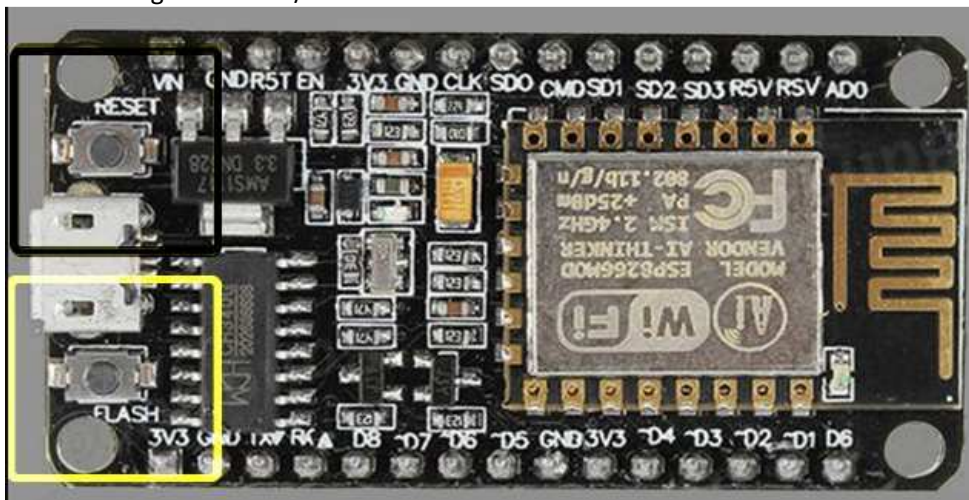
Pin RST

Cuando el pin RST se pone a BAJO, el ESP8266 se reinicia. Esto es lo mismo que presionar el botón integrado RESET.



GPIO0

Cuando GPIO0 se pone a BAJO se configura el ESP8266 en modo de carga en el arranque. Esto es lo mismo que presionar el botón integrado FLASH/BOOT.



IDE y Lenguajes

● Arduino (C++):

Para programar un ESP8266 desde Arduino hay que agregar la URL de las placas ESP8266 para poder descargar el núcleo (o core) de ESP8266 para Arduino. URL:

https://arduino.esp8266.com/stable/package_esp8266com_index.json

Una vez hecho eso. Para instalar el soporte para ESP8266 y las placas de desarrollo hay que ir a “Herramientas>Placas>Gestor de Tarjetas”.

Esto abrirá el gestor de placas o tarjetas. En cuanto se inicie, comenzará a actualizar su base de datos, utilizando las URLs que se agregaron anteriormente en preferencias. Una vez termine, hay que escribir “ESP8266” en la barra de búsqueda para filtrar las placas disponibles y seleccionar la placa a utilizar.

● Thonny (micro Python)

Thonny viene con Python 3.7 incorporado, por lo que solo se necesita un instalador simple y está listo para aprender a programar. (También puede usar una instalación separada de Python, si es necesario). La interfaz de usuario inicial está desprovista de todas las funciones que pueden distraer a los principiantes.

Dentro de la página sugerida para buscar información, encontramos lo siguiente: “Experimentamos con varios IDE para programar las placas ESP32 y ESP8266 usando Micro Python, y Thonny parecía una buena opción.

Aunque hay algunos errores, se actualiza y mejora constantemente. Te permite programar tus placas ESP32 y ESP8266 con Micro Python, y es compatible con Windows, Mac OS X y

Linux. Incluso viene instalado por defecto en el sistema operativo Raspberry Pi. Además, es fácil de instalar, por lo que no deberías tener problemas con el proceso de instalación.”

● VSC (C++, Micro Python)

Visual Studio Code (VSCode) es un editor de código sumamente versátil y potente que puede emplearse para programar en una multitud de lenguajes sobre distintas plataformas. Es altamente configurable y sus capacidades se pueden ampliar agregando distintas extensiones. Es muy recomendable para programar nuestras placas ESP32 (y ESP8266). Con VS Code podemos escribir código en una amplia variedad de lenguajes aprovechando distintas funcionalidades como coloreo de sintaxis, autocompletado, depuración y optimización de código. También integra soporte para GIT, lo que nos permite llevar de manera ordenada las distintas versiones de los proyectos en los que trabajamos.