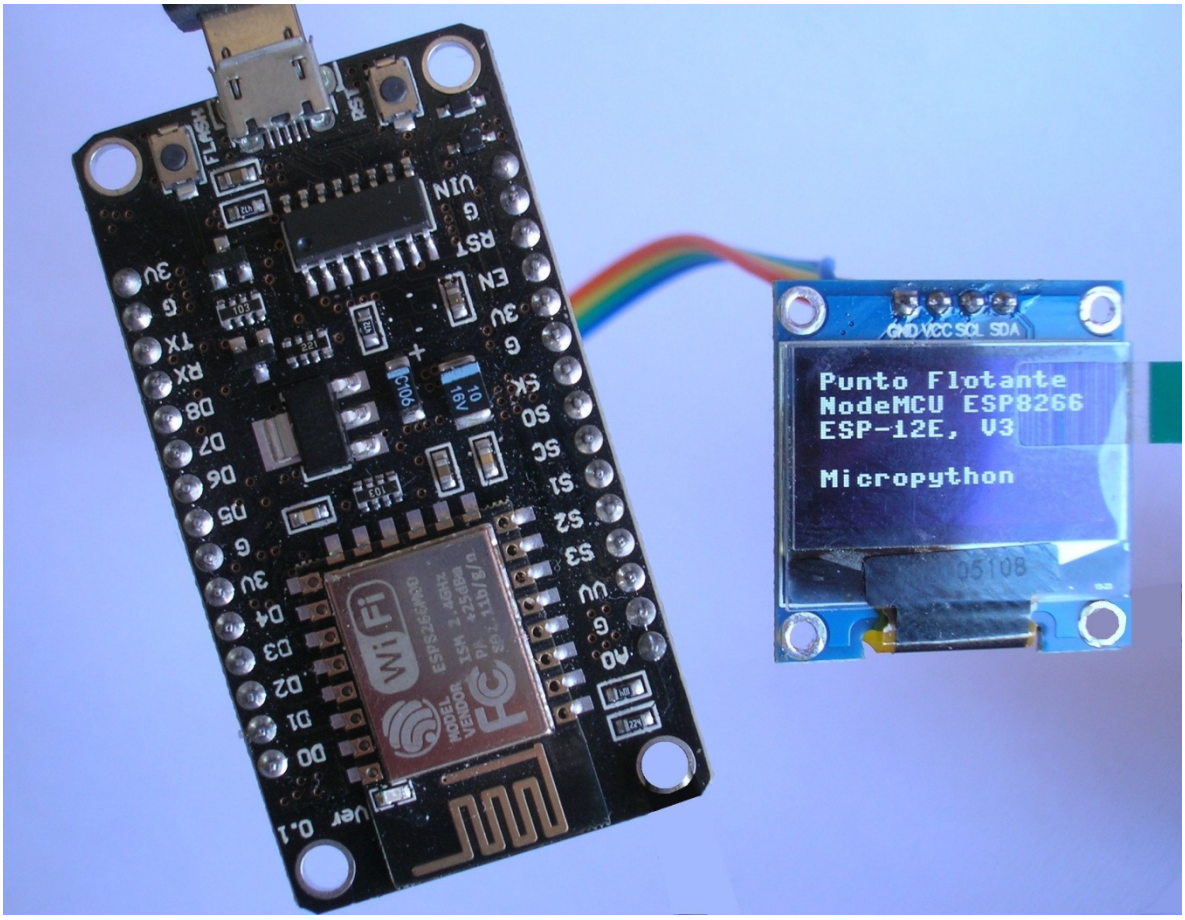
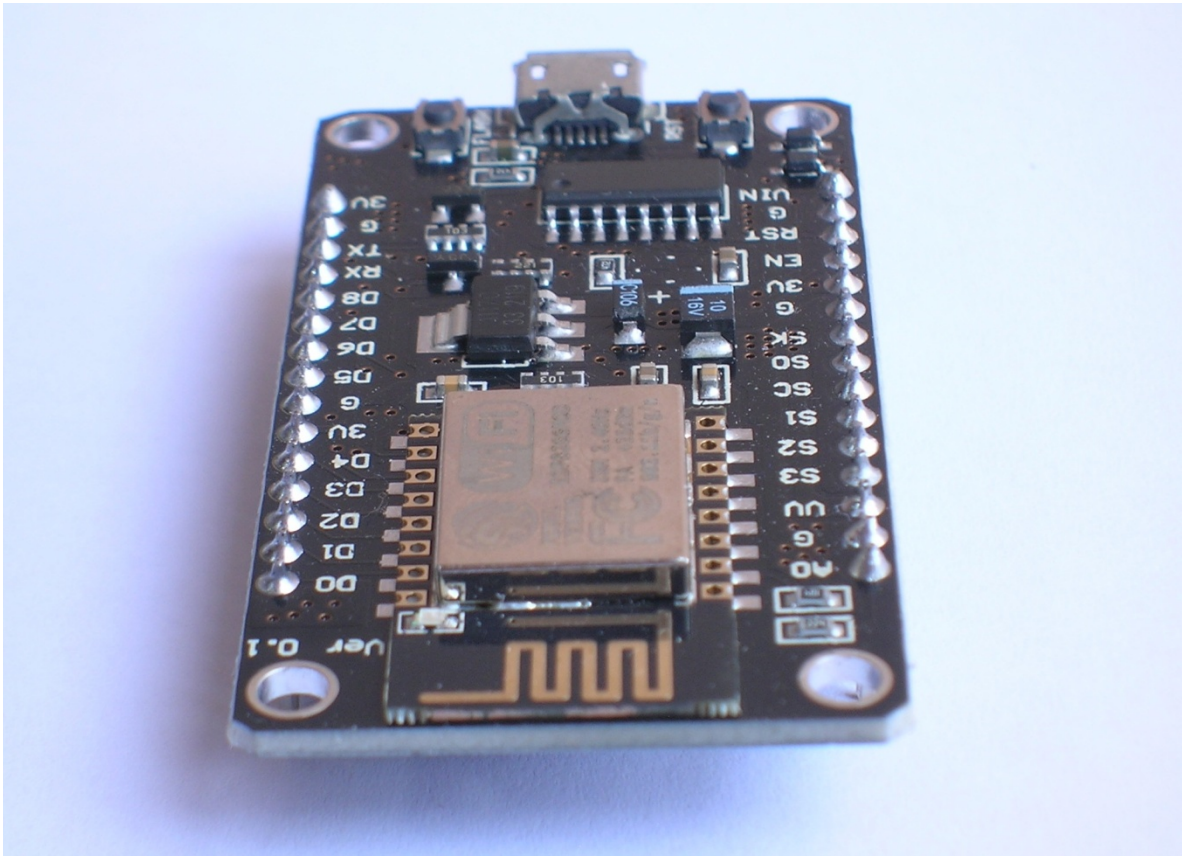


Manual básico de pruebas del NodeMCU ESP8266, versión ESP-12E. Firmware Micropython.

Contenido:

1. Introducción al NodeMCU
2. Conexión de componentes.
3. Carpeta de trabajo con software y firmware.
4. Carga del driver CH340G USB-Serial
5. Tener listo el IDE *uPyCraft*
6. Carga del firmware *Micropython* en NodeMCU
7. Carga del programa de aplicación (flashear un led).

1. Introducción al NodeMCU ESP8266, versión ESP-12E:



Un dispositivo microcontrolador nuevo ha ingresado al mundo de la electrónica, bajando los costos de sistemas que antes eran complejos en su desarrollo, como el llamado IoT (Internet of Things) y la automatización de casas (Home Automation). Se trata del nuevo NodeMCU ESP8266 en su versión ESP-12E V3.

El NodeMCU ESP-12E, V3, es una plataforma completa de desarrollo, módulo SoC (system on chip), basado en el microcontrolador Tensílica L106 de 32 bits, funcionando a 80 Mhz, con una memoria Flash de 4MB, y 32 KB de SRAM. Tiene conectividad Wifi, estándar 802.11 b/g/n, para la implementación de dispositivos operando como servidores de internet (web server).

NodeMCU incorpora además un circuito CH340G, con lo cual es posible comunicarse durante la fase de desarrollo con el puerto serial del microcontrolador desde el puerto USB de la PC y usando un cable Micro USB.

El NodeMCU integra el protocolo TCP-IP y tiene conectividad Wifi, con antena de radiofrecuencia integrada, lo cual permite su conexión a internet a través de un router cercano al módulo (digamos hasta unos 3 metros), con la posibilidad de control remoto del tipo "Anywhere", desde cualquier parte del mundo. Esto significa que, ya sea a través de un teléfono celular ó una computadora conectados a internet, se pueden enviar comandos al NodeMCU para activar dispositivos y recibir información de sensores ó alarmas instaladas en casas y oficinas.

Durante la fase de desarrollo, NodeMCU se conecta directamente a la computadora con un cable micro USB, el cual proporciona la alimentación de 5v al módulo y permite la carga de nuevos programas a una velocidad típica de 115200 bps.

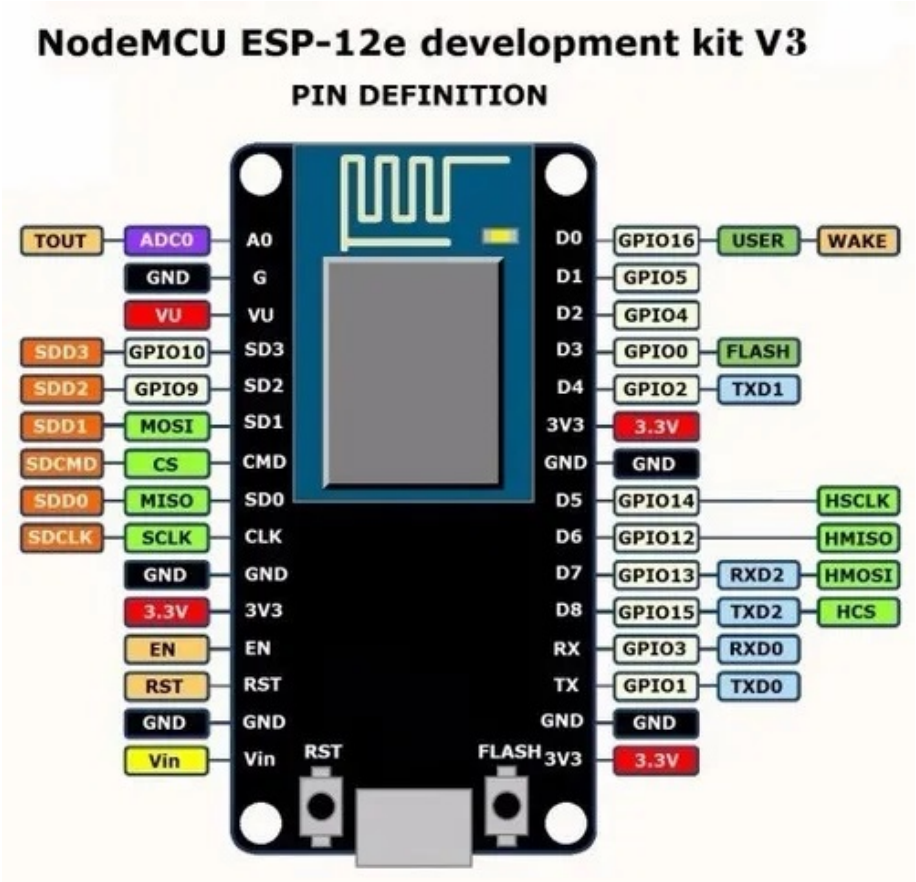
NodeMCU cuenta con 13 pines de entrada/salida, llamados GPIO (General Port Input Output), y puede funcionar de manera totalmente autónoma como servidor web y recibir comandos remotos para operar los pines mencionados, ya sea para activar dispositivos ó enviar status de sensores digitales. Incluye también pines con las funciones PWM, I2C, SPI, UART, y un convertidor A/D de 10 bits.

Puede alimentarse directamente a través del conector micro USB ó con una fuente externa. Su voltaje de alimentación es de 5v, el cual es reducido a 3.3v por medio de un regulador on board. El consumo promedio es de 100 ma. La corriente aumenta durante la trasmisión y recepción Wifi, de tal manera que, si se alimenta con fuente externa, se recomienda una capacidad de 300 ma.

Los lenguajes de programación más usados para NodeMCU, son MicroPython, Lua y Arduino. Todos son de código abierto y cuentan con herramientas de desarrollo gratuitas.

En este manual, haremos uso del lenguaje Micropython, por ofrecer la mayor disponibilidad y sencillez de uso. Micropython cuenta con un firmware para el NodeMcu que es un lenguaje interpretado, es decir, no requiere compilar los programas, sino solamente escribirlos en texto y enviarlos a la NodeMCU.

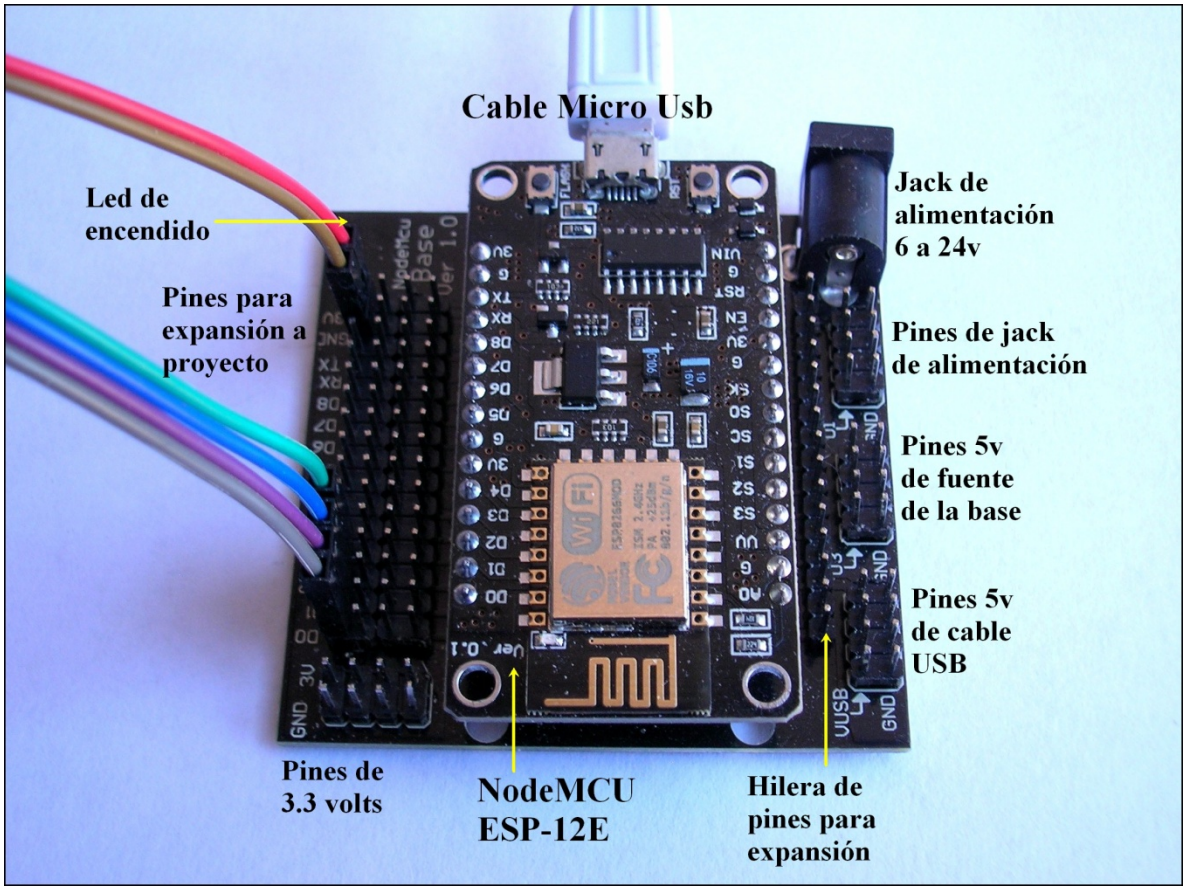
Especificaciones técnicas:
- Modelo: ESP8266, versión ESP-12E, V3
- Función: plataforma de conectividad Wifi, TCP-IP, IoT.
- Procesador: Tensílica L106 de 32 bits, bajo consumo.
- Memoria: 4 MB Flash, 32 KB SRAM
- Radiofrecuencia: WiFi, 802.11 b/g/n, 2.4 Ghz, antena integrada.
- Protocolo: TCP/IP integrado, conectividad a internet.
- Comunicación con computadora: a través de interfaz USB-Serial CH340G
- Velocidad de comunicación por USB: 9600 hasta 961,600 bps
- Alimentación: 5v, a través del puerto micro USB ó con fuente externa.
- Corriente máxima consumida: 250 ma.
- 13 pines GPIO disponibles para entrada/salida.
- Pines con funciones de PWM, 1wire, I2C, SPI, UART, ADC.
- Dimensiones: 5.7 x 3.3 x 1.5 cms
- Distancia entre pines: estándar 2.54 mm



2. Conexión de componentes:

Para la realización de la instalación del software, firmware y las primeras pruebas al módulo, usted necesita:

- 1 módulo NodeMCU versión ESP-12E
- 1 cable Micro USB, a través del cual se le da la alimentación de 5v al NodeMCU y al mismo tiempo puede tener comunicación con la PC
- En la foto de abajo se incluye una base, la cual no es indispensable para las pruebas de este manual.



Inserte el cable Micro USB al NodeMCU, con lo cual alimentará el circuito. Deben encenderse el led rojo (si cuenta con la base) y brevemente el led azul del módulo (junto a la antena de WiFi). Por el momento no utilizaremos los cables de expansión mostrados en la foto.

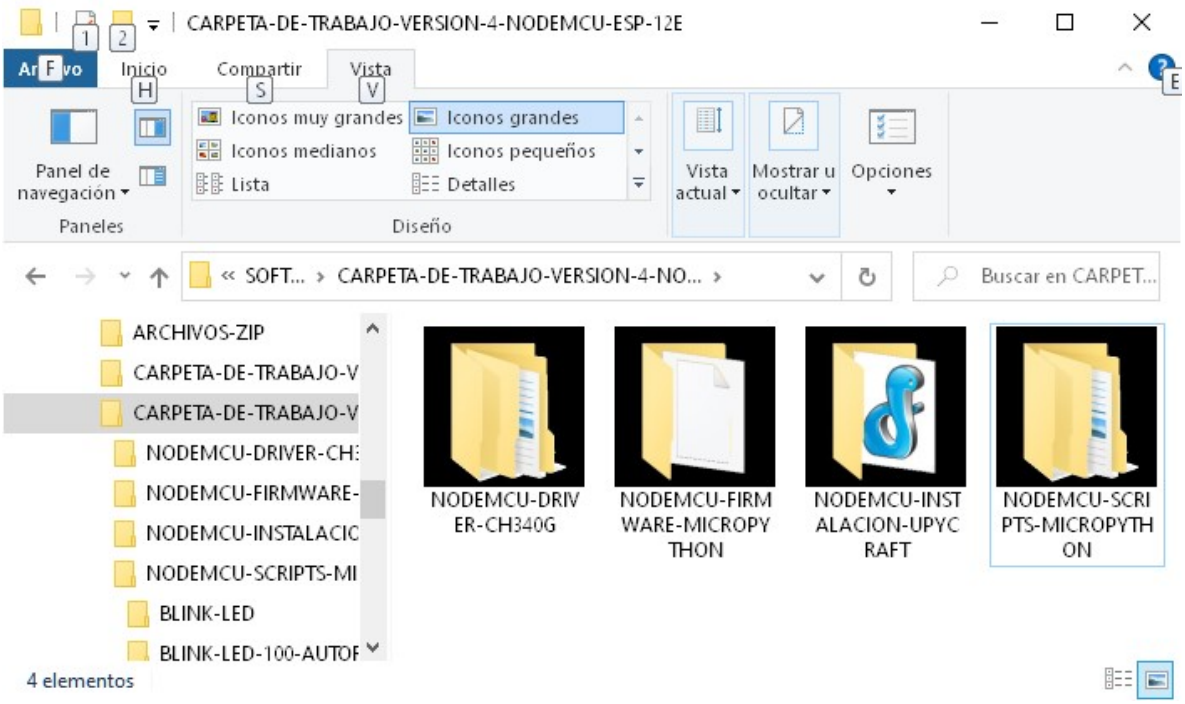
3. Carpeta de trabajo con software y firmware:

Baje las siguientes carpetas .zip de internet a su PC (uPyCraft es un archivo .exe):

Carpeta de trabajo NodeMCU ESP8266, ESP12E:

https://puntoflotante.net/NODEMCU-FIRMWARE-MICROPYTHON.zip
https://randomnerdtutorials.com/uPyCraftWindows
https://puntoflotante.net/NODE-MCU-SCRIPTS-MICROPYTHON.zip
https://puntoflotante.net/CH341SER.ZIP

Descomprima las carpeta para almacenar los archivos en su PC como se muestra:

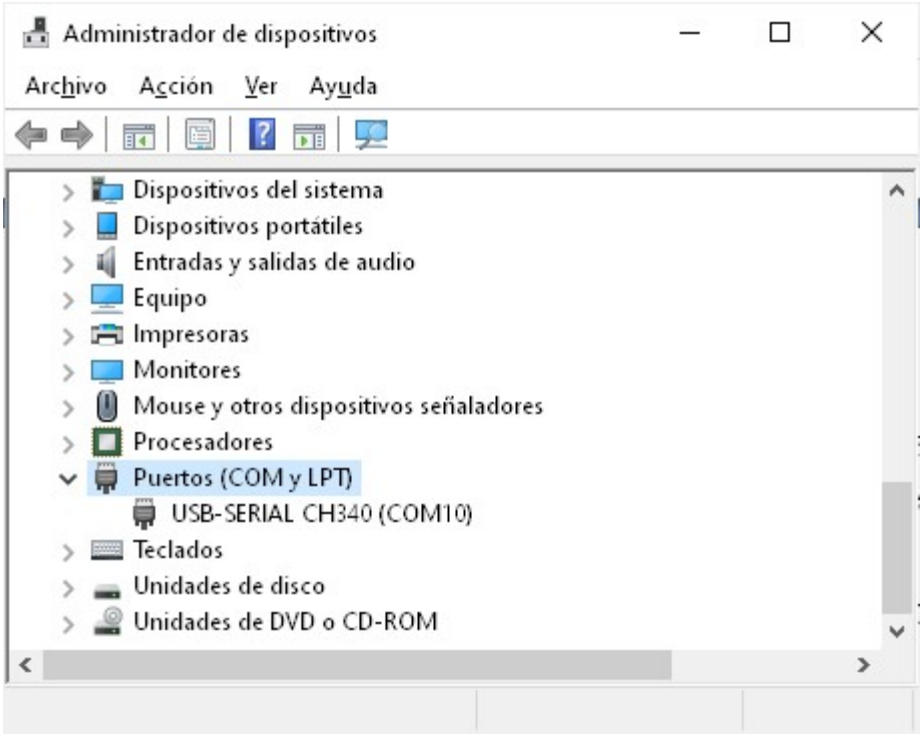


4. Dar de alta el driver USB-Serial CH340G.

El módulo NodeMCU, versión ESP-12E tiene integrado un circuito convertidor de USB a Serial. Es necesario cargar el driver CH340G en la computadora PC de tal manera que sea reconocido como un puerto serial COM por el administrador de dispositivos.

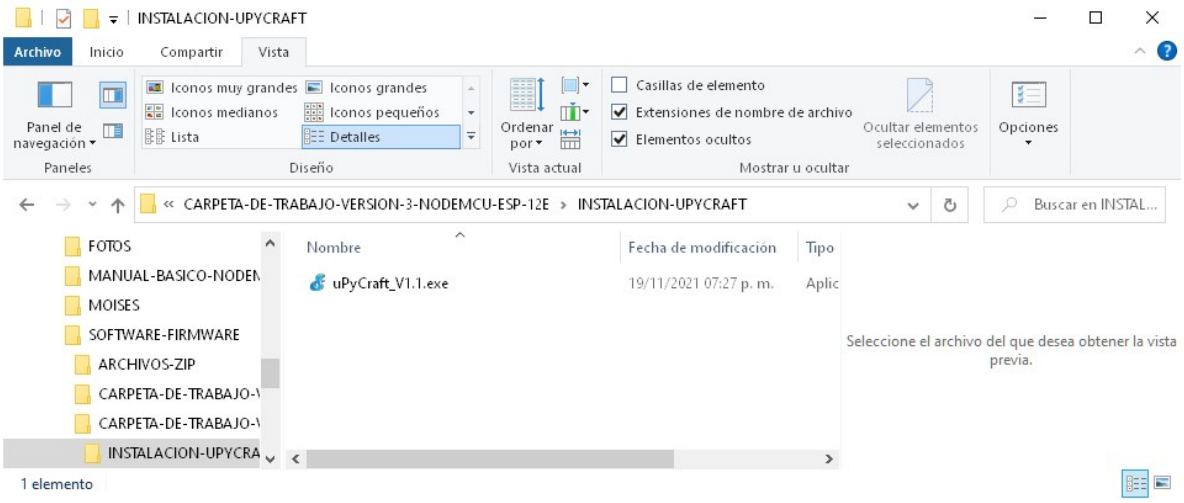
Inserte el cable Micro USB a su computadora, en cualquier puerto USB. Abra la carpeta del driver y ejecute el archivo “SETUP.EXE”.

Abra el administrador de dispositivos y deberá mostrar la ventana con al puerto serial COM asignado. En este caso se trata del COM10. En su computadora podría asignarse otro número del puerto.



5. Preparar el IDE *uPyCraft.*

El software IDE (Integrated Development Environment) más utilizado para desarrollo y prueba de nuevos programas de Micropython para el NodeMCU es el llamado *uPyCraft.* En la carpeta de trabajo (capítulo 3) encontrará la subcarpeta llamada “INSTALACION-UPYCRAFT”, con el archivo mostrados en la figura.



Recomendamos que haga un acceso directo del programa uPyCraft.exe (no requiere instalación) a su escritorio para tenerlo disponible.

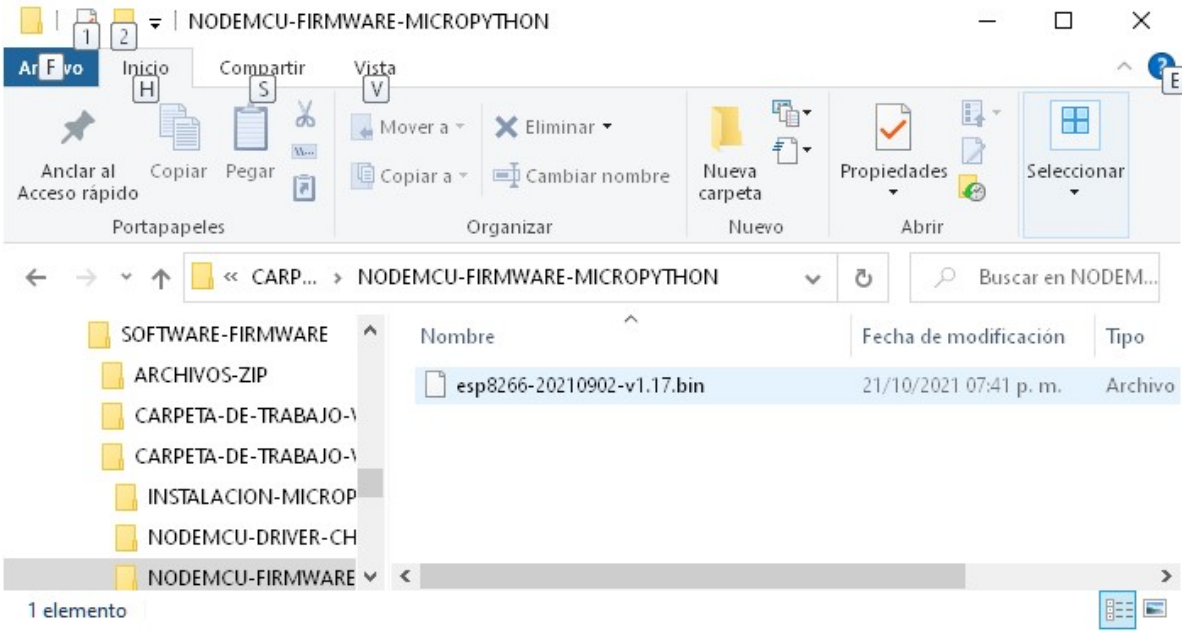
6. Carga del firmware Micropython en NodeMCU.

Micropython no es un lenguaje que necesite un compilador. Lo anterior significa que el firmware, una vez cargado en el microcontrolador, traduce cada línea del programa de alto nivel y la ejecuta una después de otra. A este tipo de lenguajes se les denomina “Intérpretes”. Un programa en Micropython puede ejecutarse línea por línea, haciendo el trabajo de debugging más simple que en los lenguajes compilados.

Para realizar la carga del firmware de Micropython, procederemos como sigue:

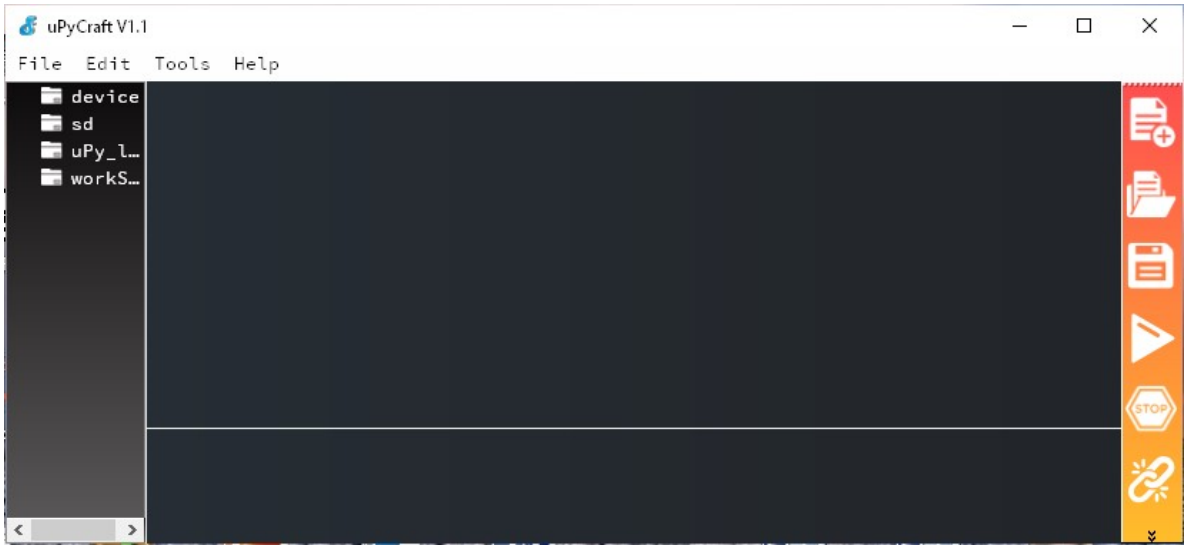
En su carpeta de trabajo (Capítulo 3) hay una subcarpeta llamada “NODEMCU-FIRMWARE-MICROPYTHON”.

En esta subcarpeta encontrará el archivo de firmware, como se muestra en la siguiente imagen:

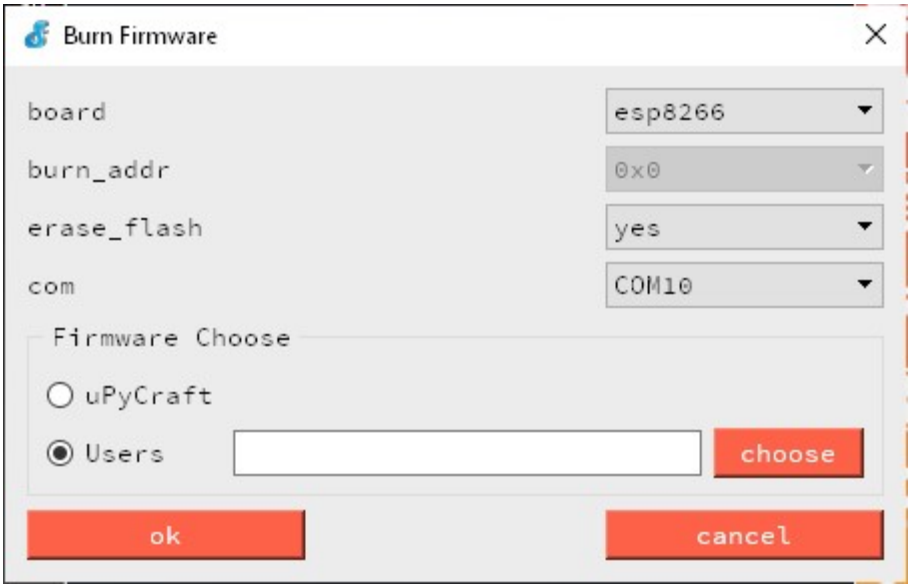


Es importante aclarar que este archivo es la última versión con fecha 2021-09-02, lo que significa que cuando usted realice estas pruebas, puede haber otra versión más reciente disponible. En cualquier caso, ésta versión le funcionará para los fines de este manual.

Con el sistema NodeMCU conectado a su puerto USB, abra el programa IDE *uPyCraft* y observará la siguiente ventana:

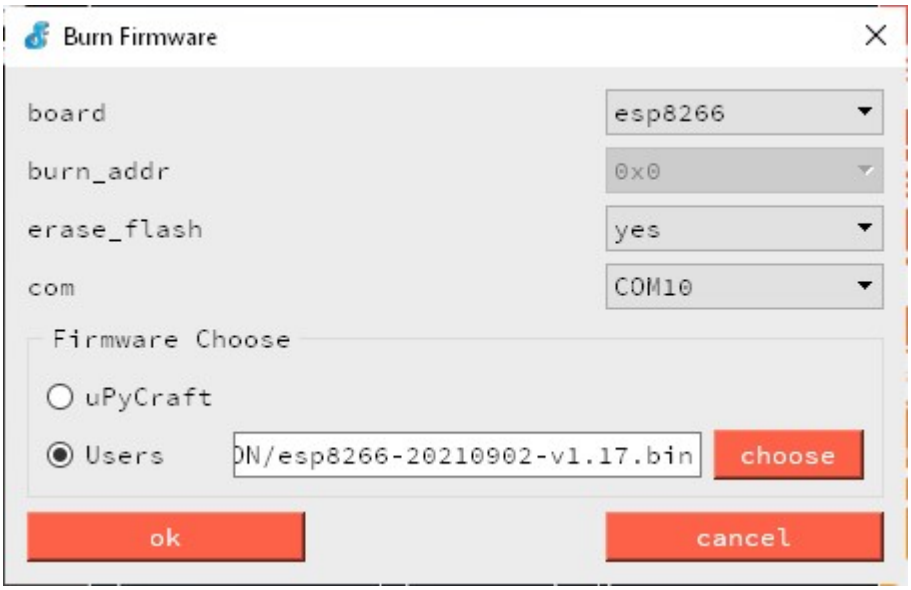


Elija la opción “Tools” y luego “BurnFirmware”. El programa responderá con la siguiente ventana:



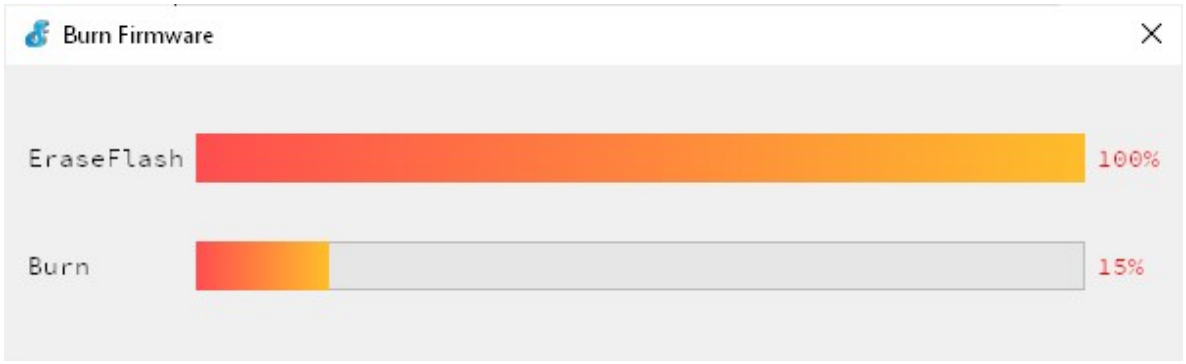
Configure la ventana como se muestra. Recuerde que en nuestro caso se dio de alta el COM10. Usted debe elegir el puerto serial dado de alta en el administrador de dispositivos de su computadora (Capítulo 4). Luego dé click en "Choose" para elegir el archivo de firmware mencionado al principio del capítulo.

Ahora la ventana contiene el path del archivo elegido, como se muestra en la imagen.

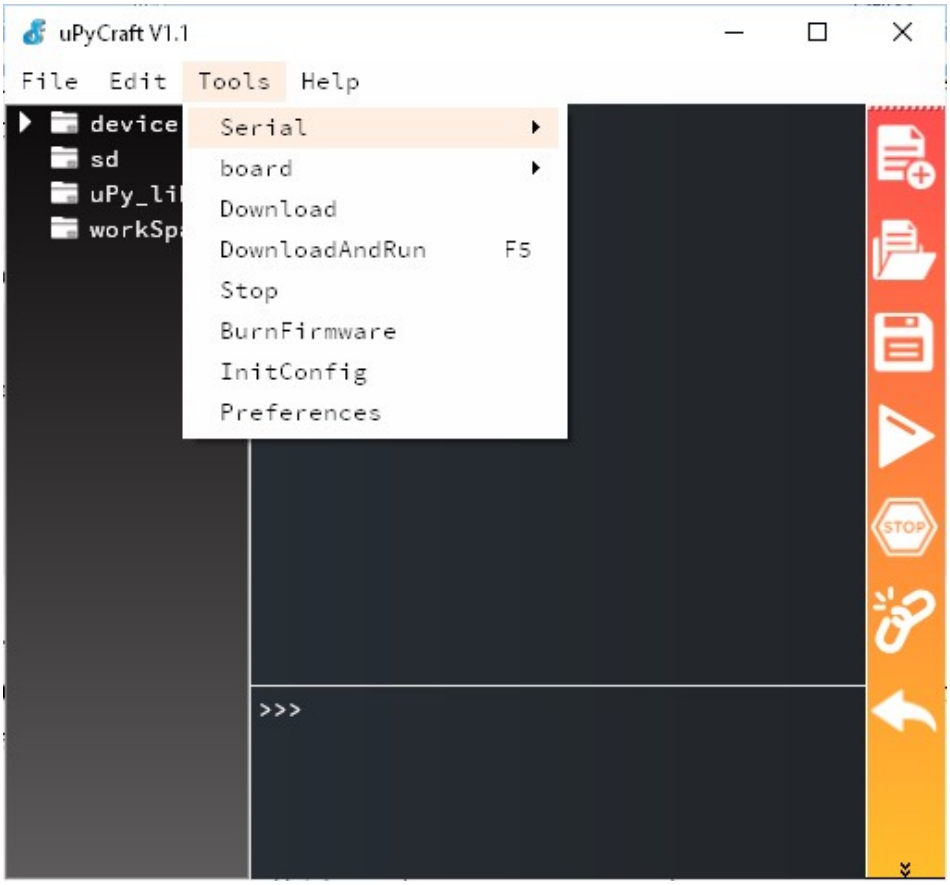


Al dar click en "OK", iniciará la transferencia del firmware de la computadora al NodeMCU. Observe cómo primeramente se borra totalmente la memoria Flash del

NodeMCU, antes de grabar el nuevo firmware. El led azul en el módulo estará parpadeando, indicando que la grabación está en proceso.



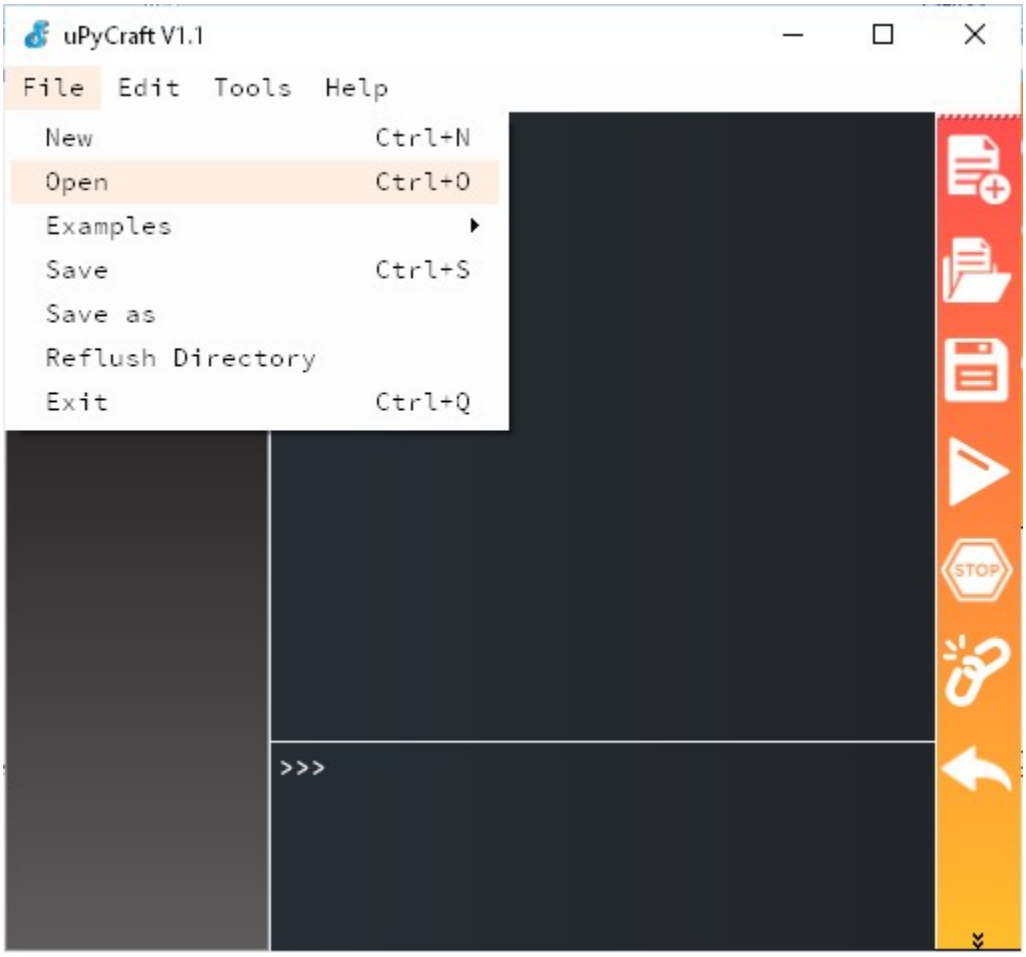
Al terminar de cargar el firmware, elija “Tools”, “Serial” y luego el puerto que en nuestro caso es COM10. Ahora el NodeMCU responde con el promt “>>>”, como se muestra en la imagen abajo.



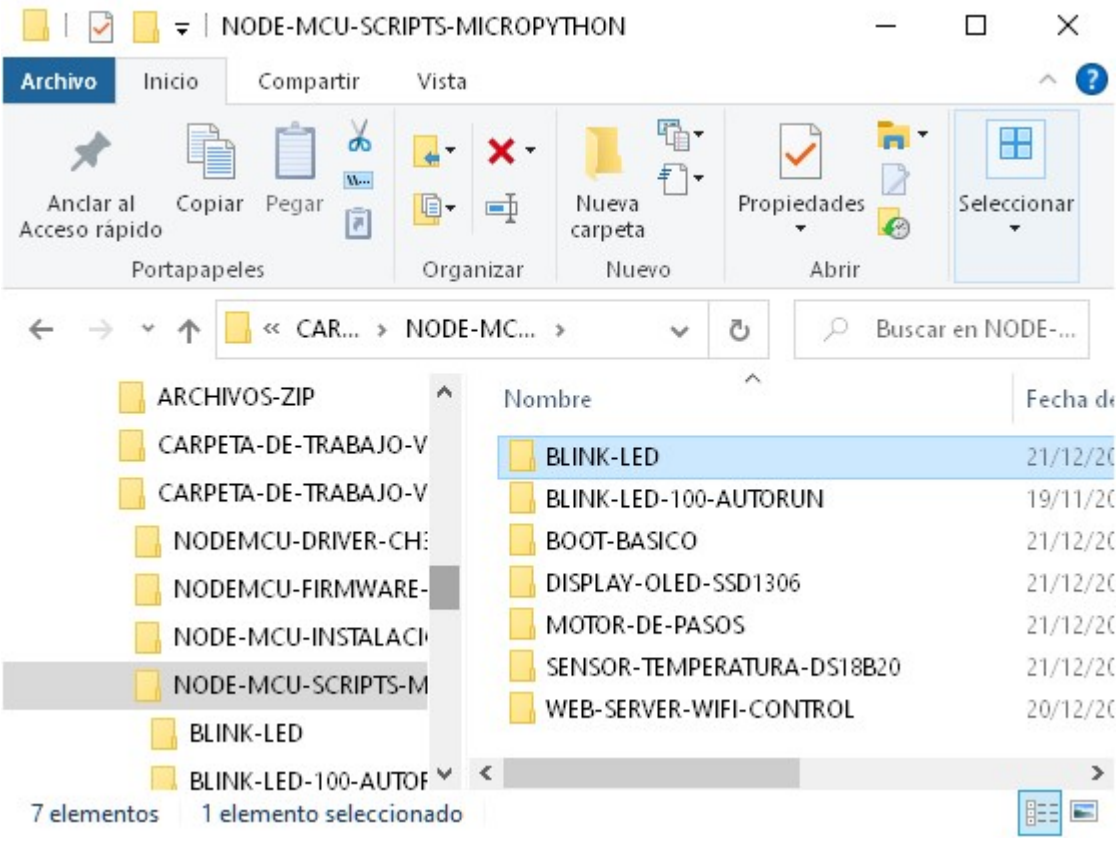
7. Carga del programa de aplicación (flashear un led).

En seguida, cargaremos al NodeMCU el programa más simple: parpadear el led azul 10 veces, con medio segundo de retraso. Este script ó programa se llama “BLINK-LED.py” y lo encuentra en su carpeta de trabajo.

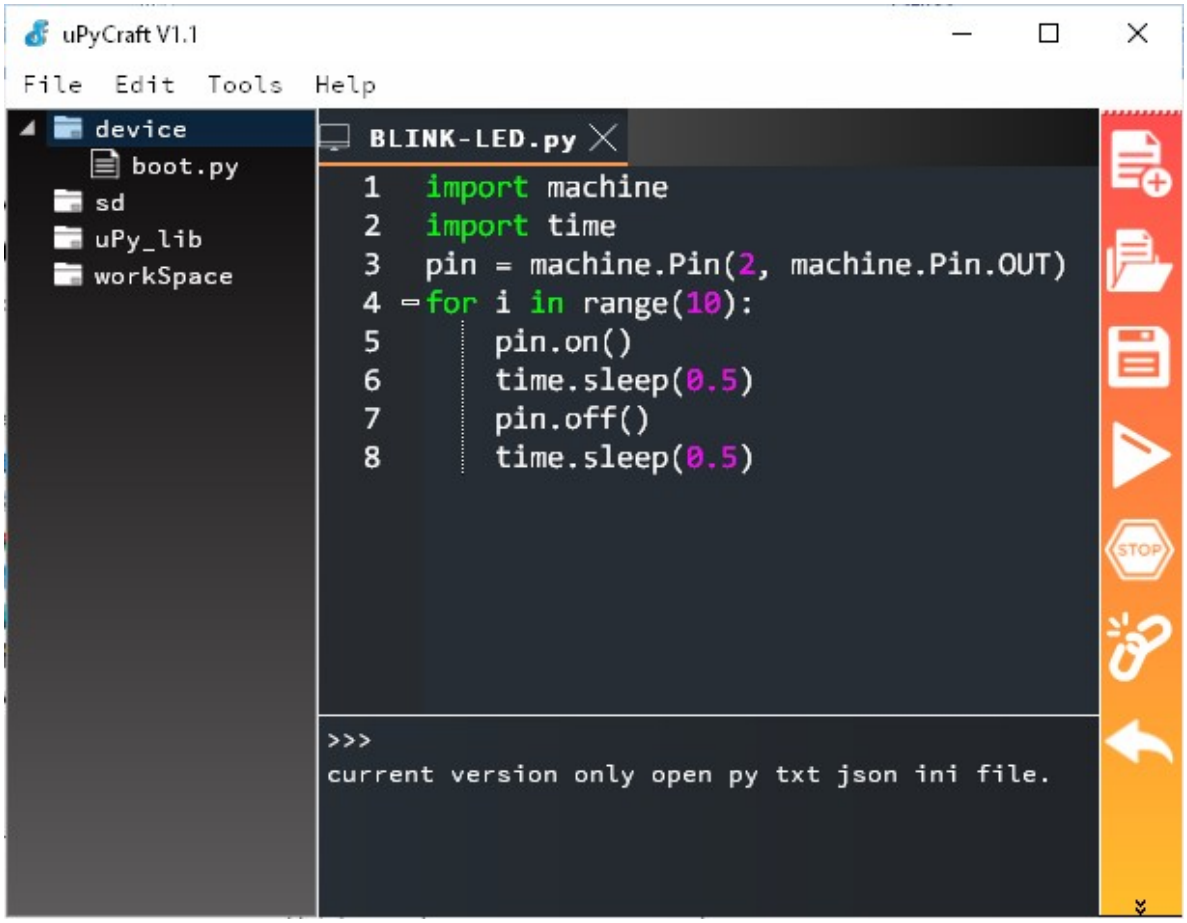
Dé click en “File” y “Open”.



Elija la carpeta “SCRIPTS-MICROPYTHON” > “BLINK-LED” y luego el programa “BLINK-LED.py”

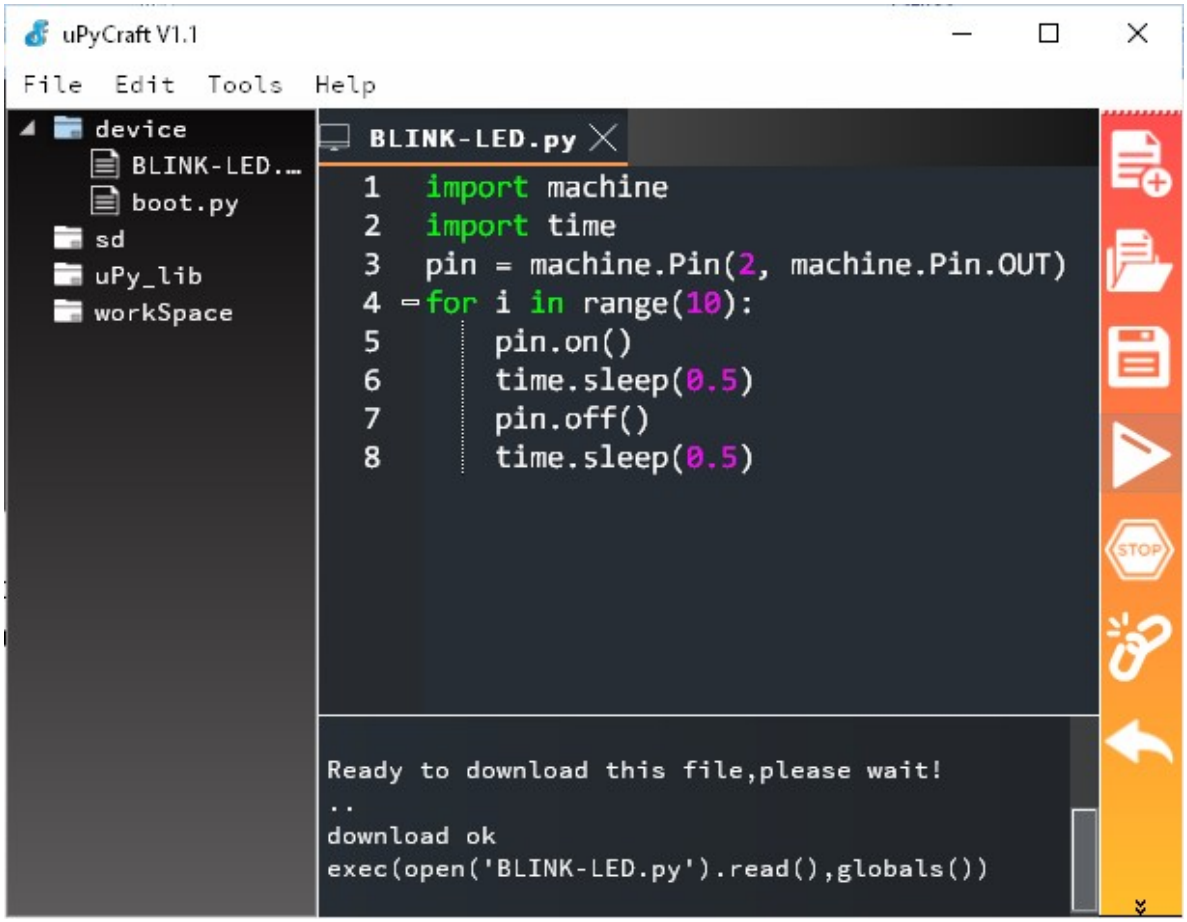


Abra el programa y lo observará en la ventana de su IDE. En este momento el programa aún no ha sido enviado al NodeMCU:



Sobre el lado derecho de la ventana, dé click en el ícono con forma de triángulo (download and run) para transferir el programa al NodeMCU y ejecutarlo automáticamente.

Observe como en la ventana del IDE arriba a la izquierda, aparecerá el nombre del programa “BLINK-LED”. Al mismo tiempo, se ejecuta y usted podrá observar el led azul del NodeMCU flasheando 10 veces. Cuando el programa termina, el NodeMCU envía de nuevo el promt “>>>”.


uPyCraft V1.1
File Edit Tools Help
device
BLINK-LED.…
boot.py
sd
uPy_lib
workSpace
BLINK-LED.py
1 import machine
2 import time
3 pin = machine.Pin(2, machine.Pin.OUT)
4 for i in range(10):
5 pin.on()
6 time.sleep(0.5)
7 pin.off()
8 time.sleep(0.5)
STOP
Ready to download this file,please wait!
..
download ok
exec(open('BLINK-LED.py').read(),globals())