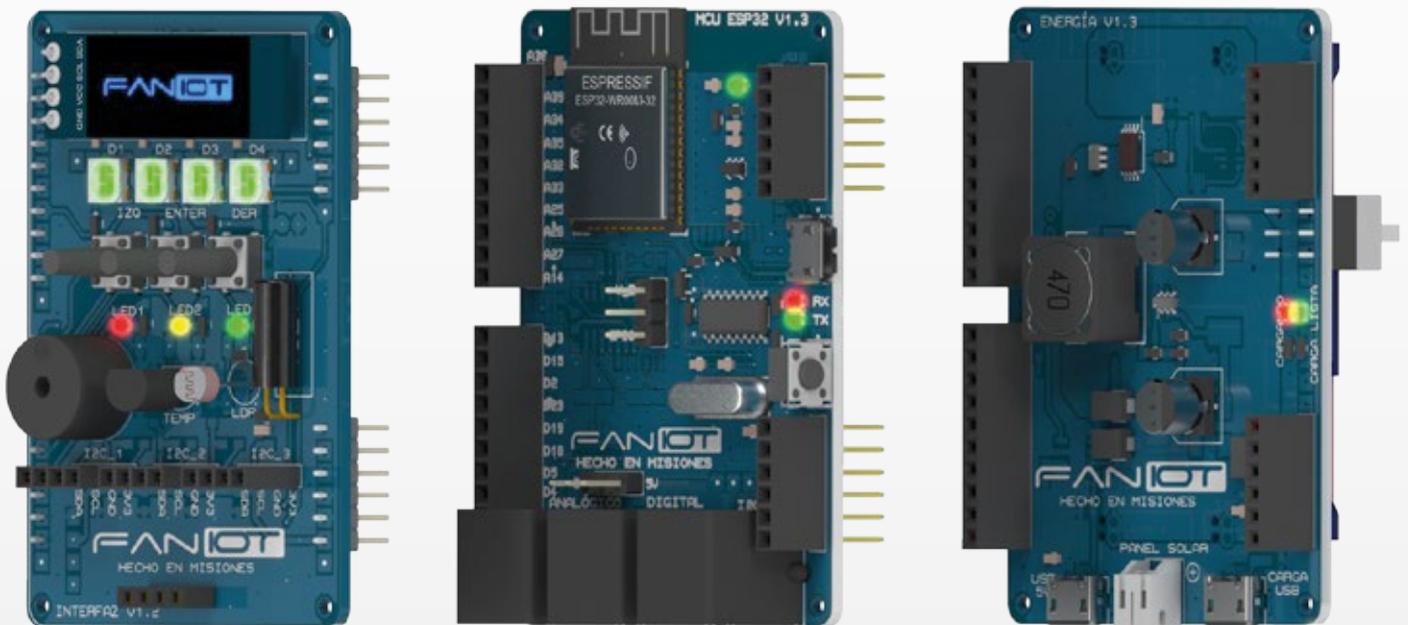


MANUAL EXTENDIDO

MÓDULOS EDUCATIVOS KIT MAKER IOT

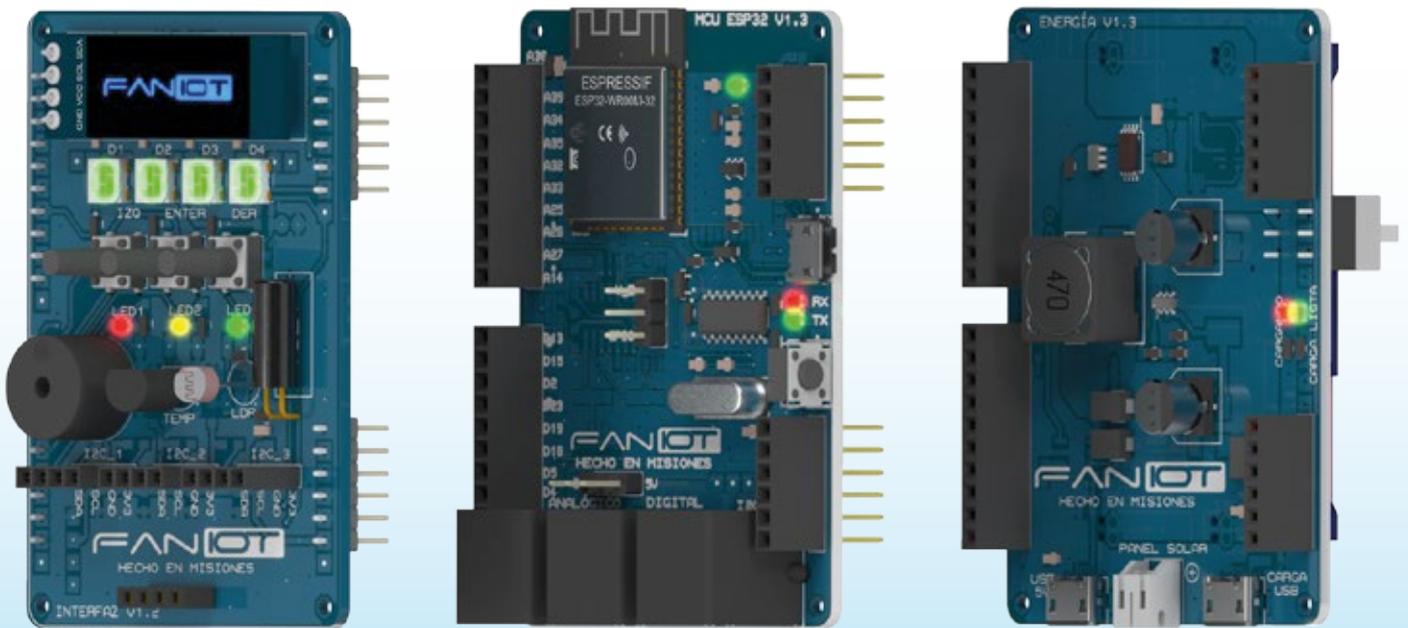


Lea detenidamente esta guía antes de usar el producto y
consérvelo para futuras consultas.

- 02** Características técnicas
- 03** **Módulo Base**
- 05** **Módulo Interfaz**
- 07** **Módulo Alimentación**
- 09** Ensamblajes
- 10** **Conexiones**
- 11** **Recomendaciones**
- 12** Programación
- 13** **Diagrama de Pines**
- 13** **Entorno de Programación**

KIT MAKER IOT

Características técnicas



Módulo BASE

El Módulo Base está integrado por el microcontrolador ESP32. Cuenta con conexión WiFi y Bluetooth, 20 entradas/salidas digitales (de los cuales, 10 se pueden utilizar como entradas analógicas).

Especificaciones Técnicas:

Microcontrolador	ESP32
Reloj	240MHz
Conectividad	WiFi (802.11 b/g/n) Bluetooth v4.2
Alcance WiFi	30m interiores - 90m aire libre
Voltaje de Operaciones	3.3V
Voltaje de Alimentación	5V
Pines E/S Digitales	20
Entradas Analógicas	10
Conectores RJ11	3



fig.1

Peso: 55 gr
Alto: 70 mm
Ancho: 50mm

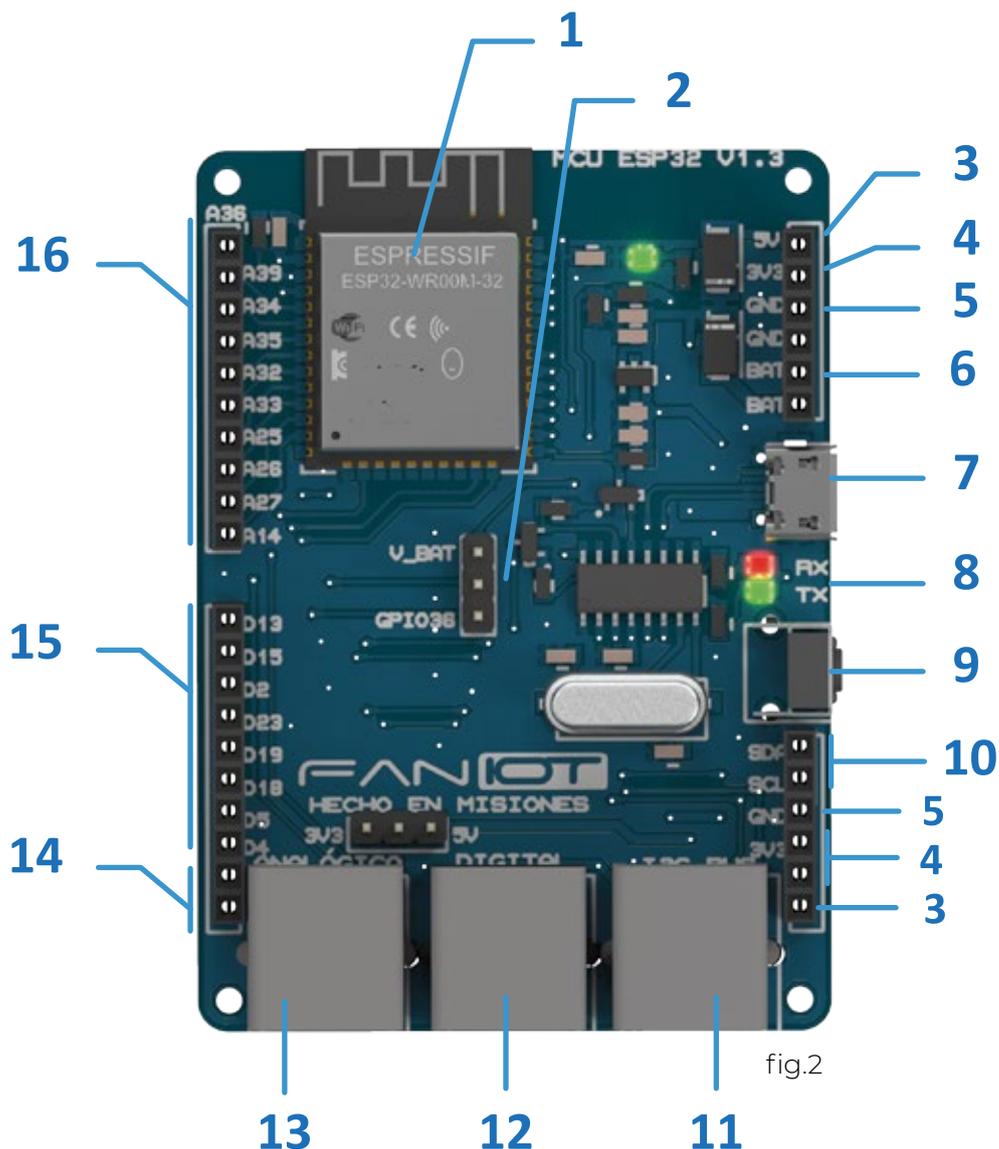


fig.2

Referencias fig.2

- 1** **ESP32:** Es el cerebro del Módulo Base.
 - 2** **Selector de Pines V_BAT/A36:** Con un jumper puede seleccionar entre habilitar el Pin V_BAT o el Pin A36.
 - 3** **Pin 5V:** Salida 5 voltios.
 - 4** **Pin 3.3V:** Salida 3.3 voltios.
 - 5** **Pin GND:** GND.
 - 6** **Pin V_BAT:** Se utiliza para medir la tensión de una batería.
-  **Recordatorio: para habilitar este Pin debe hacerlo con un jumper en el selector de Pines V_BAT/A36.**
- 7** **Micro USB:** Sirve para alimentar y cargar los programas al Módulo Base, y para la comunicación serial con la computadora.
 - 8** **LEDs TX y RX:** Estos LEDs Indican cuando se realiza una comunicación entre la Placa Base y la PC. Parpadean rápidamente cuando se carga un programa, o se establece una comunicación serie.
 - 9** **Botón RESET:** Reinicia la Placa Base.
 - 10** **BUS I2C:** Protocolo síncrono utilizando 2 cables, uno para el reloj (SCL) y otro para el dato (SDA).
 - 11** **Conector RJ11 I2C:** Conecta sensores y actuadores por el protocolo I2C mediante cables con ficha RJ11 para facilitar su conexión.
 - 12** **Conector RJ11 Digital:** Conecta sensores y actuadores como pines digitales mediante cables con ficha RJ11 para facilitar su conexión.
 - 13** **Conector RJ11 Analógico:** Conecta sensores y actuadores como pines analógicos mediante cables con ficha RJ11 para su conexión.
 - 14** **Pines TX y RX:** Pines para la recepción (RX) y transmisión (TX) de datos TTL vía serie.
 - 15** **Pines Digitales:** Se pueden utilizar como entrada/salida en donde el usuario lee y escribe solo dos estados lógicos.
 - 16** **Pines Analógicos:** Se pueden utilizar como entrada/salida en donde el usuario lee y escribe valores de tensión entre 0 y 5 Voltios.

○ Módulo INTERFAZ

El Módulo Interfaz permite la interacción entre el Usuario y el Módulo Base.

El mismo, presenta componentes que facilitan la información a través de luces LED, sonido y su pantalla integrada.

Especificaciones Técnicas:

OLED 128x32
LEDs:
Tiras Inteligentes WS2812b
Monocromáticos
Pulsadores
Buzzer
Sensores:
Temperatura DS18b20
Luz LDR
Shock TILT
Puertos Bus I2C

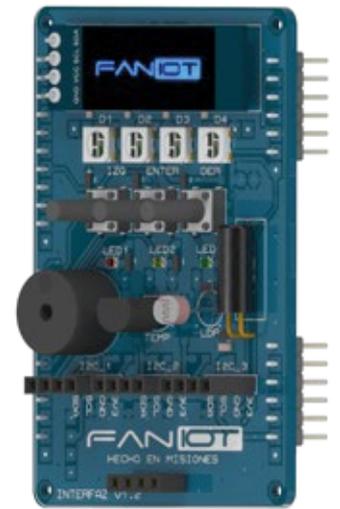


fig.3

Peso: 60 gr
Alto: 70 mm
Ancho: 50 mm

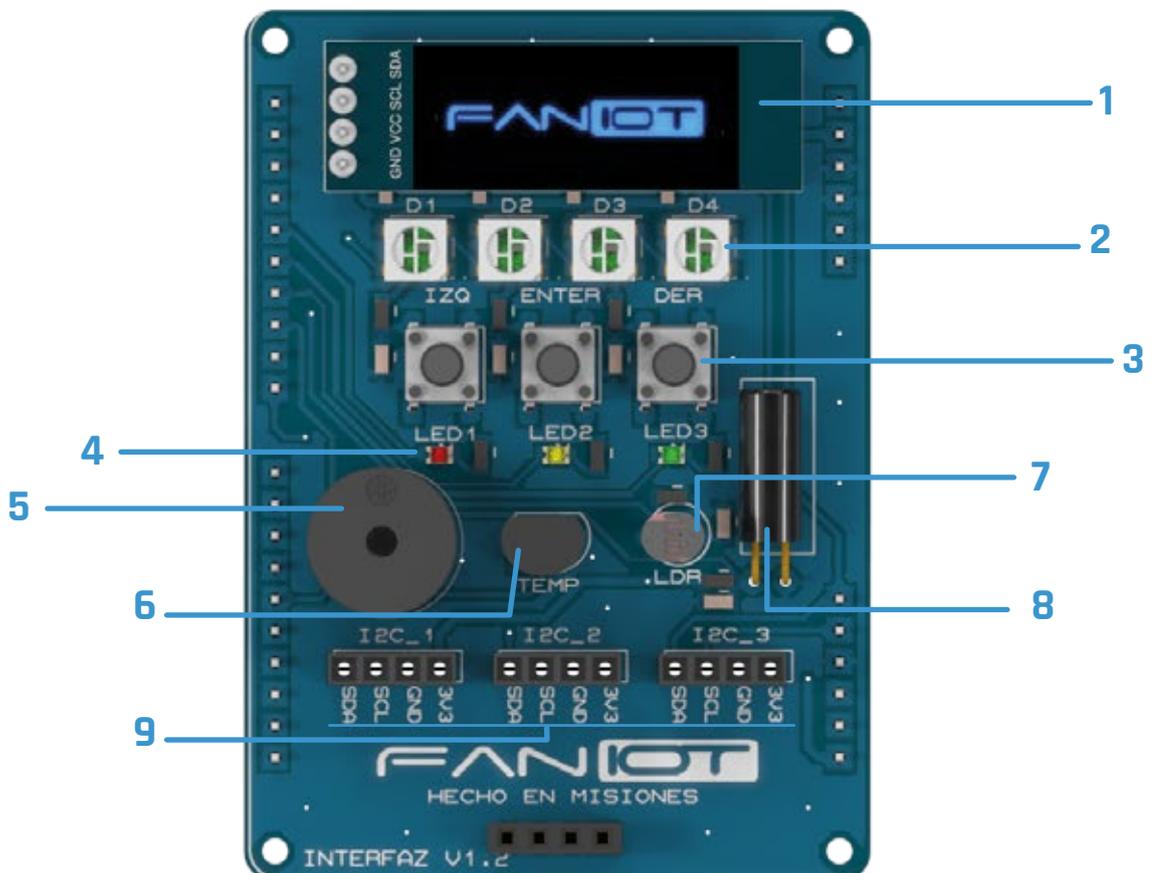


fig.4

Referencias fig.4

- 1 OLED 128x32:** Permite visualizar textos y gráficos.
- 2 LEDs Inteligentes WS2812b:** Tira de LEDs RGB (Rojo, Verde y Azul) permite combinar colores para producir otros nuevos.
- 3 Pulsadores:** Controla el paso de la corriente eléctrica.
- 4 LEDs de Color:** Emiten una luz de color rojo, verde y amarillo.
- 5 Buzzer:** Zumbador el cual convierte una señal eléctrica en una onda de sonido
- 6 Sensor de Temperatura:** Realiza mediciones de temperatura ambiente.
- 7 Sensor de Luz LDR:** Realiza mediciones de intensidad de luz.
- 8 Sensor de Shock TILT:** Define la posición del sensor.
- 9 Bus I2C:** Protocolo síncrono que utiliza 2 cables, uno para el reloj (SCL) y otro para el dato (SDA).

Módulo ALIMENTACIÓN

El Módulo Alimentación funciona como batería para el Kit Maker IOT . Posee conexión para cargarse mediante paneles solares o un cargador USB de 5V.

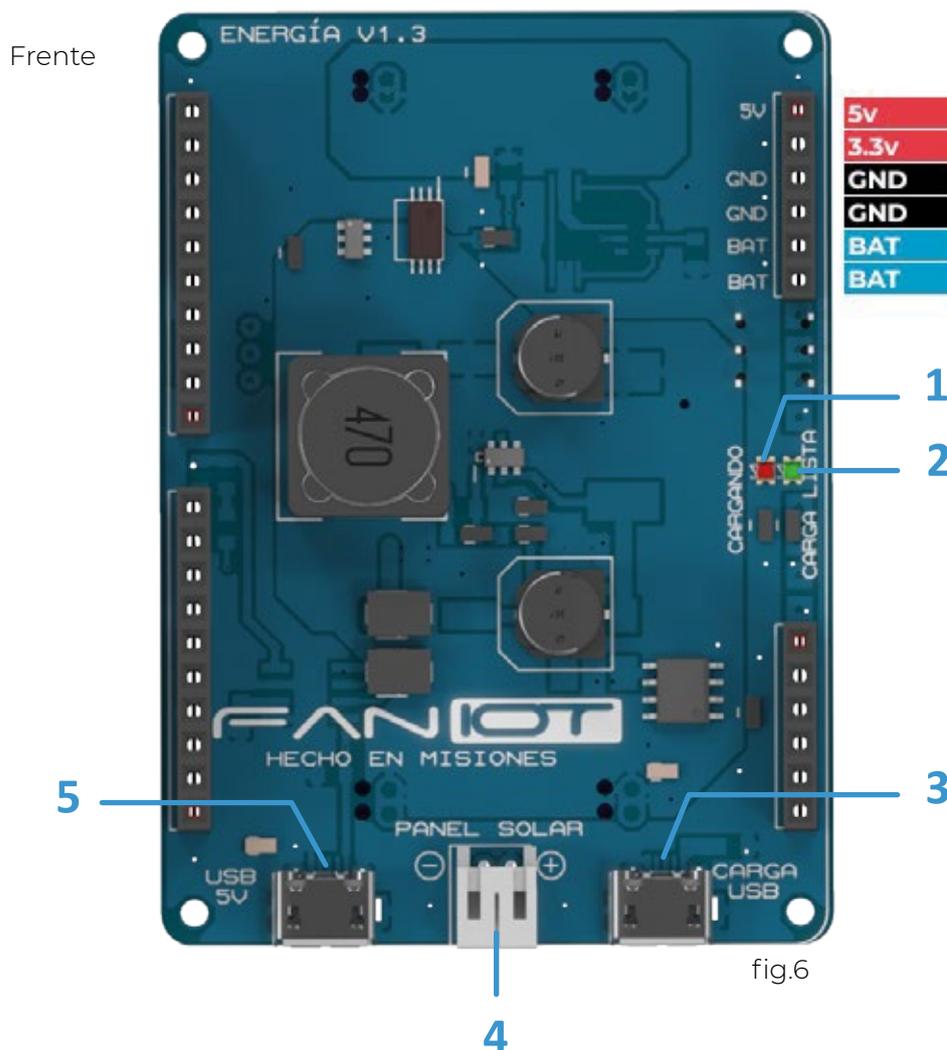
Especificaciones Técnicas:

Soporte para baterías Li-Ion 14500 3.7V
Conectores:
Micro-USB 5V
Micro-USB para la carga
JST para paneles solares
Interruptor Encendido/Apagado
LED indicadores



fig.5

Peso: 90 gr
Alto: 70 mm
Ancho: 50 mm



Referencias fig.6 Y 7

- 1** **LED Charge/Carga:** Indica la carga completa.
Cargando: LED rojo o amarillo encendido.
Carga Finalizada: LED azul o verde encendido.
- 2** **LED Standby:** Indica que el Módulo Alimentación está encendido.
- 3** **Conector USB Carga:** SÓLO para cargar las baterías.
- 4** **Conector JST:** Permite conectar paneles solares de 6V.
- 5** **Conector USB 5V:** Sirve para alimentar todos los Módulos.

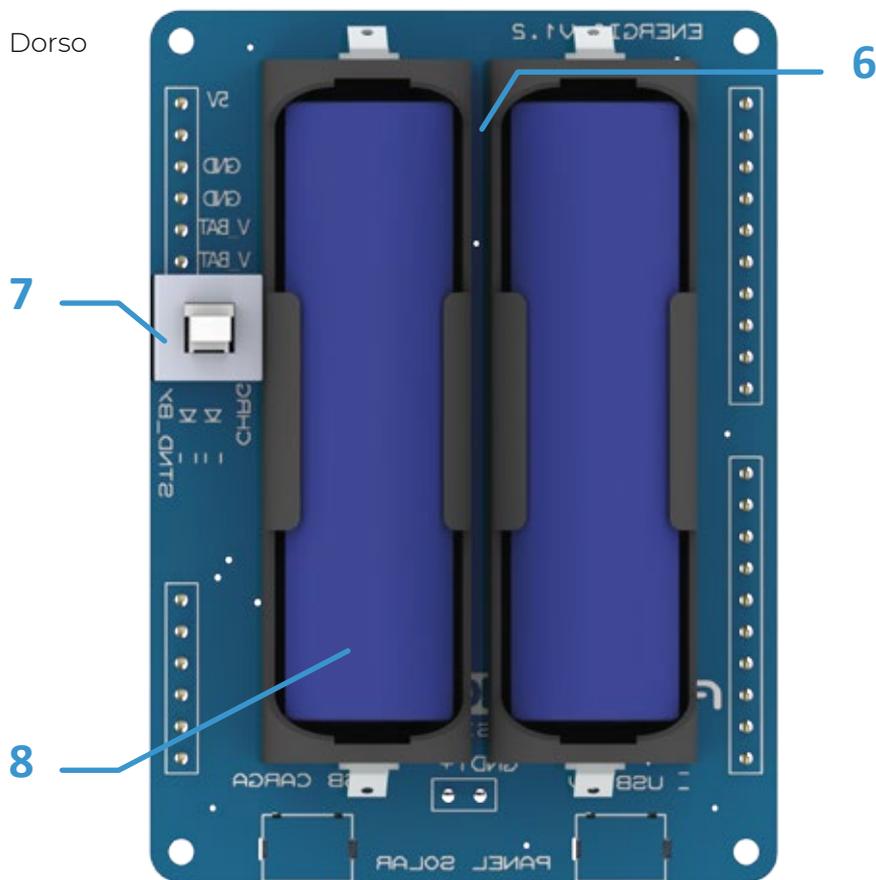


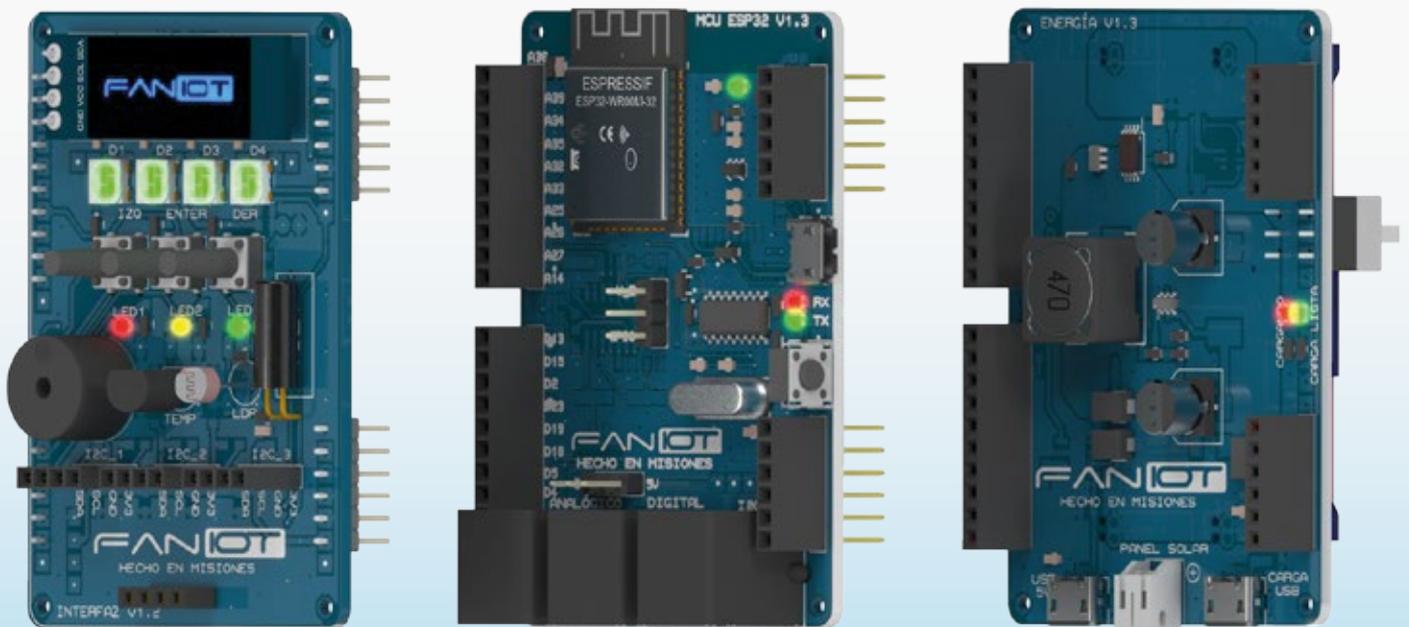
fig7

- 6** **Soporte batería:** Contiene las baterías Li-Ion 14500 3.7V
- 7** **Interruptor:** Enciende o apaga el módulo de alimentación.
- 8** **Batería:** Pila Li-Ion 14500 3.7V

! **IMPORTANTE:** Las baterías deben estar correctamente colocadas, caso contrario pueden ocasionarse daños al equipo.

KIT MAKER IOT

Conexiones



- 1- Acoplar el **Módulo Interfaz** al **Módulo Base** conectando la tira de pines machos con la tira de pines hembra.

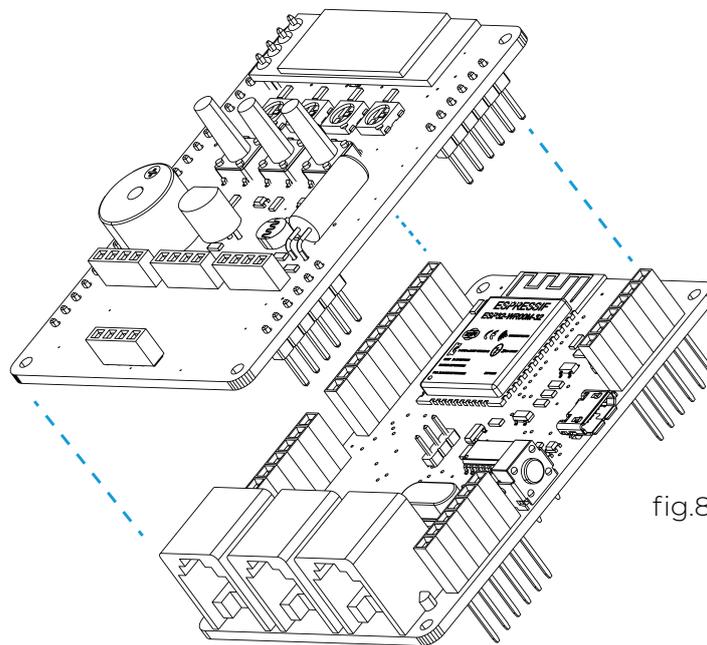
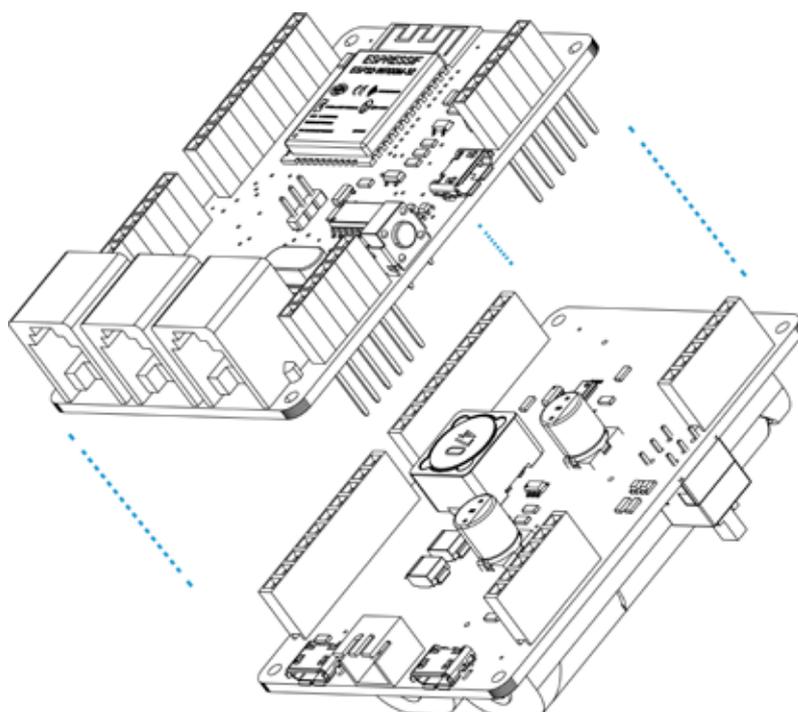


fig.8

- 2- Verificar que el Módulo Alimentación se encuentre apagado.
- 3- Corroborar que las baterías estén correctamente colocadas.
- 4- Acoplar el Módulo Alimentación por debajo del Módulo Base.
- 5- Presionar el interruptor para encender el dispositivo.



Consideraciones para la Carga

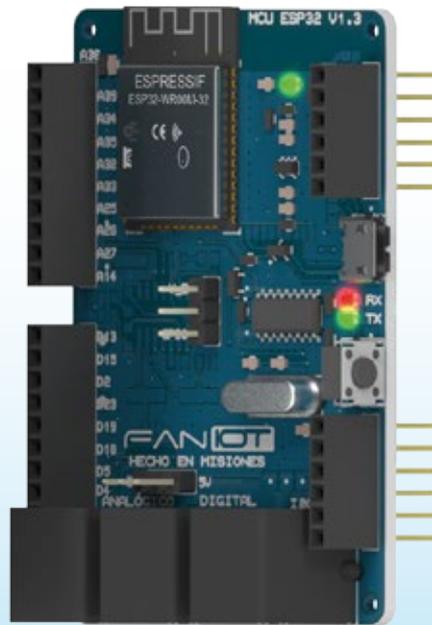
- Para aumentar la autonomía, utilizar las dos baterías.
- Es necesario un cable Micro USB y un transformador 5V/1A (un cargador de celular).
- Conectar **un sólo** puerto Micro USB a la vez.
- La finalización de la carga es indicada en el **LED Charge/Carga** (luz azul o verde).
- Realizar la conexión con el **Interruptor** apagado.
- Utilizar paneles solares de corriente mayor a 200mA.
- La autonomía de las baterías dependerá de su uso.



Los paneles solares se adquieren por separado.

KIT MAKER IOT

Programación



01 Diagrama de Pines

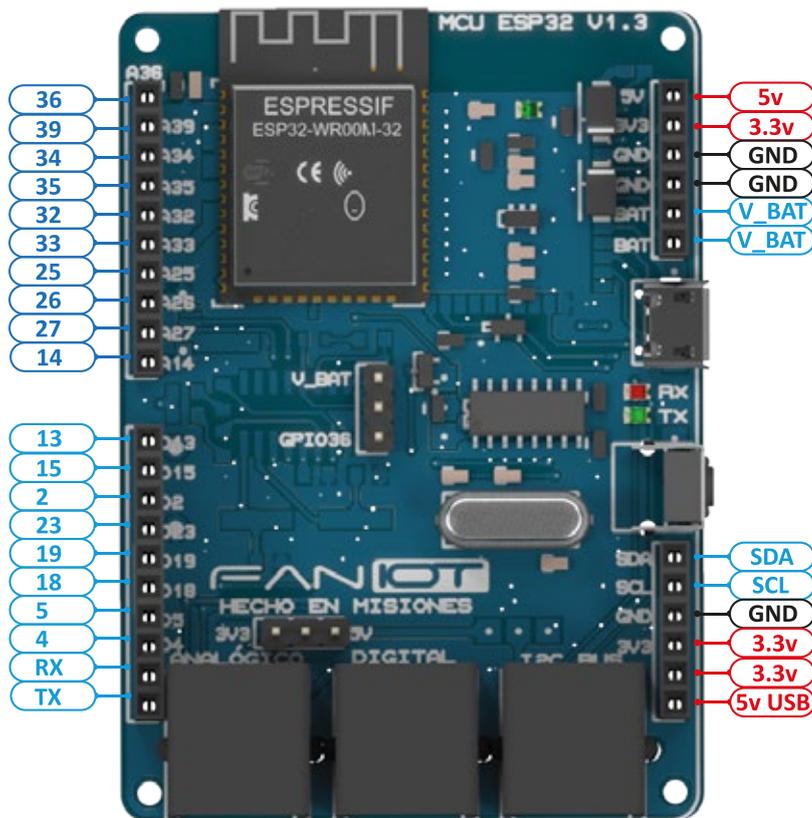


fig.8 - Módulo Base

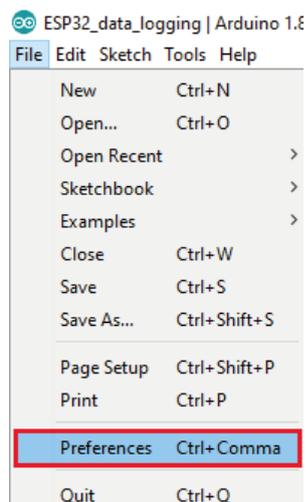
Cada PIN contiene un número por el cual podemos identificar y llamarlo dentro del código.

02 Entorno de Programación

Para instalar la placa ESP32 en tu Arduino IDE, sigue las siguientes instrucciones:

A) Instalación del complemento ESP32 en Arduino IDE

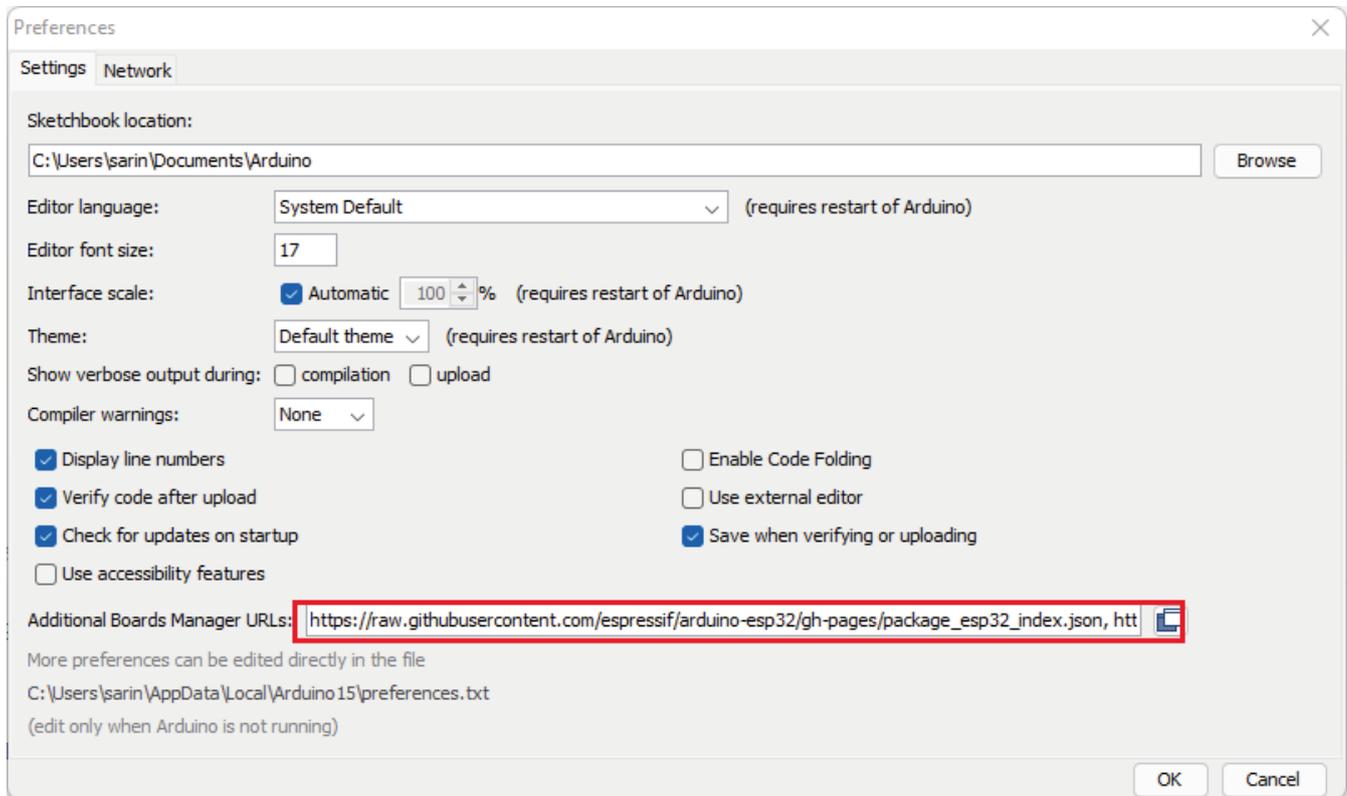
A.1 En tu IDE de Arduino, ve a **File > Preferences**



A.2 Ingrese lo siguiente en el campo "URL adicionales del administrador de la junta":

```
https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json
```

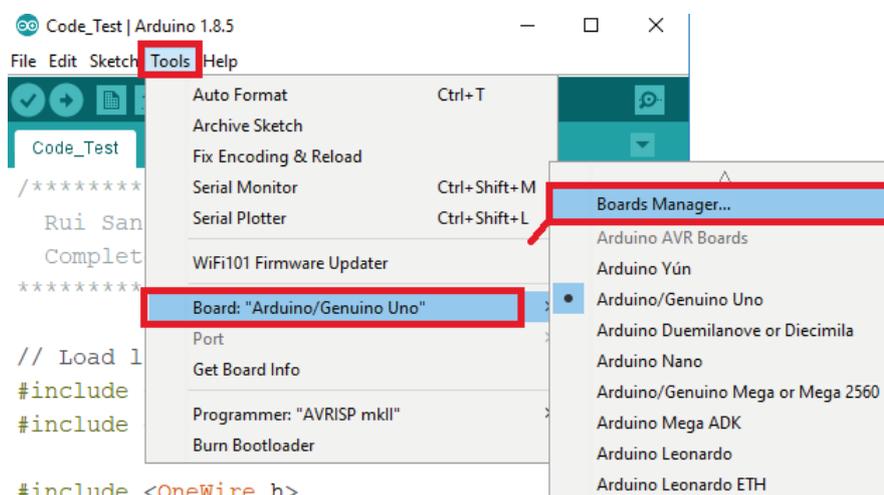
Luego, haga clic en el botón "Aceptar":



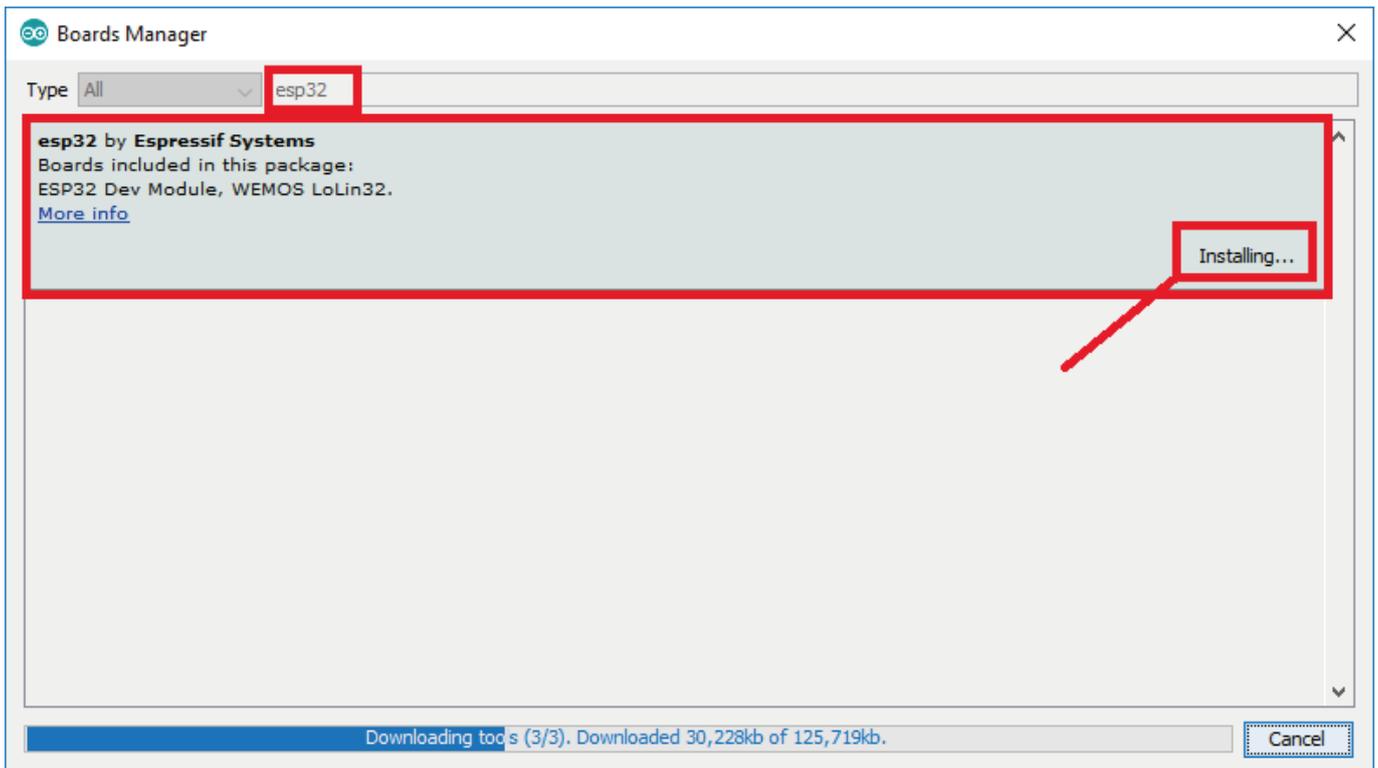
Nota: si ya tiene la URL de los tableros ESP8266, puede separar las URL con una coma como sigue:

```
https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json,  
http://arduino.esp8266.com/stable/package_esp8266com_index.json
```

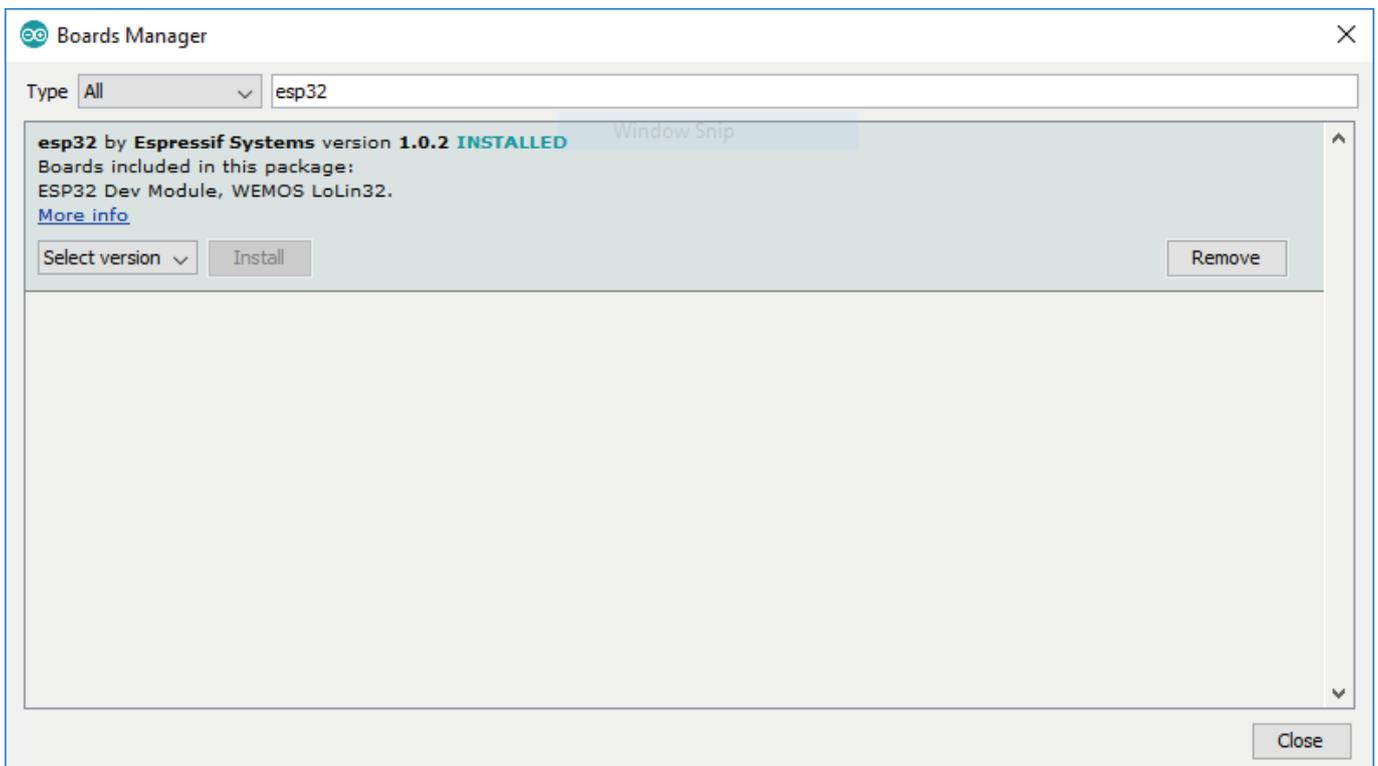
A.3 Abra el Administrador de tableros. Ir a **Tools > Board > Boards Manager...**



A.4 Busque **ESP32** y presione el botón de instalación para "ESP32 by Espressif Systems":



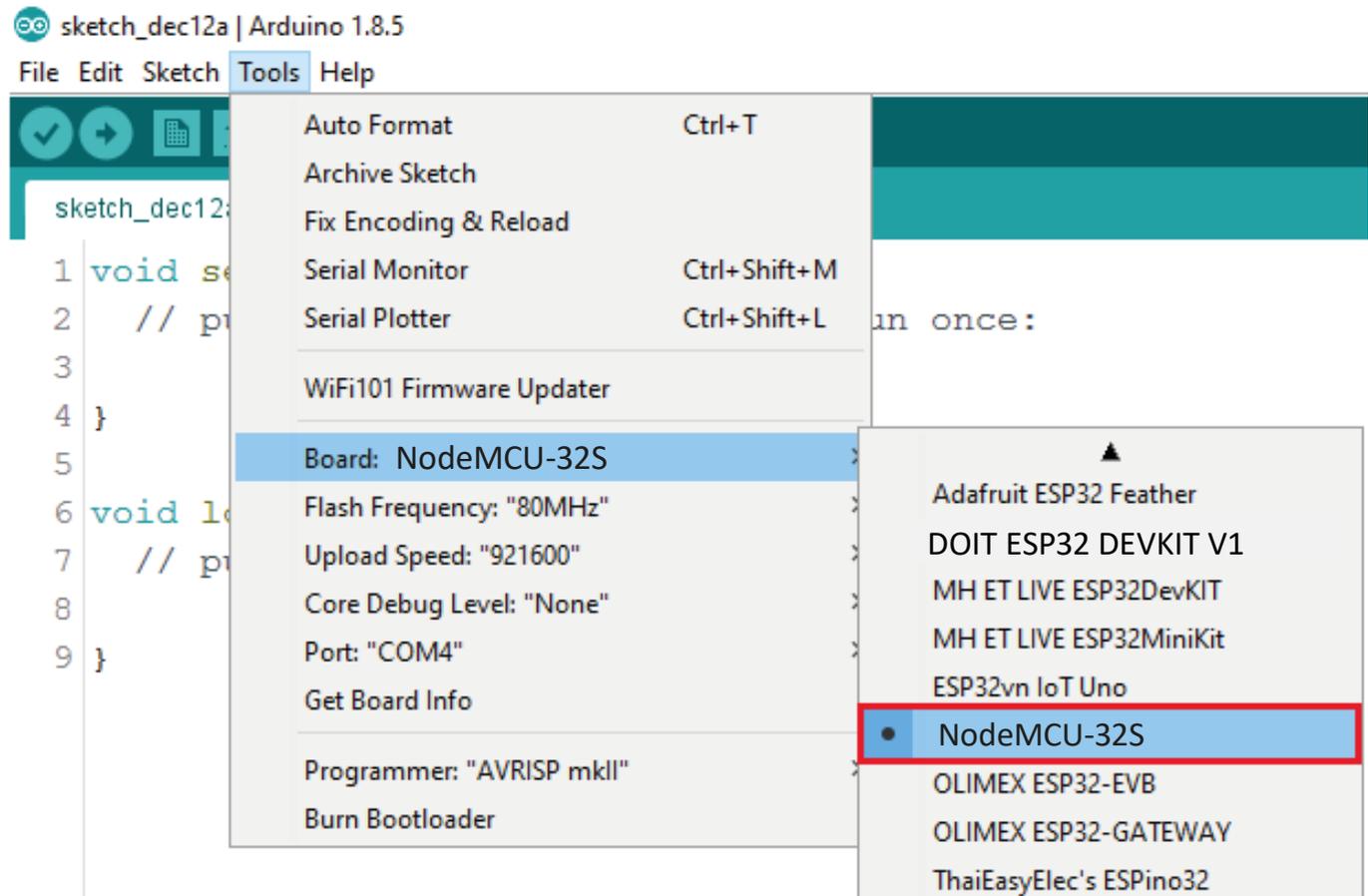
A.5 Eso es todo. Debe instalarse después de unos segundos.



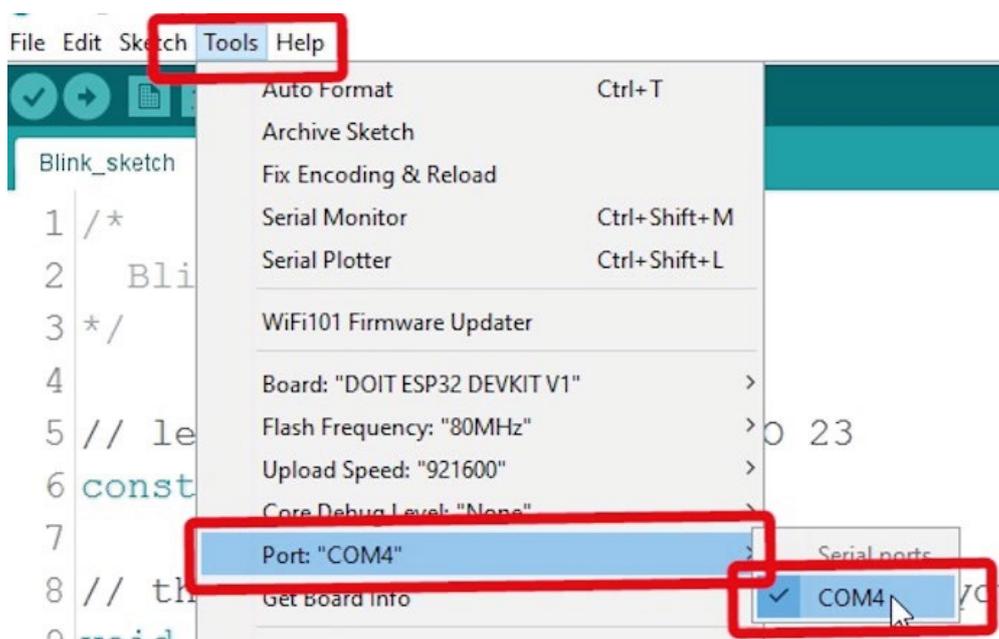
B) Prueba de la instalación

Conecte la placa ESP32 a su computadora. Con su Arduino IDE abierto, siga estos pasos:

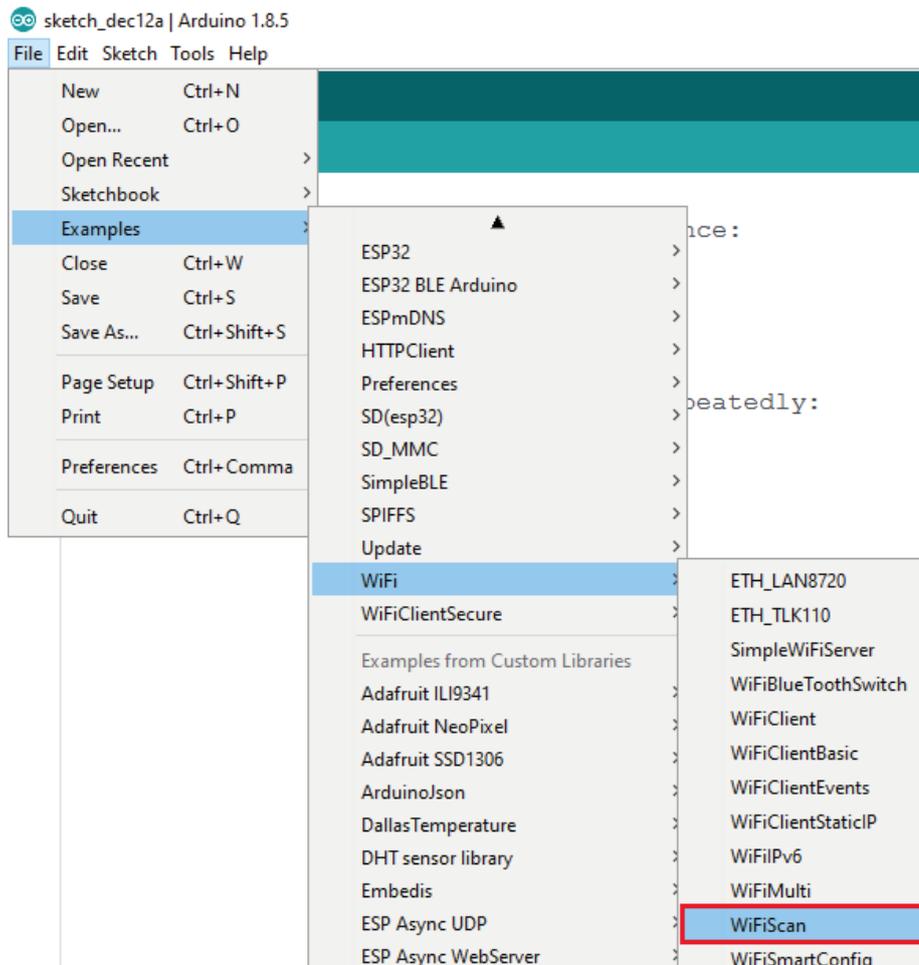
B.1 Selecciona tu Tablero en el menú **Tools > Board** (en mi caso es el NodeMCU-32S).



B.2 Seleccione el puerto (si no ve el puerto COM en su IDE de Arduino, debe instalar el Controladores CP210x USB a UART Bridge VCP)



B.3 Abra el siguiente ejemplo bajo **File > Examples > WiFi (ESP32) > WiFiScan**



B.4 Se abre un nuevo boceto en su IDE de Arduino:



B.5 Presione el botón **Cargar** en el IDE de Arduino. Espere unos segundos mientras el código se compila y carga en su tablero.



B.6 Si todo salió como se esperaba, debería ver el mensaje "**Carga finalizada**". mensaje.

```
Done uploading.
writing at 0x0004c000... (84 %)
Writing at 0x00050000... (89 %)
Writing at 0x00054000... (94 %)
Writing at 0x00058000... (100 %)
Wrote 481440 bytes (299651 compressed) at 0x00010000 in 4.7 seconds
Hash of data verified.
Compressed 3072 bytes to 122...

Writing at 0x00008000... (100 %)
Wrote 3072 bytes (122 compressed) at 0x00008000 in 0.0 seconds (elapsed 0.0 s)
Hash of data verified.

Leaving...
Hard resetting...
```

DOIT ESP32 DEVKIT V1, 80MHz, 921600, None on COM4

B.7 Abra Arduino IDE Serial Monitor a una velocidad de transmisión de 115200:



B.8 Presione el botón **Habilitar** a bordo ESP32 y debería ver las redes disponibles cerca tu ESP32:

```
COM4
scan done
2 networks found
1: MEO-620B4B (-49)*
2: MEO-WiFi (-50)

scan start
scan done
2 networks found
1: MEO-620B4B (-48)*
2: MEO-WiFi (-49)
```

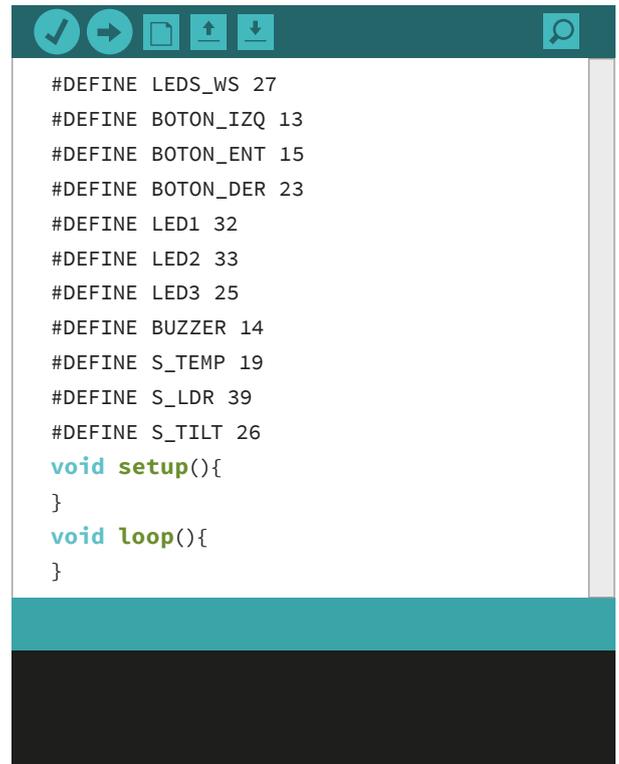
Autoscroll Both NL & CR 115200 baud Clear output

C) Declaración de Variables

Para utilizar cada componente del Módulo de Interfaz es necesario declarar, en nuestro entorno para programar, el número de pin correspondiente.

C.1 Nombre del componente y número de pin correspondiente:

Componentes	N° de PIN
LEDs WS2812B	27
Botón izquierdo	13
Botón Enter	15
Botón derecho	23
LED 1	32
LED 2	33
LED 3	25
Buzzer	14
Sensor de temperatura	19
Sensor de luz LRD	39
Sensor de shock TILT	26



```
#DEFINE LEDES_WS 27
#DEFINE BOTON_IZQ 13
#DEFINE BOTON_ENT 15
#DEFINE BOTON_DER 23
#DEFINE LED1 32
#DEFINE LED2 33
#DEFINE LED3 25
#DEFINE BUZZER 14
#DEFINE S_TEMP 19
#DEFINE S_LDR 39
#DEFINE S_TILT 26
void setup(){
}
void loop(){
}
```

D) IDE Code

El código IDE se compone de varias partes:

- 1) Verificar
- 2) Cargar
- 3) Nuevo
- 4) Abrir
- 5) Guardar
- 6) Monitor Serial
- 7) Editor
- 8) Notificaciones

D.1 Para hacer tu primer programa debes: digitar el siguiente código como en la imagen.



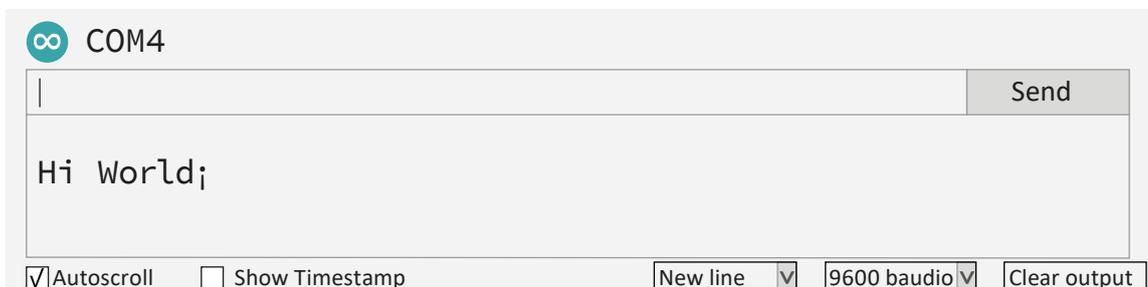
```
1 2 3 4 5 6
void setup(){
7   Serial.begin(9600);
   delay(5000);
   Serial.println("Hola Mundo!");
   }
   void loop(){
   }
8
```

D.2 Cargue el código en el Módulo. Para ello, debe hacer clic en el botón de carga y esperar unos segundos para que se cargue el código.

D.3 Cuando el código esté **completamente cargado**, el mensaje Cargado se mostrará en la barra de estado.

D.4 Reinicie la placa presionando el botón **RESET** y haciendo clic en el icono del **monitor en serie**.

D.5 Se abrirá una ventana con el siguiente mensaje:



*Recuerde establecer este valor en 9600 baudios.



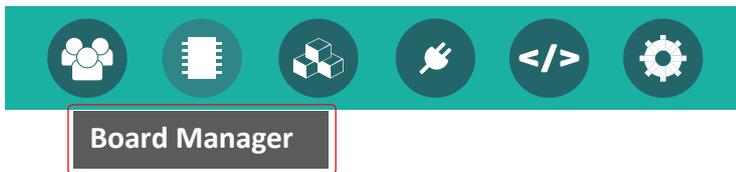
¡Terminado!

Ya puedes empezar a programar tu Módulo FANIOT

E) Descarga, Instalación y Configuración del Entorno Programación de bloques

El entorno de programación de bloques le permite programar instrucciones para su módulo, por lo que práctico y visual. Permite al alumno abordar la programación desde una usabilidad amigable en que integra conceptos de robótica educativa con IOT.

- E.1 Para utilizar el Entorno de Programación de Bloques debes descargarlo desde:
<https://faniot.com.ar/descarga/FAN-BLOCK.exe>
- E.2 Ejecute el instalador del programa y siga los pasos de instalación.
- E.3 Una vez instalado el entorno de programación de bloques, debe inicializarlo. Haga clic en "Administrador de la Junta".

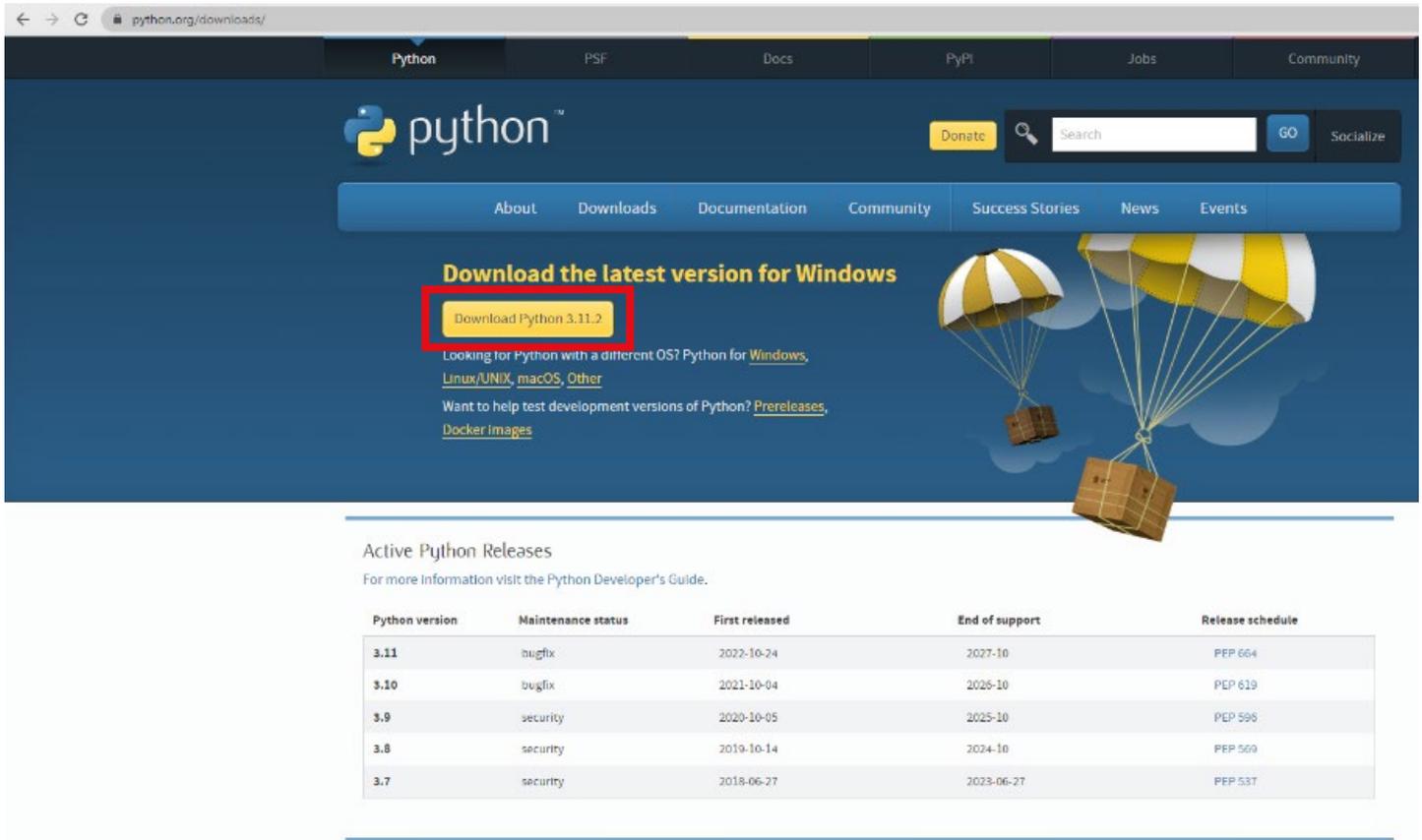


- E.4 Busque en Instalado la opción "Generic ESP32 WROOM", haga clic en "Cambiar tablero" y confirme.



F) Instalar python

F.1 Instalar python (ultima versión) desde <https://www.python.org/downloads/>



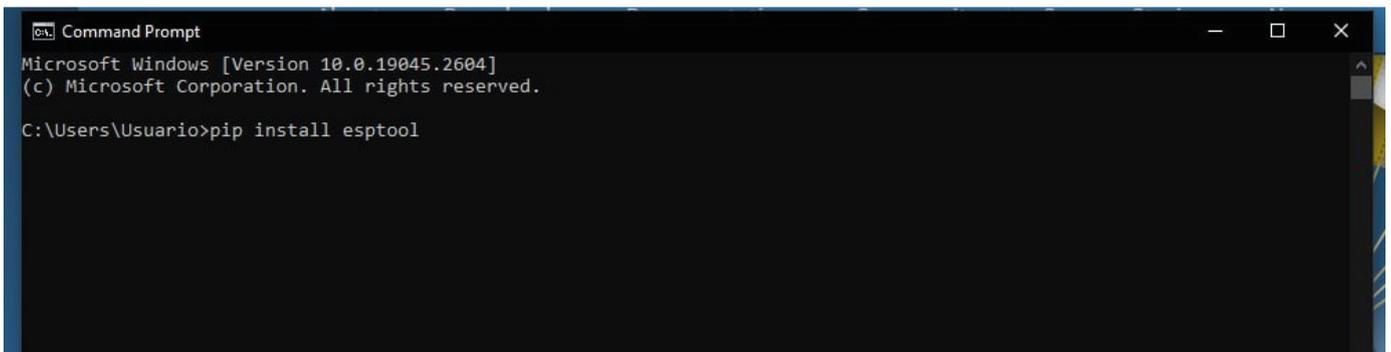
The screenshot shows the Python.org website's download page. The main heading is "Download the latest version for Windows". A yellow button labeled "Download Python 3.11.2" is highlighted with a red box. Below it, there are links for other operating systems: "Looking for Python with a different OS? Python for [Windows](#), [Linux/UNIX](#), [macOS](#), [Other](#)". There are also links for "Prereleases" and "Docker images". The page features a background image of two parachutes with cargo boxes. Below the main content, there is a section titled "Active Python Releases" with a table of release information.

Python version	Maintenance status	First released	End of support	Release schedule
3.11	bugfix	2022-10-24	2027-10	PEP 664
3.10	bugfix	2021-10-04	2026-10	PEP 619
3.9	security	2020-10-05	2025-10	PEP 596
3.8	security	2019-10-14	2024-10	PEP 569
3.7	security	2018-06-27	2023-06-27	PEP 537

F) Instalar herramienta ESP:

F.1 Abra cmd y ejecute: (en caso de que uno no funcione, intente lo siguiente).

- pip install esptool
- python -m pip install esptool
- pip2 install esptool



The screenshot shows a Windows Command Prompt window. The text displayed is: "Microsoft Windows [Version 10.0.19045.2604] (c) Microsoft Corporation. All rights reserved. C:\Users\Usuario>pip install esptool".

```
Command Prompt
Microsoft Windows [Version 10.0.19045.2604]
(c) Microsoft Corporation. All rights reserved.

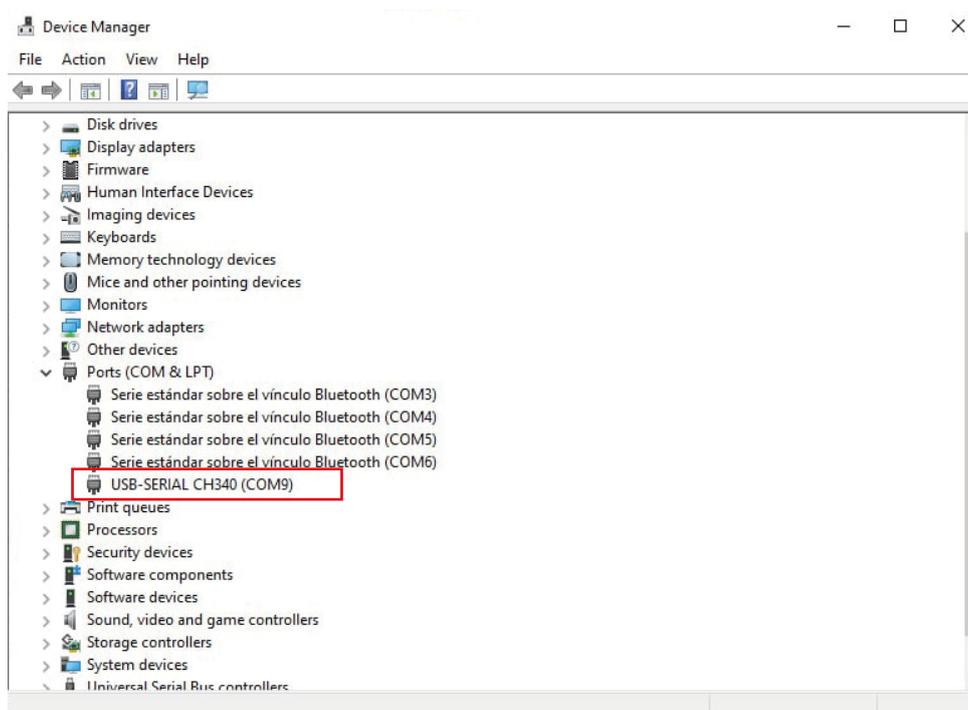
C:\Users\Usuario>pip install esptool
Collecting esptool
  Downloading esptool-4.5.1.tar.gz (252 kB)
----- 252.2/252.2 KB 5.1 MB/s eta 0:00:00
  Preparing metadata (setup.py) ... done
Collecting bitstring>=3.1.6
  Downloading bitstring-4.0.1-py3-none-any.whl (45 kB)
----- 45.6/45.6 KB 2.4 MB/s eta 0:00:00
Collecting cryptography>=2.1.4
  Downloading cryptography-39.0.2-cp36-abi3-win_amd64.whl (2.5 MB)
----- 2.5/2.5 MB 13.0 MB/s eta 0:00:00
Collecting ecdsa>=0.16.0
  Downloading ecdsa-0.18.0-py2.py3-none-any.whl (142 kB)
----- 142.9/142.9 KB ? eta 0:00:00
Collecting pyserial>=3.0
  Using cached pyserial-3.5-py2.py3-none-any.whl (90 kB)
Collecting reedsolo<=1.6.0,>=1.5.3
  Downloading reedsolo-1.6.0-cp310-cp310-win_amd64.whl (174 kB)
----- 174.2/174.2 KB 5.3 MB/s eta 0:00:00
Requirement already satisfied: cffi>=1.12 in c:\users\usuario\AppData\Local\Programs\Python\Python310\lib\site-packages (
from cryptography>=2.1.4->esptool) (1.15.0)
Requirement already satisfied: six>=1.9.0 in c:\users\usuario\AppData\Local\Programs\Python\Python310\lib\site-packages (f
rom ecdsa>=0.16.0->esptool) (1.16.0)
Requirement already satisfied: pycparser in c:\users\usuario\AppData\Local\Programs\Python\Python310\lib\site-packages (f
rom cffi>=1.12->cryptography>=2.1.4->esptool) (2.21)
Using legacy 'setup.py install' for esptool, since package 'wheel' is not installed.
Installing collected packages: reedsolo, pyserial, ecdsa, bitstring, cryptography, esptool
  Running setup.py install for esptool ... done
Successfully installed bitstring-4.0.1 cryptography-39.0.2 ecdsa-0.18.0 esptool-4.5.1 pyserial-3.5 reedsolo-1.6.0
WARNING: You are using pip version 22.0.4; however, version 23.0.1 is available.
You should consider upgrading via the 'C:\Users\Usuario\AppData\Local\Programs\Python\Python310\python.exe -m pip install
--upgrade pip' command.

C:\Users\Usuario>
```

F.2 La memoria flash se borra:

esptool.py --chip esp32 --port COMX erase_flash (Reemplace la X con el número de puerto COM)

(mientras mantiene presionado el botón de inicio, no es posible en el KMI)



Abra el administrador de dispositivos y reemplace la x con el número de puerto com

```
Command Prompt
----- 252.2/252.2 KB 5.1 MB/s eta 0:00:00
Preparing metadata (setup.py) ... done
Collecting bitstring>=3.1.6
  Downloading bitstring-4.0.1-py3-none-any.whl (45 kB)
----- 45.6/45.6 KB 2.4 MB/s eta 0:00:00
Collecting cryptography>=2.1.4
  Downloading cryptography-39.0.2-cp36-abi3-win_amd64.whl (2.5 MB)
----- 2.5/2.5 MB 13.0 MB/s eta 0:00:00
Collecting ecdsa>=0.16.0
  Downloading ecdsa-0.18.0-py2.py3-none-any.whl (142 kB)
----- 142.9/142.9 KB ? eta 0:00:00
Collecting pyserial>=3.0
  Using cached pyserial-3.5-py2.py3-none-any.whl (90 kB)
Collecting reedsolo<=1.6.0,>=1.5.3
  Downloading reedsolo-1.6.0-cp310-cp310-win_amd64.whl (174 kB)
----- 174.2/174.2 KB 5.3 MB/s eta 0:00:00
Requirement already satisfied: cffi>=1.12 in c:\users\usuario\AppData\Local\Programs\Python\Python310\lib\site-packages (
from cryptography>=2.1.4->esptool) (1.15.0)
Requirement already satisfied: six>=1.9.0 in c:\users\usuario\AppData\Local\Programs\Python\Python310\lib\site-packages (
from ecdsa>=0.16.0->esptool) (1.16.0)
Requirement already satisfied: pycparser in c:\users\usuario\AppData\Local\Programs\Python\Python310\lib\site-packages (f
rom cffi>=1.12->cryptography>=2.1.4->esptool) (2.21)
Using legacy 'setup.py install' for esptool, since package 'wheel' is not installed.
Installing collected packages: reedsolo, pyserial, ecdsa, bitstring, cryptography, esptool
  Running setup.py install for esptool ... done
Successfully installed bitstring-4.0.1 cryptography-39.0.2 ecdsa-0.18.0 esptool-4.5.1 pyserial-3.5 reedsolo-1.6.0
WARNING: You are using pip version 22.0.4; however, version 23.0.1 is available.
You should consider upgrading via the 'C:\Users\Usuario\AppData\Local\Programs\Python\Python310\python.exe -m pip install
--upgrade pip' command.

C:\Users\Usuario>esptool.py --chip esp32 --port COM9 erase_flash
```

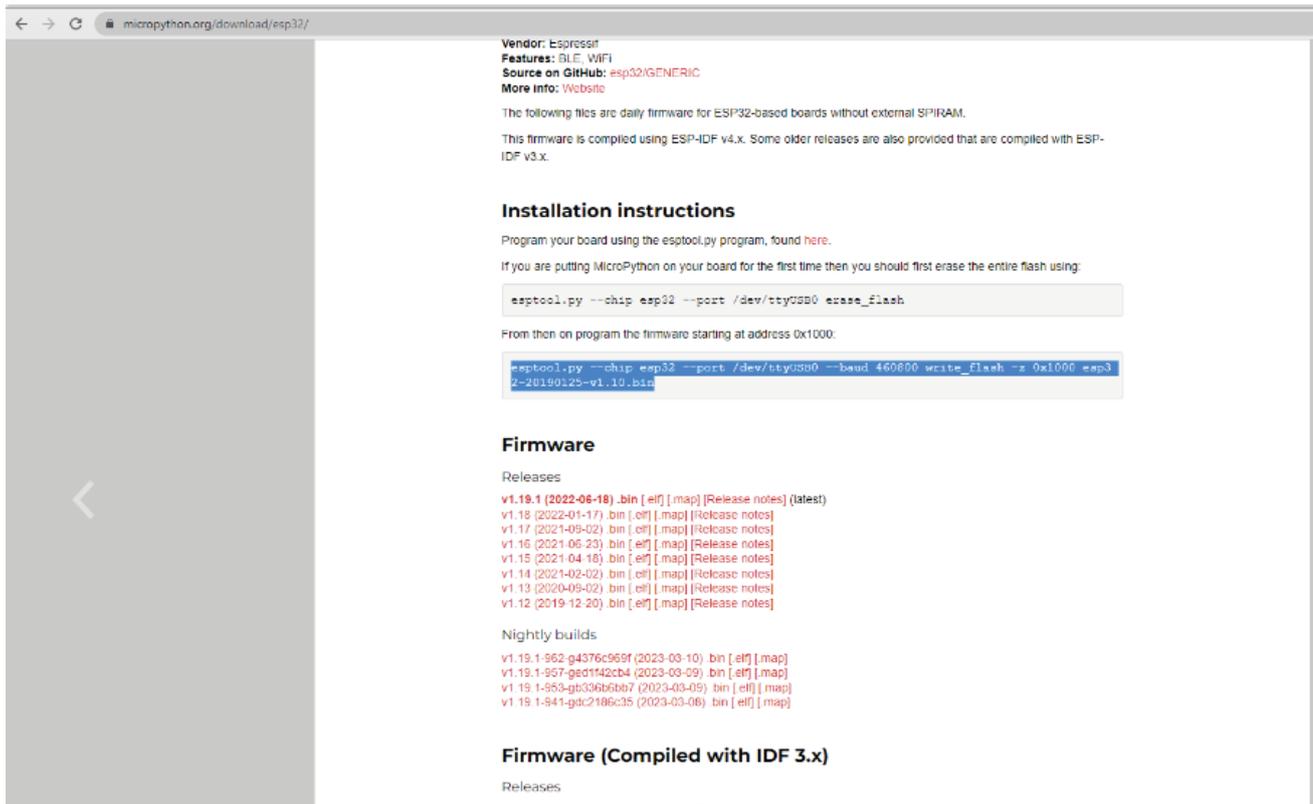
F.3 Descargue el último firmware de micropython para ESP32 desde:

<https://micropython.org/download/esp32/>

The screenshot shows the Micropython.org website for downloading ESP32 firmware. The page includes the following sections:

- Vendor:** Espressif
- Features:** BLE, WiFi
- Source on GitHub:** esp32/GENERIC
- More info:** Website
- Installation instructions:** Program your board using the esptool.py program, found here. If you are putting MicroPython on your board for the first time then you should first erase the entire flash using:
`esptool.py --chip esp32 --port /dev/ttyUSB0 erase_flash`
From then on program the firmware starting at address 0x1000:
`esptool.py --chip esp32 --port /dev/ttyUSB0 --baud 460800 write_flash -z 0x1000 esp32-20190125-v1.10.bin`
- Firmware Releases:** A list of releases with dates and download links, including v1.19.1 (2022-06-18) as the latest.
- Firmware (Compiled with IDF 3.x):** A section for releases compiled with IDF 3.x.

F.4 Copio el comando para flashear el firmware, cambiando el puerto com de linux a COMX



The screenshot shows the MicroPython website for ESP32. It includes a navigation menu on the left, a header with vendor and source information, and a main content area with installation instructions and a list of firmware releases. The installation instructions section contains two code blocks. The first block shows the command to erase the flash, and the second block shows the command to write the firmware to the flash. The firmware releases section lists various versions with their dates and download links.

Vendor: Espressif
Features: BLE, WiFi
Source on GitHub: esp32/GENERIC
More info: Website

The following files are daily firmware for ESP32-based boards without external SPIRAM.
This firmware is compiled using ESP-IDF v4.x. Some older releases are also provided that are compiled with ESP-IDF v3.x.

Installation instructions

Program your board using the `esptool.py` program, found [here](#).

If you are putting MicroPython on your board for the first time then you should first erase the entire flash using:

```
esptool.py --chip esp32 --port /dev/ttyUSB0 erase_flash
```

From then on program the firmware starting at address 0x1000:

```
esptool.py --chip esp32 --port /dev/ttyUSB0 --baud 460800 write_flash -z 0x1000 esp32-20190125-v1.10.bin
```

Firmware

Releases

- v1.19.1 (2022-06-18) .bin [elf] [map] [Release notes] (latest)
- v1.19 (2022-01-17) .bin [elf] [map] [Release notes]
- v1.17 (2021-09-02) .bin [elf] [map] [Release notes]
- v1.16 (2021-06-23) .bin [elf] [map] [Release notes]
- v1.15 (2021-04-18) .bin [elf] [map] [Release notes]
- v1.14 (2021-02-02) .bin [elf] [map] [Release notes]
- v1.13 (2020-09-02) .bin [elf] [map] [Release notes]
- v1.12 (2019-12-20) .bin [elf] [map] [Release notes]

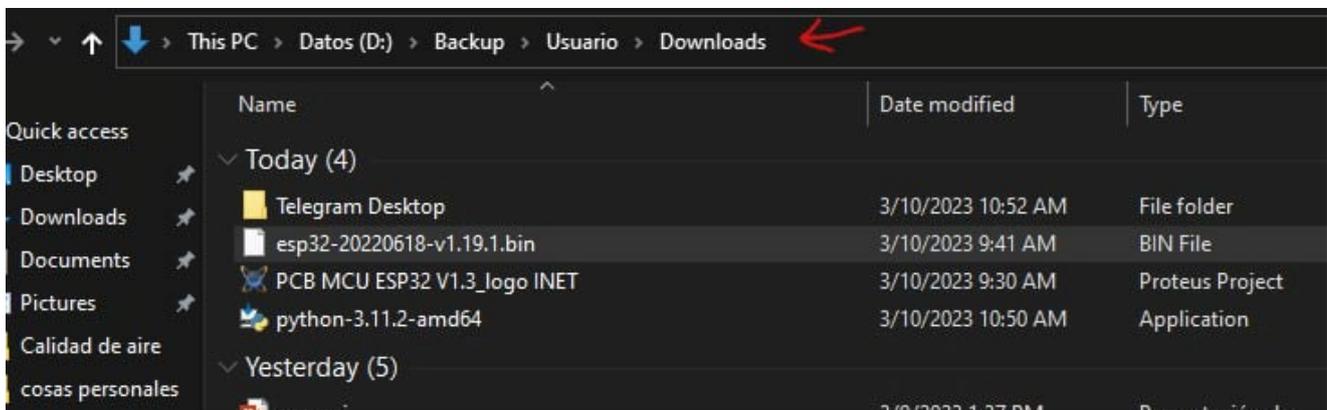
Nightly builds

- v1.19.1-962-g4376c969f (2023-03-10) .bin [elf] [map]
- v1.19.1-957-ged1f42cb4 (2023-03-09) .bin [elf] [map]
- v1.19.1-953-gb336b6b7 (2023-03-09) .bin [elf] [map]
- v1.19.1-941-gdc2f86c35 (2023-03-08) .bin [elf] [map]

Firmware (Compiled with IDF 3.x)

Releases

El firmware en este caso se guarda en la carpeta de descarga.

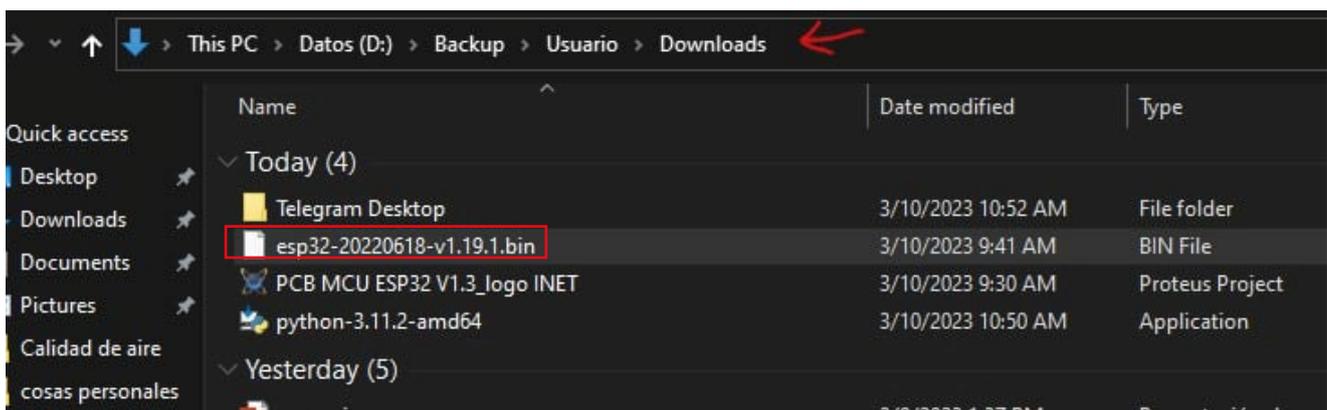


The screenshot shows a Windows File Explorer window with the path 'This PC > Datos (D:) > Backup > Usuario > Downloads'. A red arrow points to the 'Downloads' folder in the path. The main pane shows a list of files and folders:

Name	Date modified	Type
Today (4)		
Telegram Desktop	3/10/2023 10:52 AM	File folder
esp32-20220618-v1.19.1.bin	3/10/2023 9:41 AM	BIN File
PCB MCU ESP32 V1.3_logo INET	3/10/2023 9:30 AM	Proteus Project
python-3.11.2-amd64	3/10/2023 10:50 AM	Application
Yesterday (5)		
ana-agi	3/9/2023 1:27 PM	Presentación de...

F.5 Abrir una terminal en descargas

F.6 Copie el nombre del firmware descargado



The screenshot shows the same Windows File Explorer window as in the previous image. A red box highlights the file 'esp32-20220618-v1.19.1.bin' in the list. A red arrow points to the 'Downloads' folder in the path.

F.7 Coloque el comando copiado, colocando el puerto COM correspondiente y cambiando el nombre de firmware descargado. En este caso el comando permanece:

```
esptool.py --chip esp32 --port COMX --baud 460800 write_flash -z 0x1000 esp32-20220618-v1.19.1.bin
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS D:\Backup\Usuario\Downloads> esptool.py --chip esp32 --port COM9 --baud 460800 write_flash -z 0x1000 esp32-20220618-v1.19.1.bin
```

Entonces ejecutar

F.8 Comando ejecutado, KMI con micropython firmware cargado

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

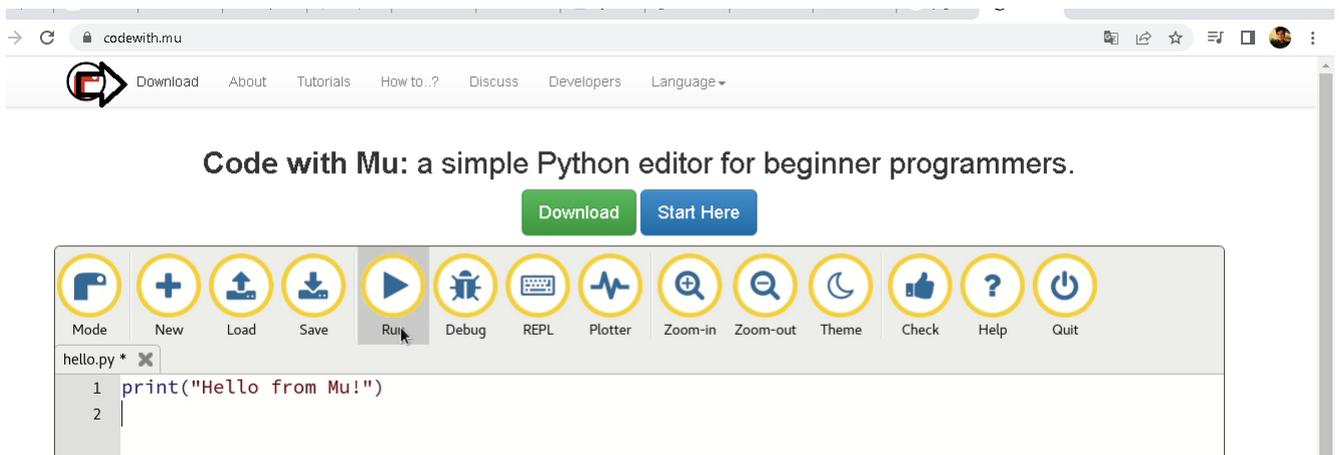
Try the new cross-platform PowerShell https://aka.ms/pscore6

PS D:\Backup\Usuario\Downloads> esptool.py --chip esp32 --port COM9 --baud 460800 write_flash -z 0x1000 esp32-20220618-v1.19.1.bin
esptool.py v4.5.1
Serial port COM9
Connecting....
Chip is ESP32-D0WDQ6-V3 (revision v3.0)
Features: WiFi, BT, Dual Core, 240MHz, VRef calibration in efuse, Coding Scheme None
Crystal is 40MHz
MAC: c8:c9:a3:f9:b1:90
Uploading stub...
Running stub...
Stub running...
Changing baud rate to 460800
Changed.
Configuring flash size...
Flash will be erased from 0x00001000 to 0x0017efff...
Compressed 1560976 bytes to 1029132...
Wrote 1560976 bytes (1029132 compressed) at 0x00001000 in 23.4 seconds (effective 534.3 kbit/s)...
Hash of data verified.

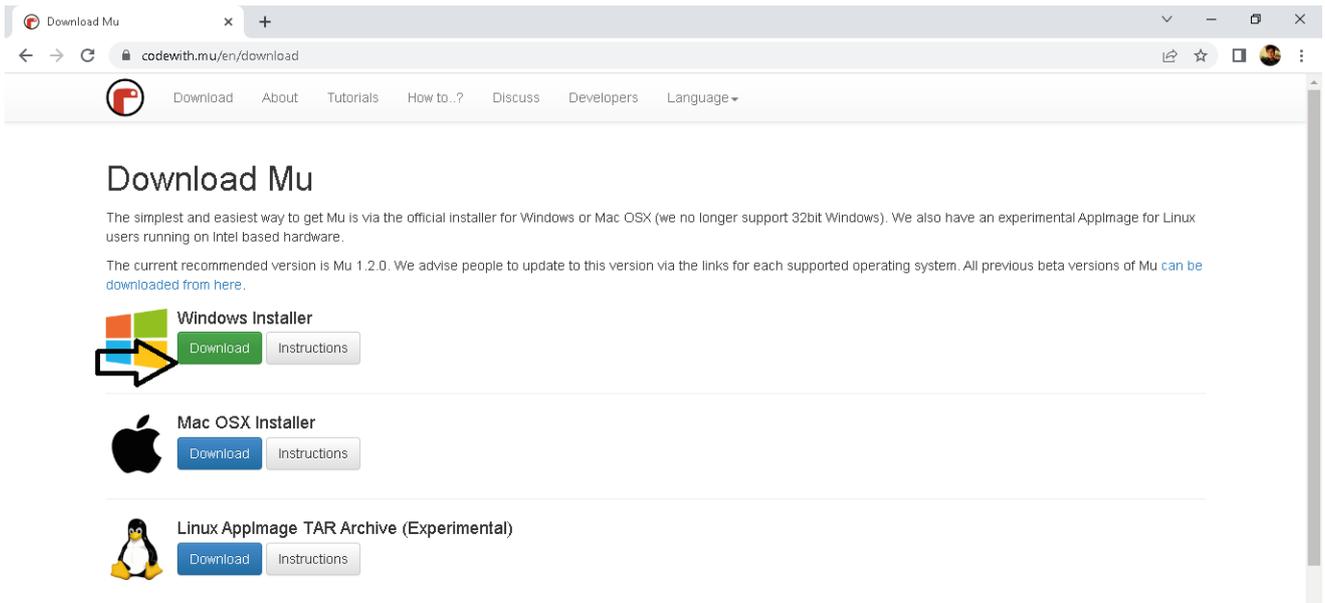
Leaving...
Hard resetting via RTS pin...
PS D:\Backup\Usuario\Downloads>
```

G) Pasos para programar en micropython:

G.1 Ir a la siguiente url: codewith.mu



G.2 Vaya a Descargar y descargue la versión para Windows.



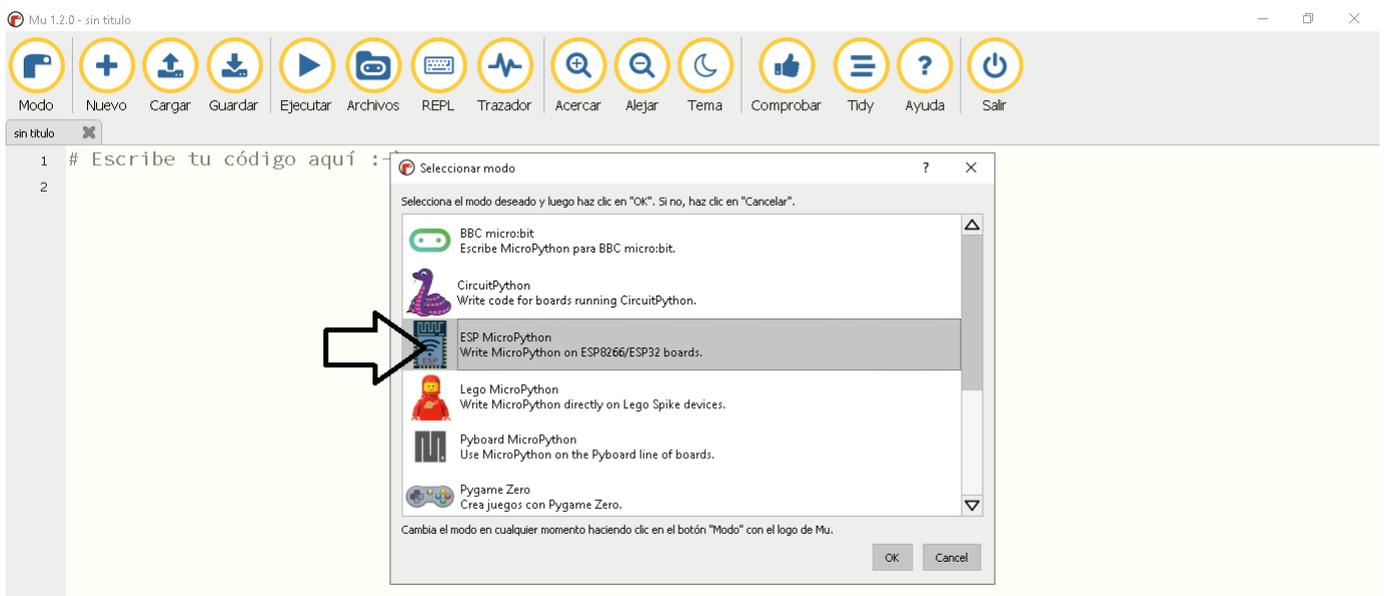
G.3 Abra el instalador, acepte los términos y presione instalar.



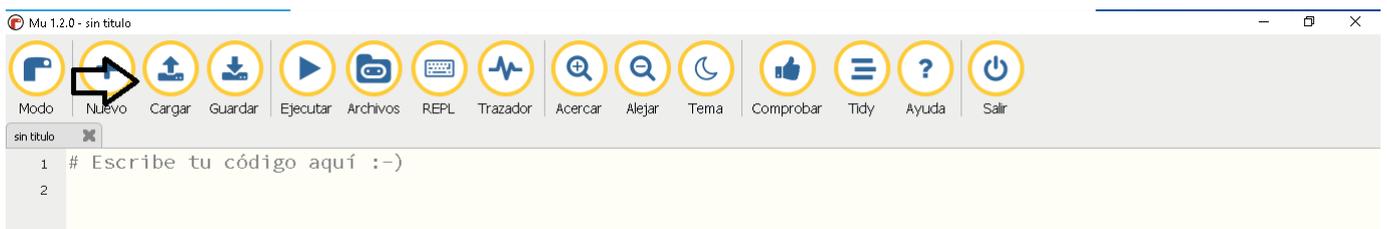
G.4 Abra el programa Mu, tardará un tiempo en ejecutarse.



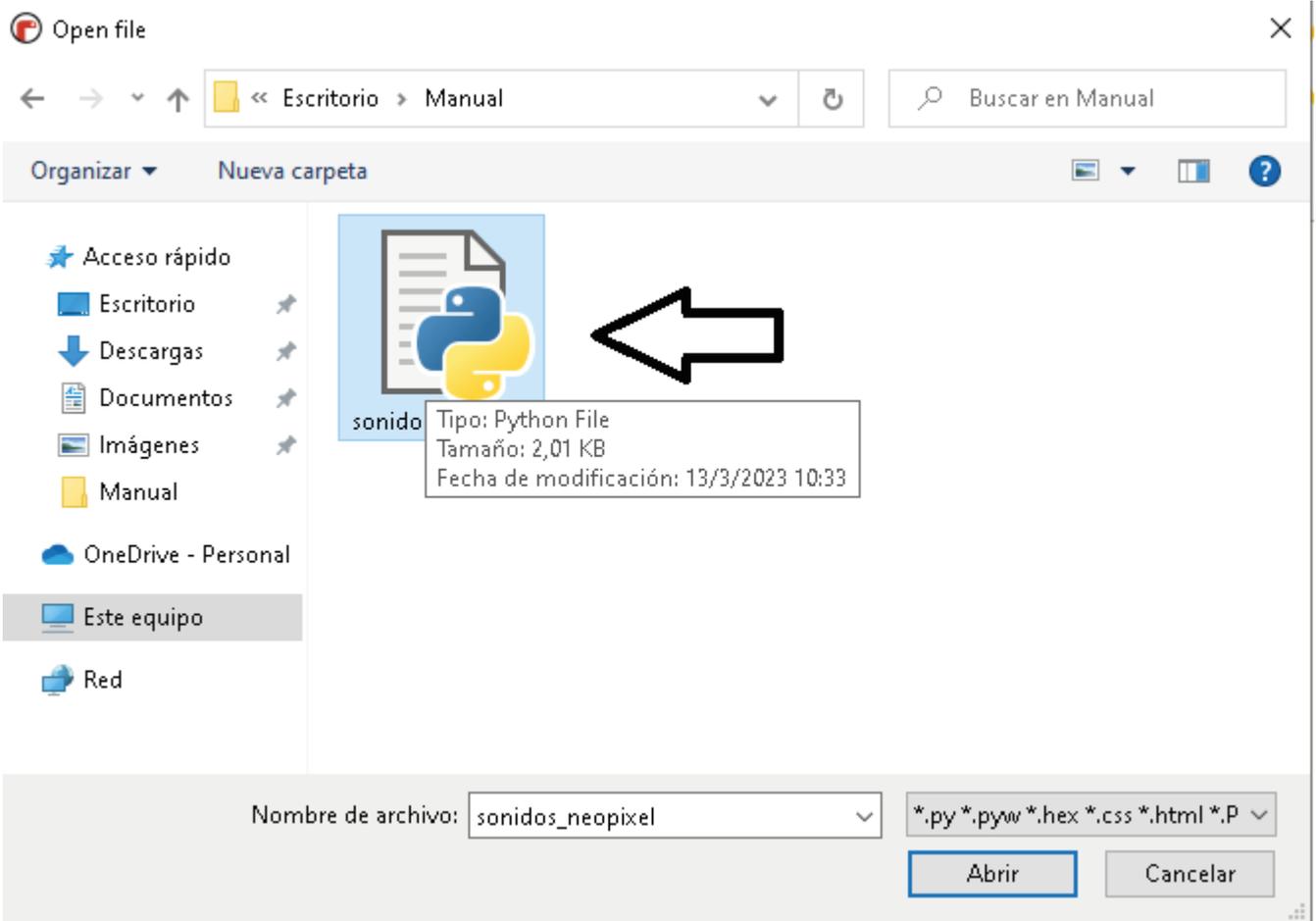
G.5 Seleccionar para programar con ESP MicroPython



G.6 Una vez abierto el programa, vaya a la opción de carga.



G.7 Abra el archivo python con el programa



G.8 Conecte el esp32.

G.9 Haga clic en ejecutar



G.9 Ahora el programa se ejecuta en tiempo real en el ESP32