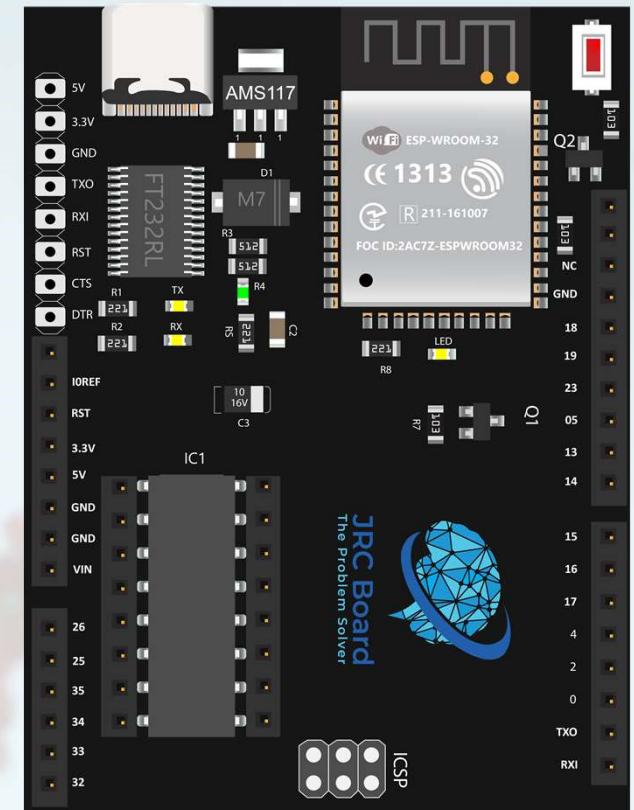
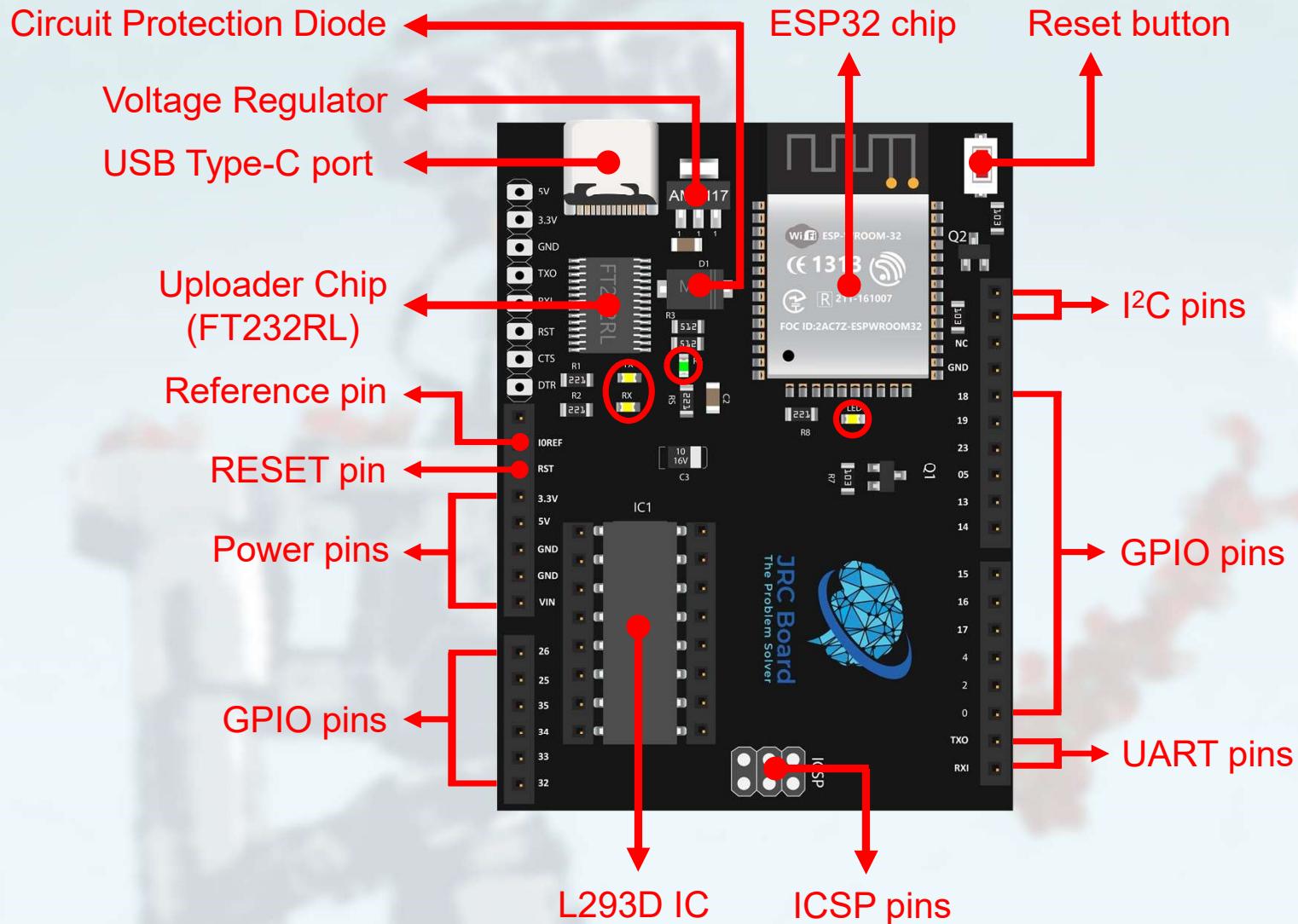




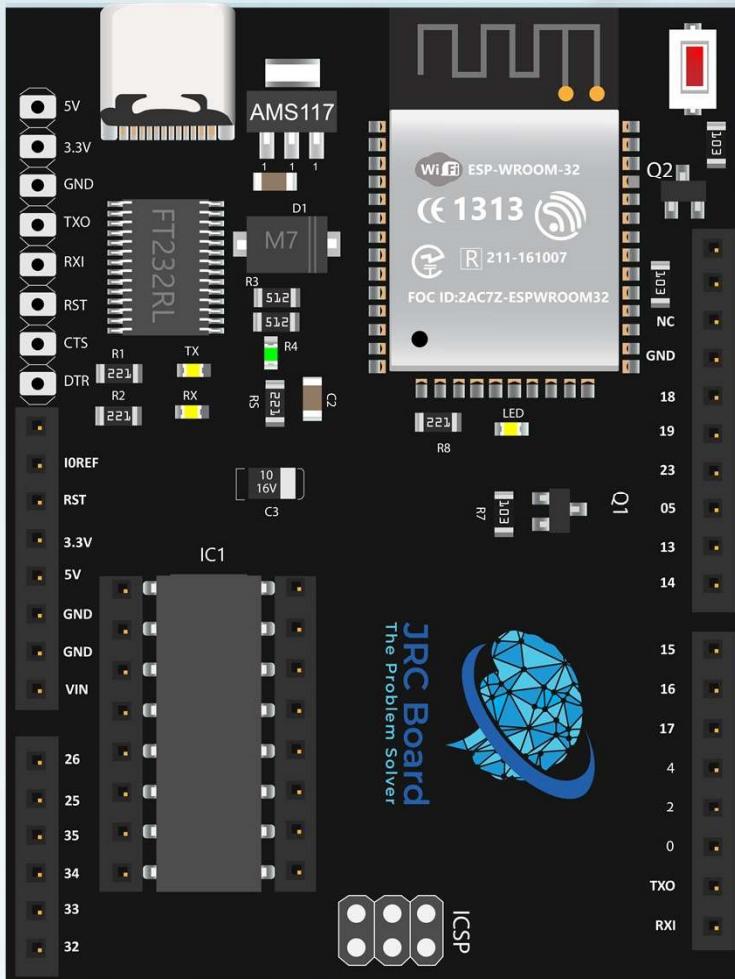
INTRODUCTION

JRC board is a microcontroller board based on the ESP-WROOM-32. It has 20 digital input/output pins of which pin 25 and 26 can be used as analog outputs, 12 analog inputs, an USB type 3 connection, an ICSP header and a reset button. It contains everything needed to support the microcontroller. Simply connect it to a computer with a USB cable or battery to get started. Keep Up with the World A Low-Cost Board for Schools in Bangladesh and Beyond. Powerful System on Chip. An In-House Development Board for our future products to blend in the Existing Ecosystem.





Specs and Features:



Processors:

- CPU: Xtensa Dual-Core 32-bit LX6 Microprocessor, operating at 240 MHz and performing up to 600 DMIPS.
- Ultra-Low Power (ULP) Co-Processor.

Memory:

- 520KB of RAM (expendable up to 8MB externally)
- 1024 locations for EEPROM
- 4MB of flash memory (expandable up to 16 MB externally)

Wireless connectivity:

- Wi-Fi: 802.11 b/g/n
- v4.2 BR/EDR and BLE

Peripheral Interfaces:

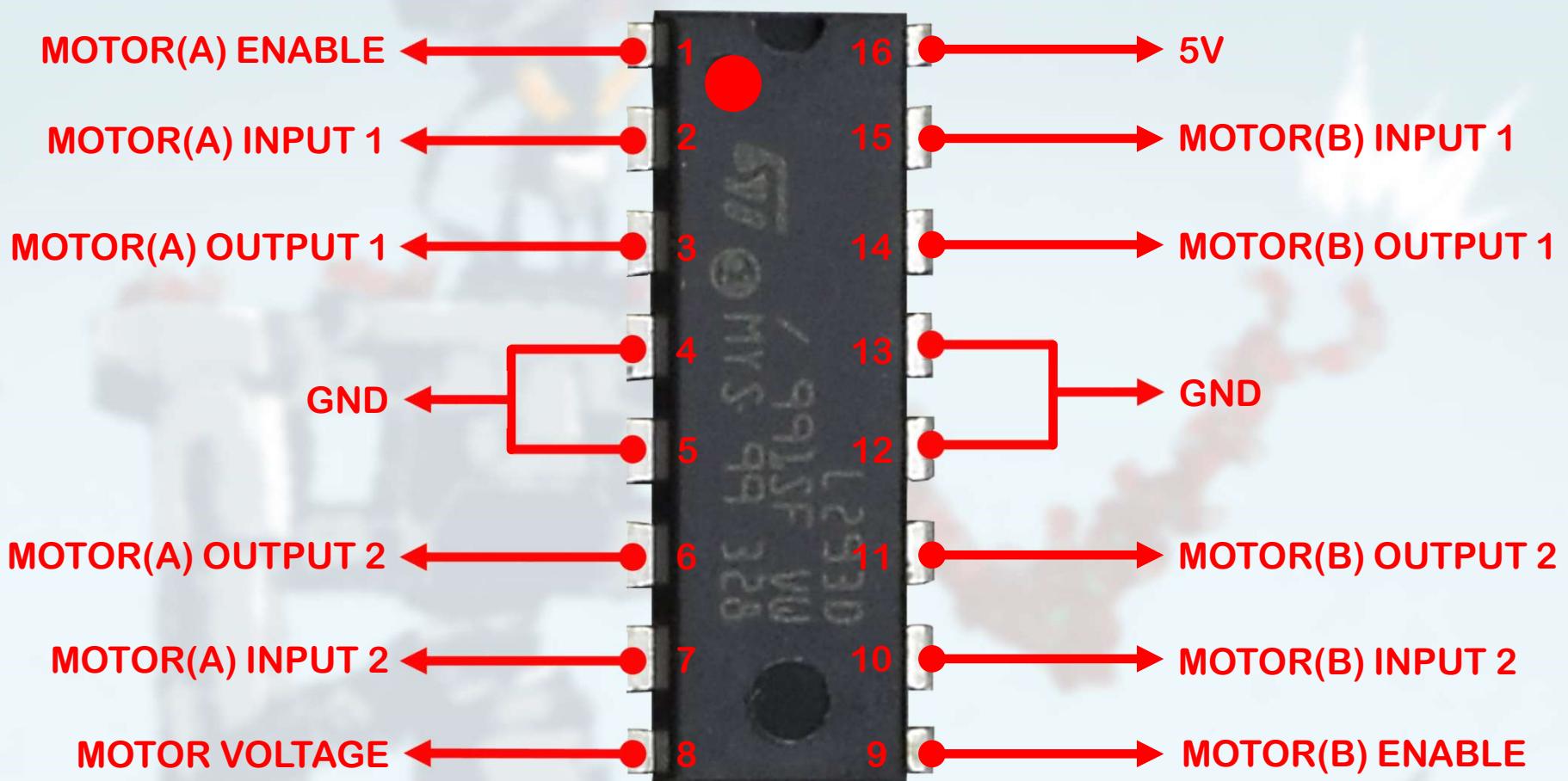
- 12-bit SAR ADC up to 18 channels
- 2 × 8-bit DACs (GPIO 25 and 26)
- 18 × PWM pins (Controlled by up to 16 channels)
- 7 × Touch Sensors (Capacitive Sensing GPIOs)
- 1 × SPI
- 1 × I2S Interfaces
- 1 × I2C Interfaces
- 2 × UART
- All digital pins compatible with external interrupt

Details about L293D:



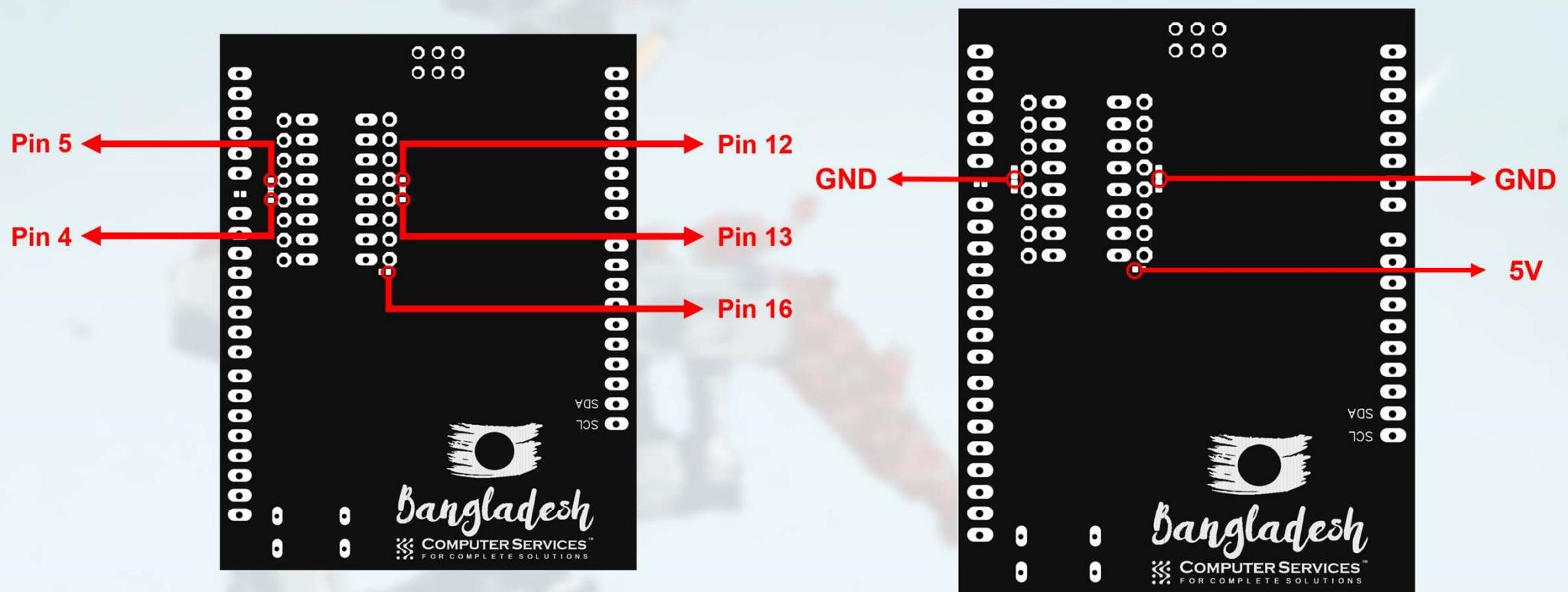
L92D is a Dual H-bridge motor driver circuit that can operate two motor simultaneously. Its maximum operating voltage is 16V and it can deliver maximum 600mA of current to each channel. Both channel can be combined to get better current output. Each channel has 3input for polarity control and voltage level control. We included this in our board for better power output system and convenience.

Details about L293D:

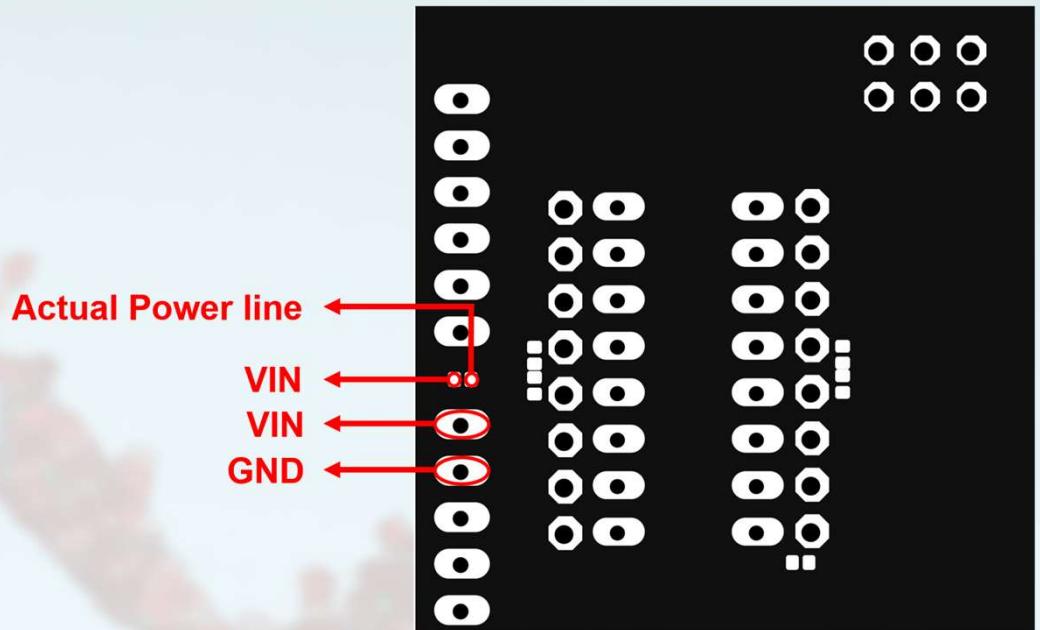


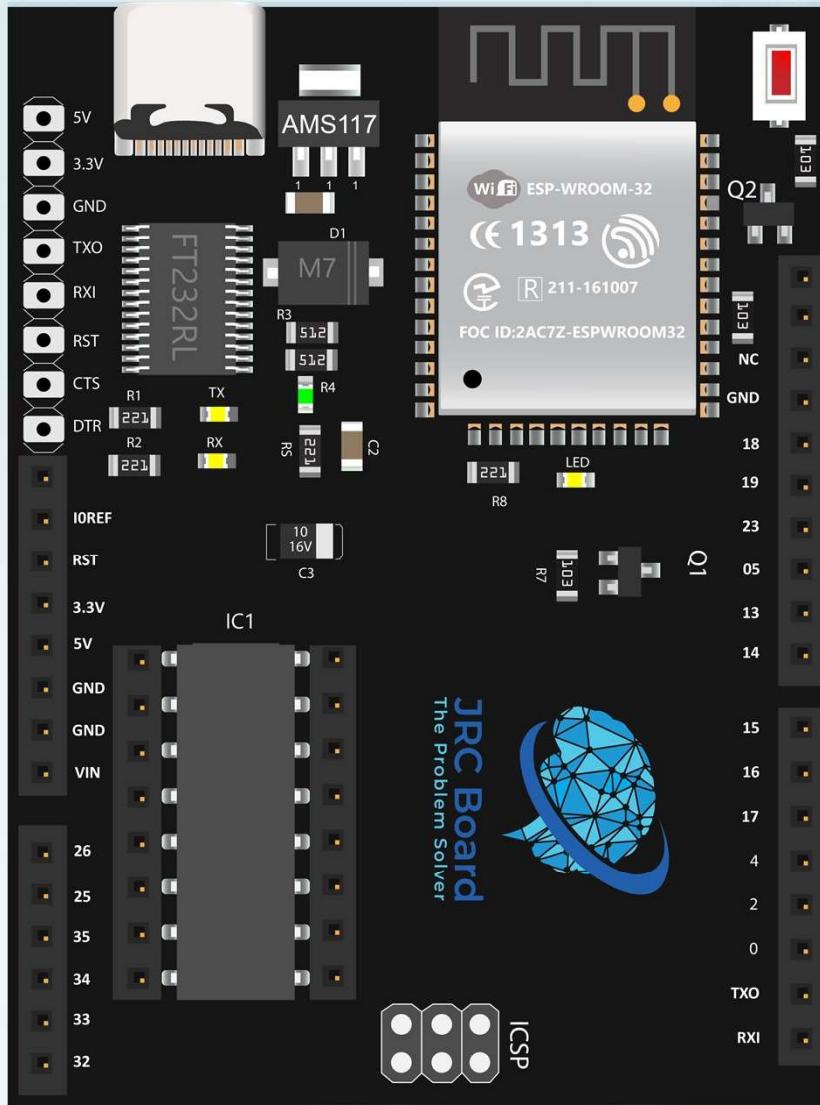
You will see 8 soldering pads marked on the back of the board. There are 4 soldering pads on each side. Nearly two of these are given nearby. The ground pin of the board is directly attached to these pads. And there were four pads at the four corners. Among them is the connection of pin number 4 ,5 ,12 ,13 of IC. If we solder and shorten the pads, these pins will be attached to the ground. This is because if we use the motor driver IC here, the chip that the pins have to be connected to ground. The work can be done only by welding the jumper wire without the hassle of connecting the top and the circuit is clearer to see. But if you use any other logic chip, you can easily remove the solder from this pad and continue working.

In addition to the previous 4 pairs of pads, there is another pair of pads in the right corner of the IC, one of which is connected to a five-volt pin on the pad board. If you connect the pad solder, you will get 5 volts in pin 16 of the chip.



If you connect the battery to the VIN at the beginning, you will see that the board is not working. The reason is that this one pair of solder pads has been kept open. We have kept the USB power line the same in this circuit, so if someone inadvertently connects to the computer while giving more than 5 volt power to the VIN PIN, the computer's USB port is likely to be damaged. So in the beginning it was kept open for safety. The connection is soldered by soldering, but if the battery is connected, the board will turn on





Key Features:

- Dual Core Low Power Chip with better clock frequency
- Built-in Wi-Fi & Bluetooth
- Built-in RTC module
- Internal Temperature and Hall sensor
- More Flash Memory & RAM
- Arduino Uno Compatible and shield available
- USB Type C port
- Built-in Motor Driver chip (L293D)
- DIP-16 socket for any other purpose
- Debugging LED
- 16 PWM channel Attachable to almost any pin
- Better PWM resolution and adjustable by program
- PWM frequency adjustable by program.
- 2 DAC output
- 12 analog input with better resolution
- Attach external interrupt on any GPIO pin you want
- Software Reset Compatible

Some precautions and tips:

- This is a 3.3V microcontroller board. Don't input more than 3.3V into any GPIO pins
- Wi-Fi and Bluetooth feature draw about 300mA on average. Regular USB port can not deliver that much of current. So you would better plugged in a external power source (battery) through VIN pin.
- Check if you soldered VIN pad before connecting any battery to VIN pin.
- It is okay to input up to 9V into VIN pin. But when you plug into pc by cable while keeping battery connected, the battery voltage directly effects the computer. So make sure if your USB port of pc is tolerant above 5V. Else, use buck converter to step down voltage into 5V
- Don't expect much current delivery from 5V and 3.3V pin. They actually come from FTDI chip which acts as uploader for ESP32. If you draw more than 40-50 mA from this pin, the chip resets immediately send reset signal to ESP32. So, the board will keep restarting if you draw much current.

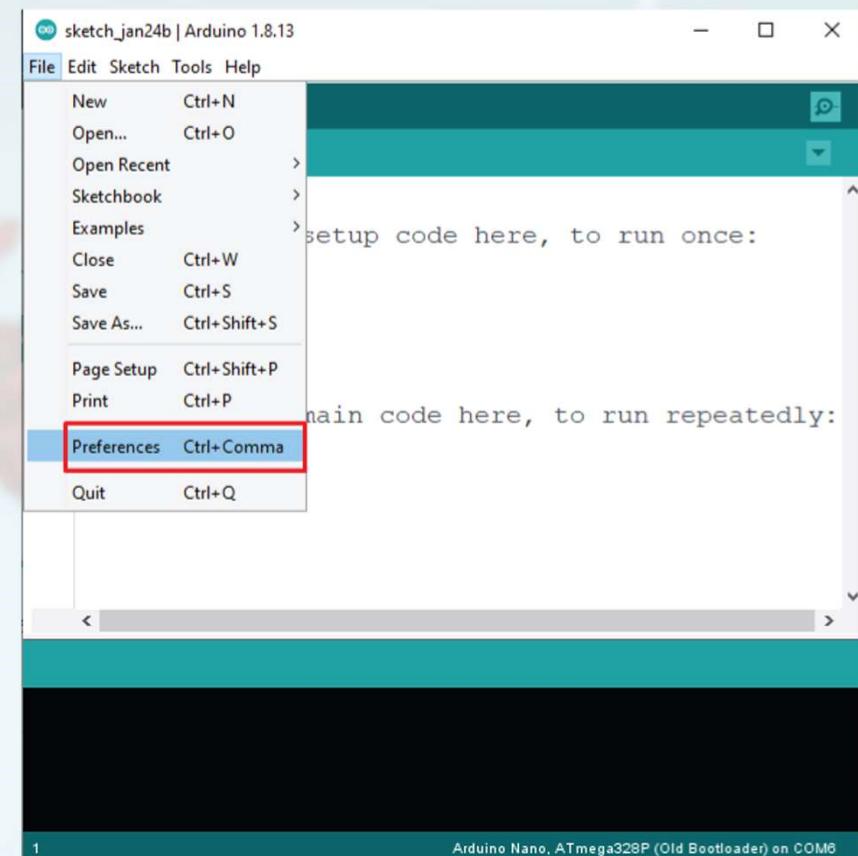
Preparing Board for Arduino IDE

- Open Arduino IDE on your pc
- Click on “file”, then “preferences”

The screenshot shows the Arduino IDE interface. The title bar reads "sketch_jan24b | Arduino 1.8.13". The menu bar has "File" highlighted with a red box. The code editor contains the following sketch:

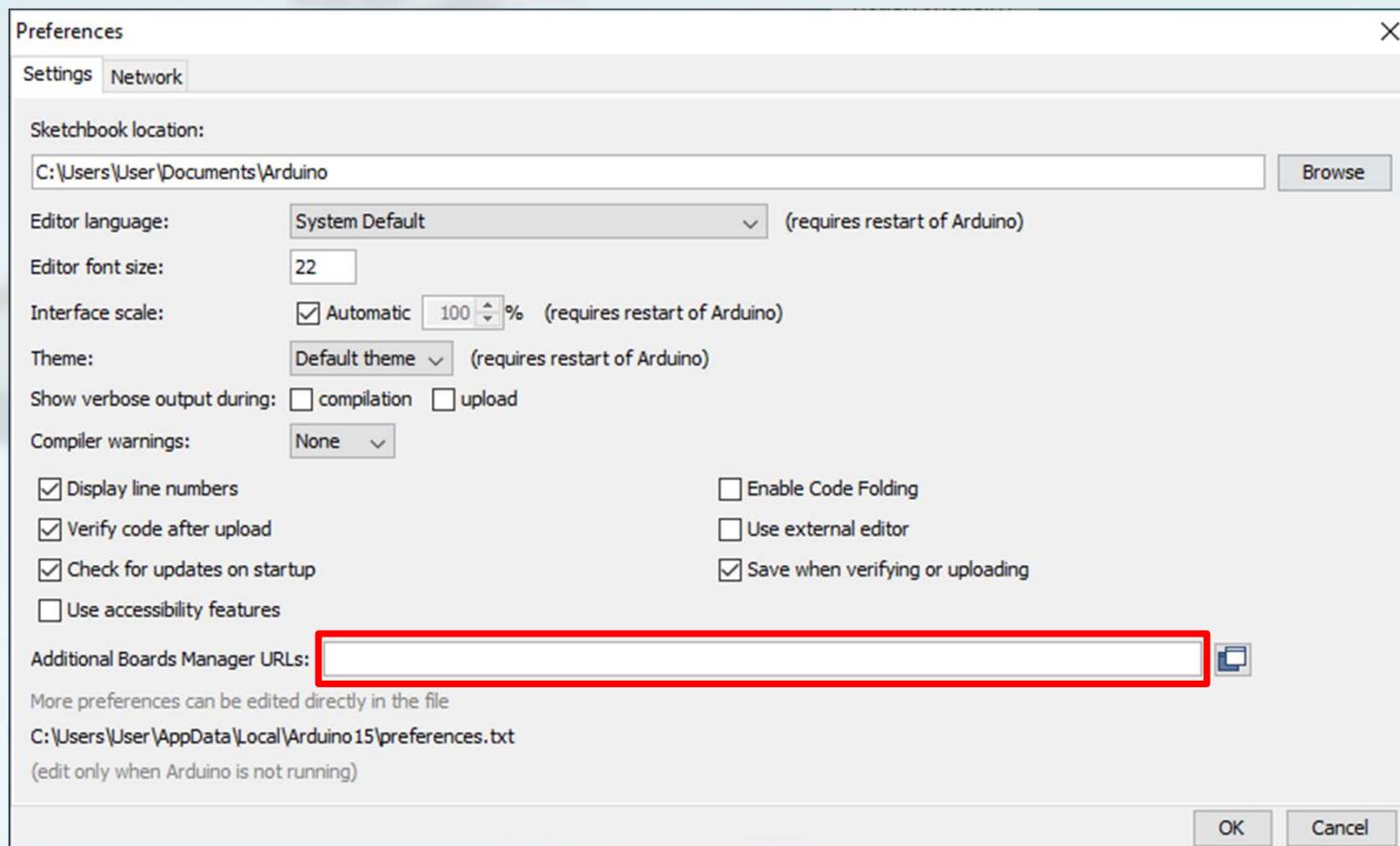
```
1 void setup() {  
2     // put your setup code here, to run once:  
3  
4 }  
5  
6 void loop() {  
7     // put your main code here, to run repeatedly:  
8  
9 }
```

The status bar at the bottom indicates "Arduino Nano, ATmega328P (Old Bootloader) on COM6".



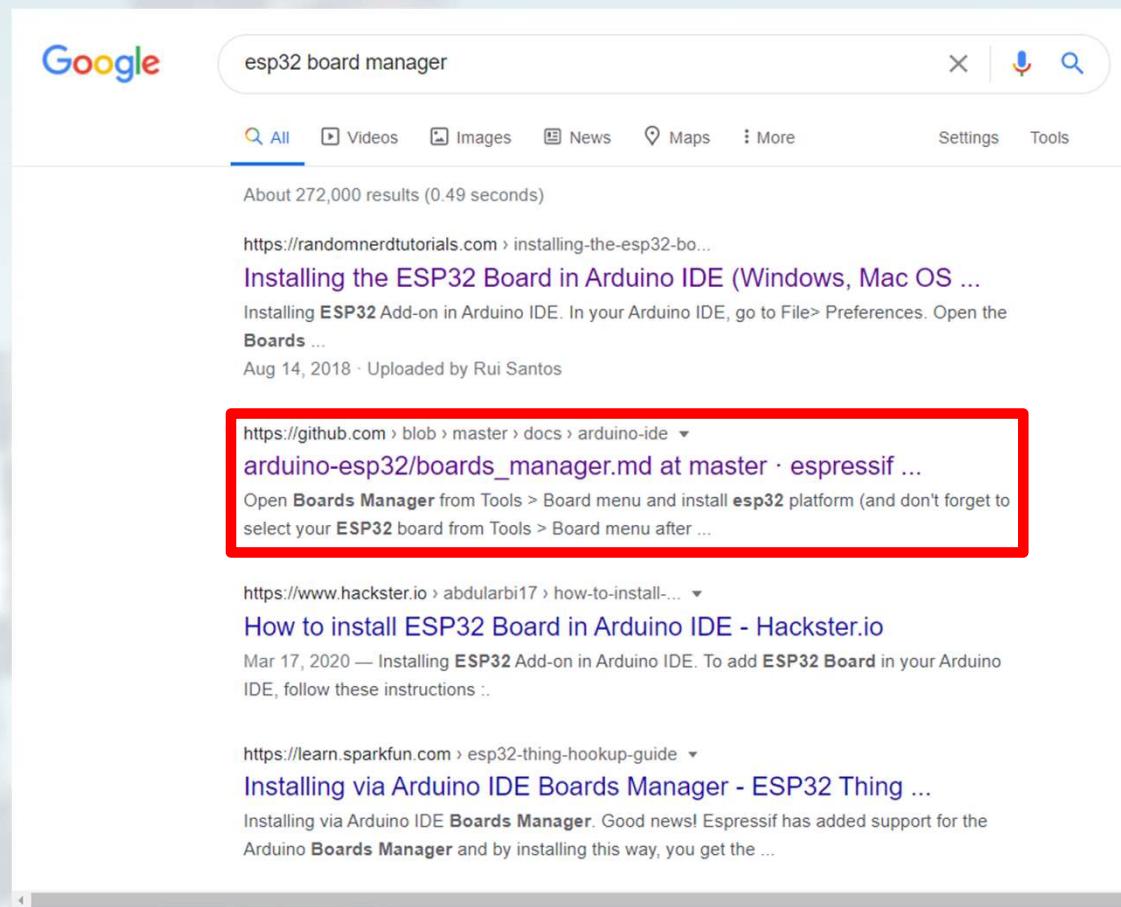
Preparing Board for Arduino IDE

- A window will pop up like this:



Preparing Board for Arduino IDE

- To get the link, search for “**ESP32 board manager**” in google.



Preparing Board for Arduino IDE

- A window will pop out like this:
- Copy the link from marked area

The screenshot shows a GitHub repository page for `espressif/arduino-esp32`. The page includes navigation links for Why GitHub?, Team, Enterprise, Explore, Marketplace, and Pricing. It features a search bar, a sign-in button, and a sign-up button. The repository has 207 issues, 51 pull requests, 1 action, 1 project, security information, and insights. The `boards_manager.md` file is selected in the code tab. The file content is as follows:

```
Installation instructions using Arduino IDE Boards Manager

=====


- Stable release link: https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package\_esp32\_index.json
- Development release link: https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package\_esp32\_dev\_index.json



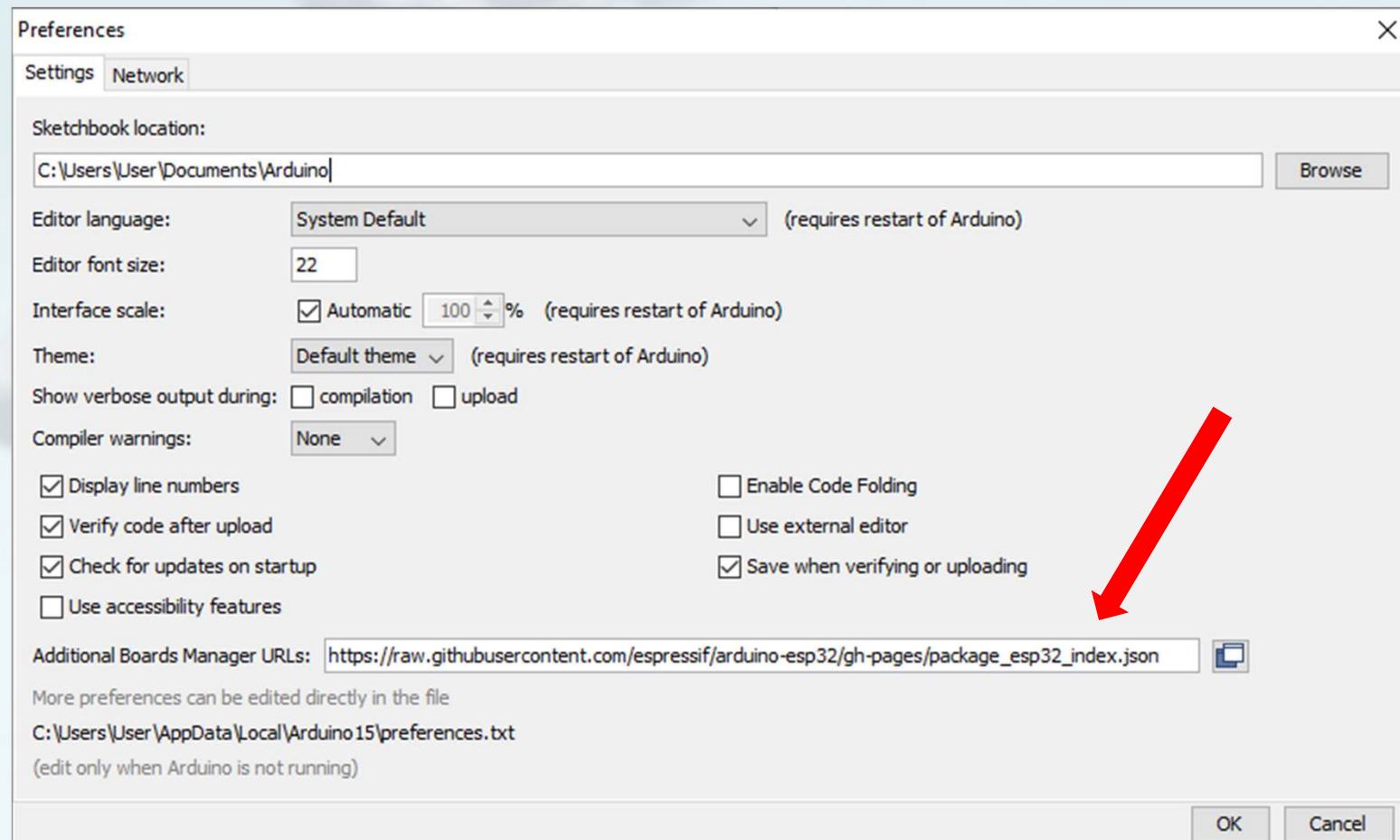
Starting with 1.6.4, Arduino allows installation of third-party platform packages using Boards Manager. We have packages available for Windows, Mac OS, and Linux (x86, amd64, armhf and arm64).


- Install the current upstream Arduino IDE at the 1.8 level or later. The current version is at the Arduino website.
- Start Arduino and open Preferences window.

```

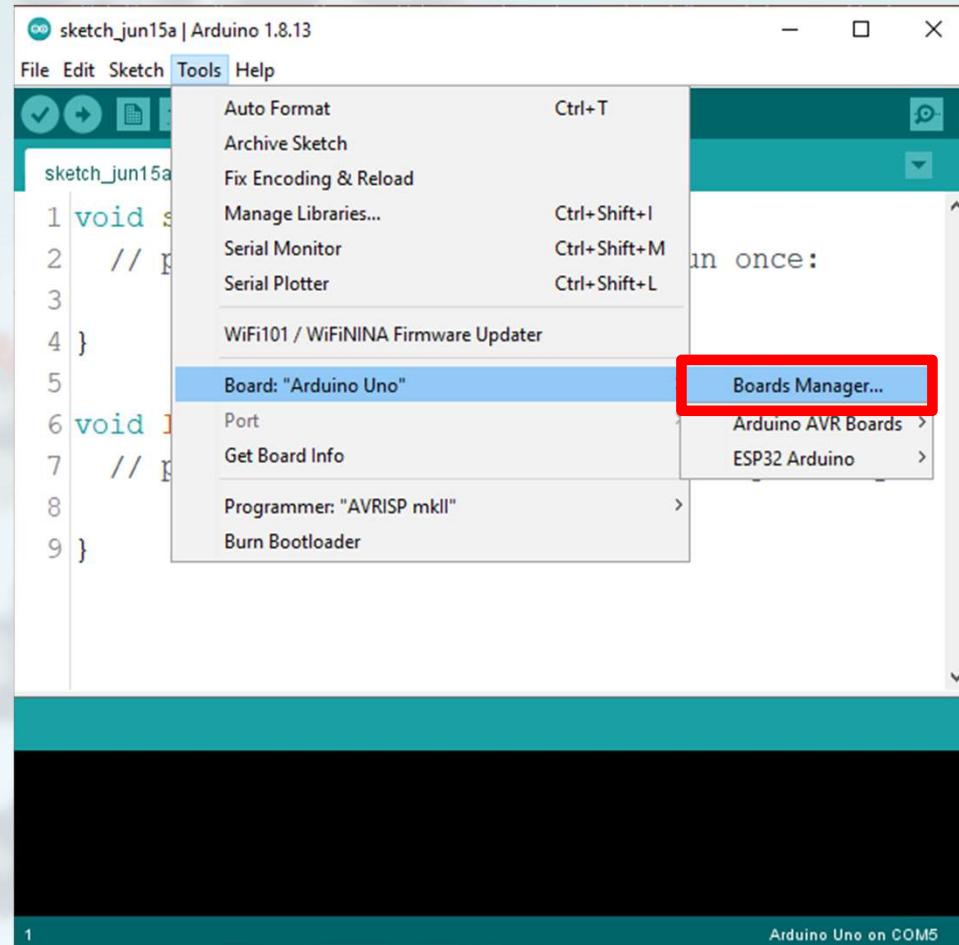
Preparing Board for Arduino IDE

- Paste the text on previously opened window, then select okay.



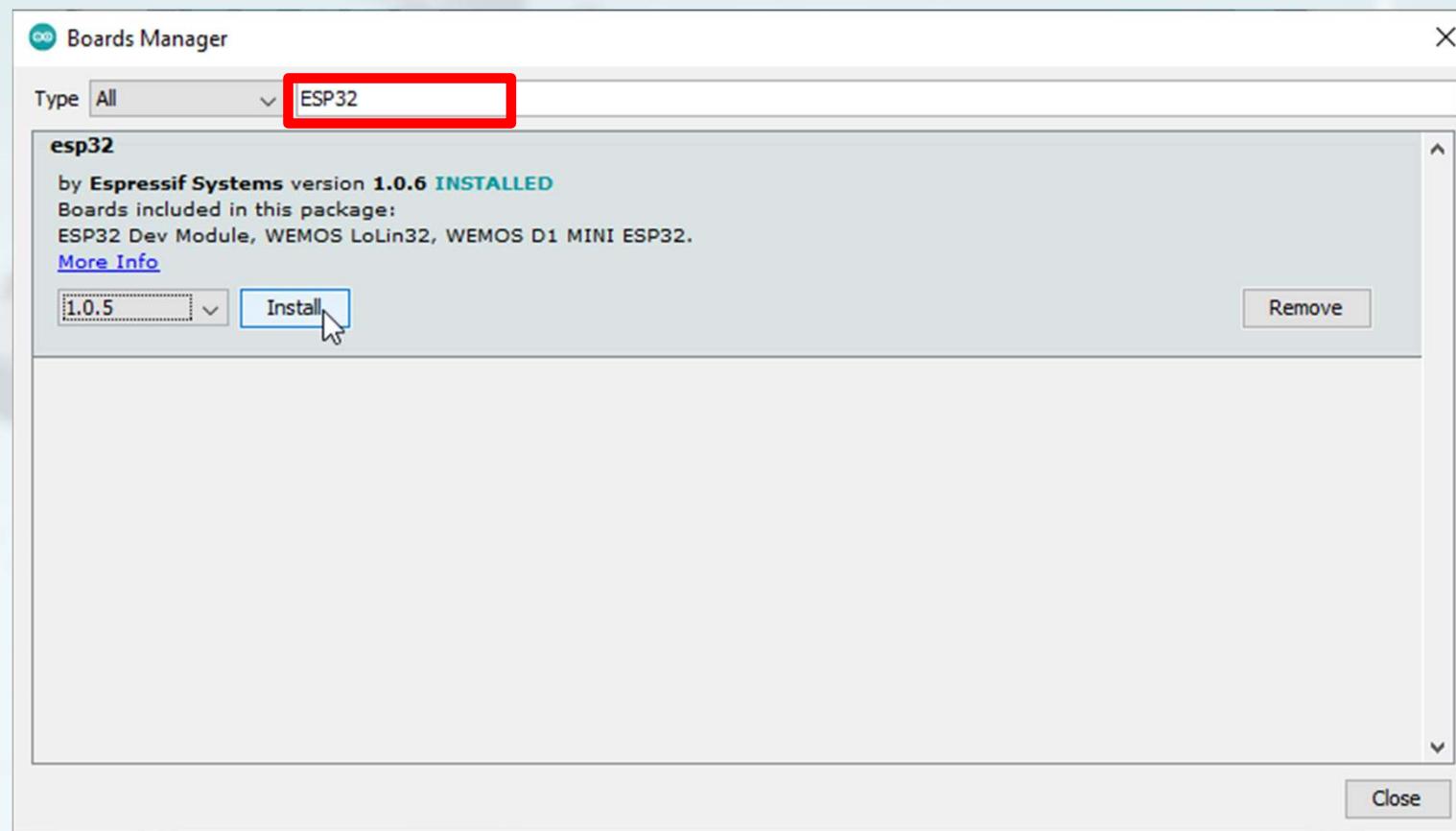
Preparing Board for Arduino IDE

- Click on “Tools”, then “Boards”, then “Boards Manager”



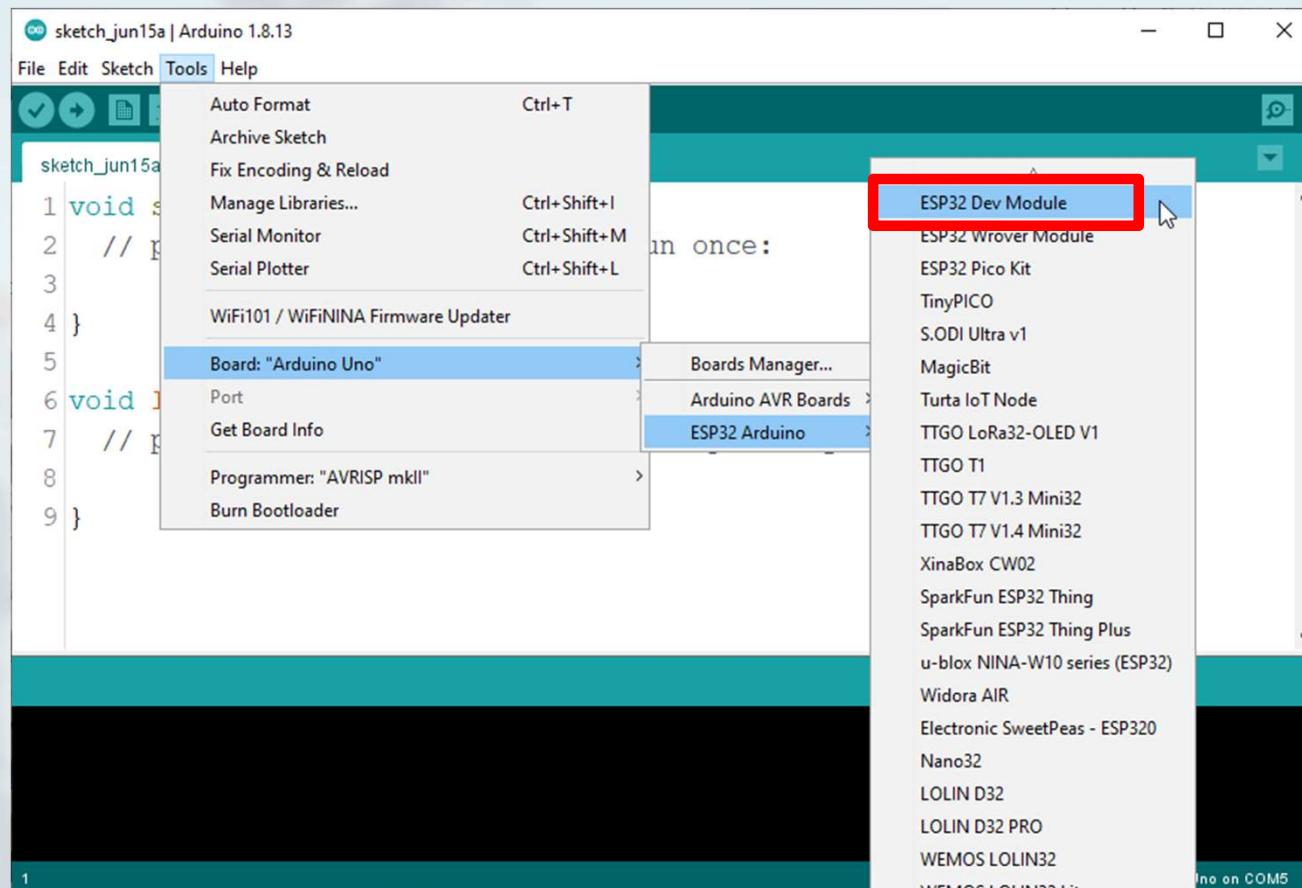
Preparing Board for Arduino IDE

- The Board manager window will open
- Search for “ESP32” in search bar. Select the latest version and install it



Preparing Board for Arduino IDE

- ESP32 Board Manager has been installed. Now you will have to select the right board
- Click on “Tools”, then “Boards”, then “ESP32 Arduino”, then “ESP32 Dev module”. Rest will be the same.



Preparing Driver for JRC Board

- Search “FT232RL Driver” on google. Click on “VCP Drivers”. That will bring you to a website.

A screenshot of a Google search results page for "FT232RL Driver". The search bar at the top contains the query. Below it, the "All" tab is selected, along with other options like Videos, Images, News, and More. A message indicates there are about 440,000 results found in 0.46 seconds. The first result is highlighted with a red box and shows a link to "https://ftdichip.com > Drivers". The title of this result is "VCP Drivers - FTDI". The snippet below the title explains that VCP drivers allow the USB device to appear as an additional COM port. The second result, which is not highlighted, shows a link to "https://ftdichip.com > drivers" and the title "Drivers - FTDI". It describes the Windows Driver Installer (VCP & D2XX) and lists supported operating systems. The third result shows a link to "https://www.usb-drivers.org > USB to Serial" and the title "FT232R USB UART Driver | USB Driver". It provides a download link for the driver and instructions for disconnecting and reconnecting the USB cable.

FT232RL Driver

All Videos Images News More Settings Tools

About 440,000 results (0.46 seconds)

<https://ftdichip.com > Drivers>

VCP Drivers - FTDI

Virtual COM port (VCP) **drivers** cause the USB device to appear as an additional COM port available to the PC. Application software can access the USB device ...

<https://ftdichip.com > drivers>

Drivers - FTDI

Windows **Driver** Installer (VCP & D2XX), please click here. **Drivers** are available which allow FTDI devices to work with the following operating systems: Windows ...

[VCP · D2XX Drivers · D3XX Drivers](#)

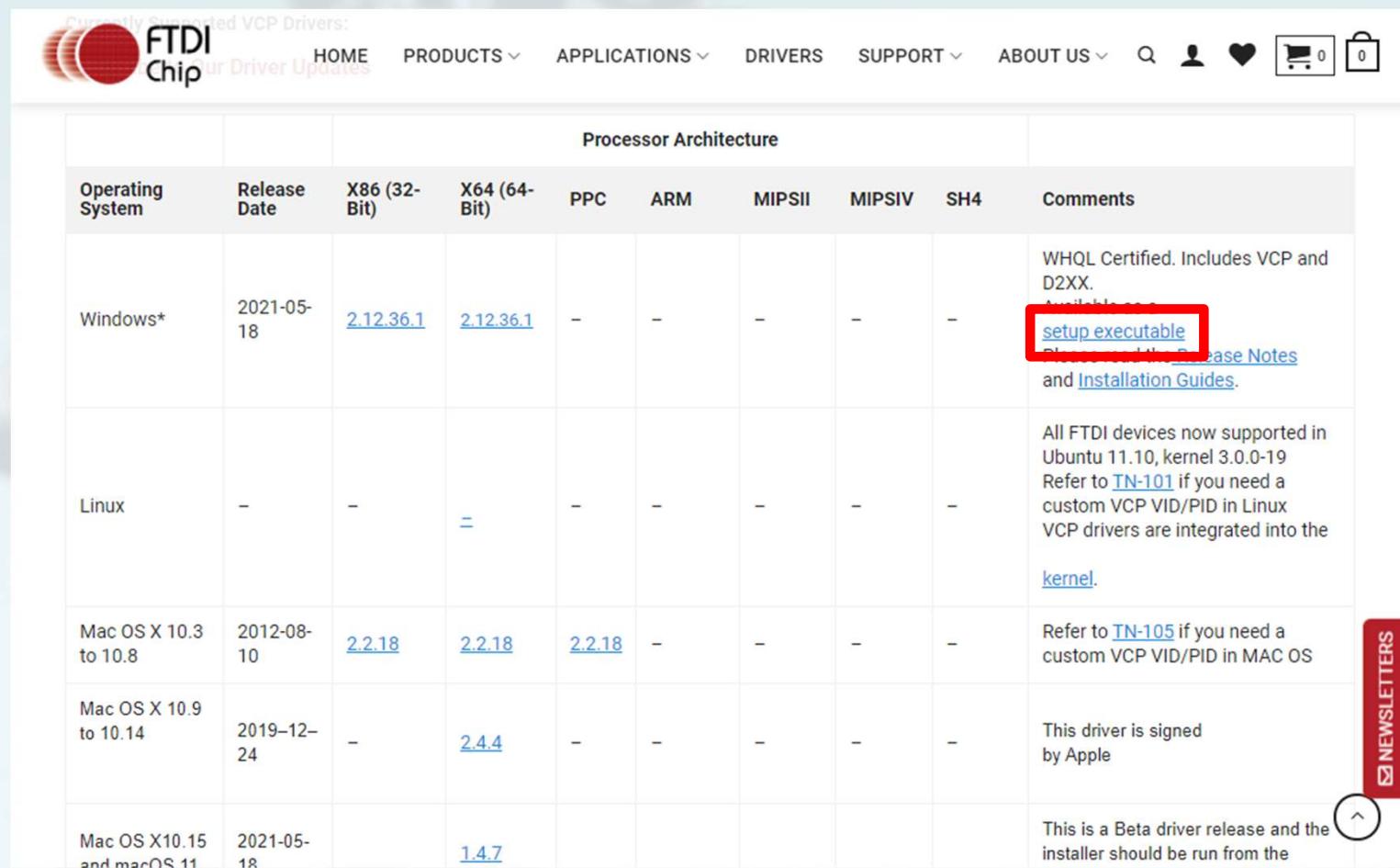
<https://www.usb-drivers.org > USB to Serial>

FT232R USB UART Driver | USB Driver

Apr 24, 2017 — **FT232R USB UART Driver** Download · Disconnect the USB cable from your PC to KMX1 or KMX2 · reconnect the USB cable from your PC to ...

Preparing Driver for JRC Board

- Scroll down a little until you find this section. Click on marked area to download a zip file

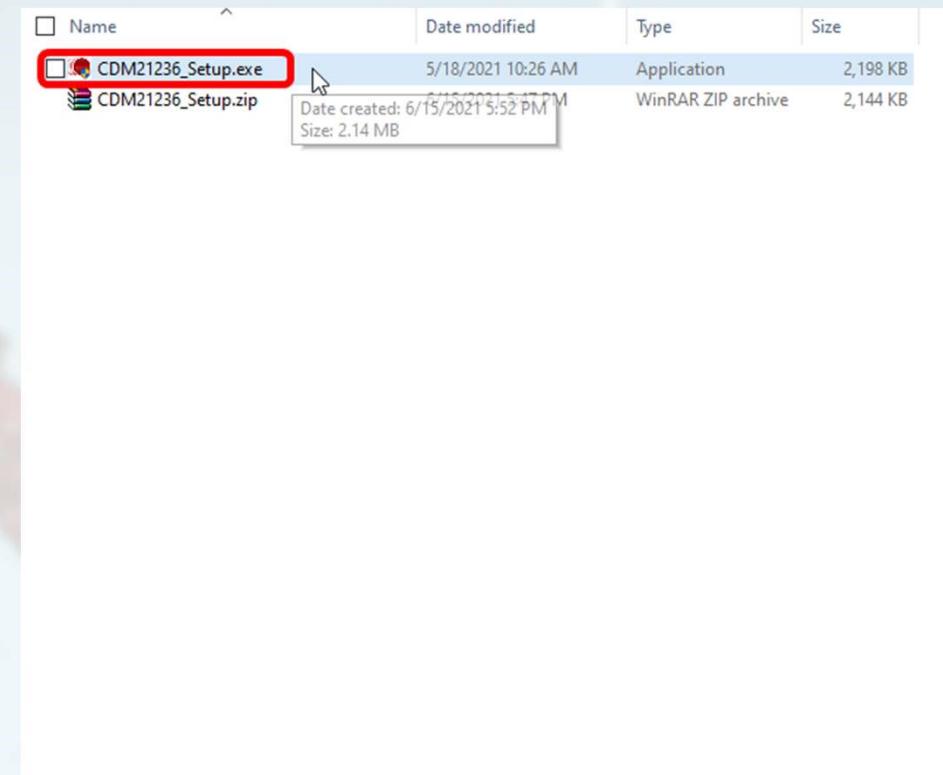
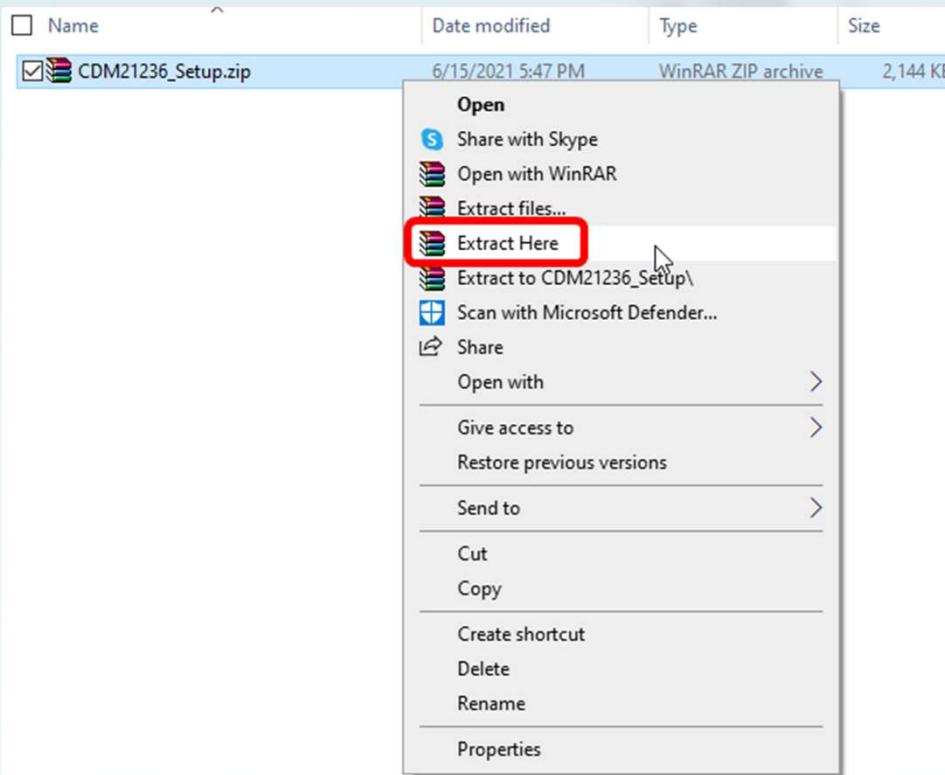


Currently Downloaded VCP Drivers:

Processor Architecture									
Operating System	Release Date	X86 (32-Bit)	X64 (64-Bit)	PPC	ARM	MIPSII	MIPSIV	SH4	Comments
Windows*	2021-05-18	2.12.36.1	2.12.36.1	-	-	-	-	-	WHQL Certified. Includes VCP and D2XX. Available here . setup executable Please see FAQ , Release Notes and Installation Guides .
Linux	-	-	-	-	-	-	-	-	All FTDI devices now supported in Ubuntu 11.10, kernel 3.0.0-19 Refer to TN-101 if you need a custom VCP VID/PID in Linux VCP drivers are integrated into the kernel .
Mac OS X 10.3 to 10.8	2012-08-10	2.2.18	2.2.18	2.2.18	-	-	-	-	Refer to TN-105 if you need a custom VCP VID/PID in MAC OS
Mac OS X 10.9 to 10.14	2019-12-24	-	2.4.4	-	-	-	-	-	This driver is signed by Apple
Mac OS X10.15 and macOS 11	2021-05-18		1.4.7						This is a Beta driver release and the installer should be run from the

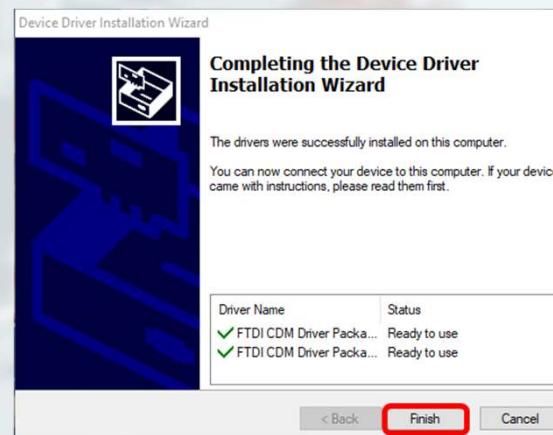
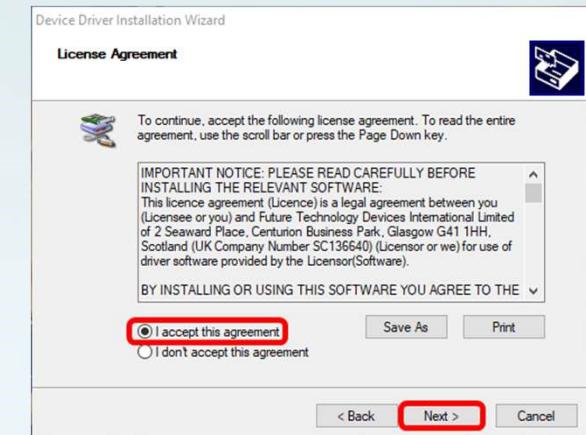
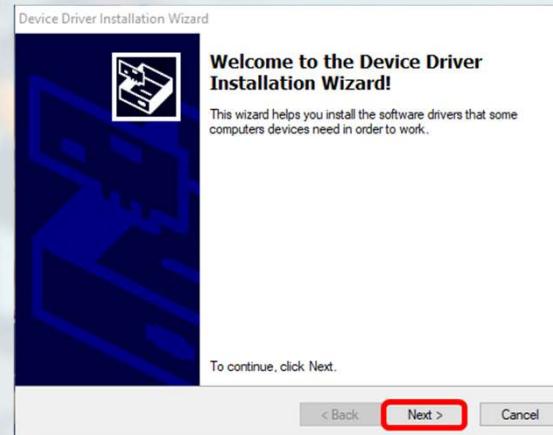
Preparing Driver for JRC Board

- Extract that zip file to find a executable file. Double click on that to install driver for JRC Board.



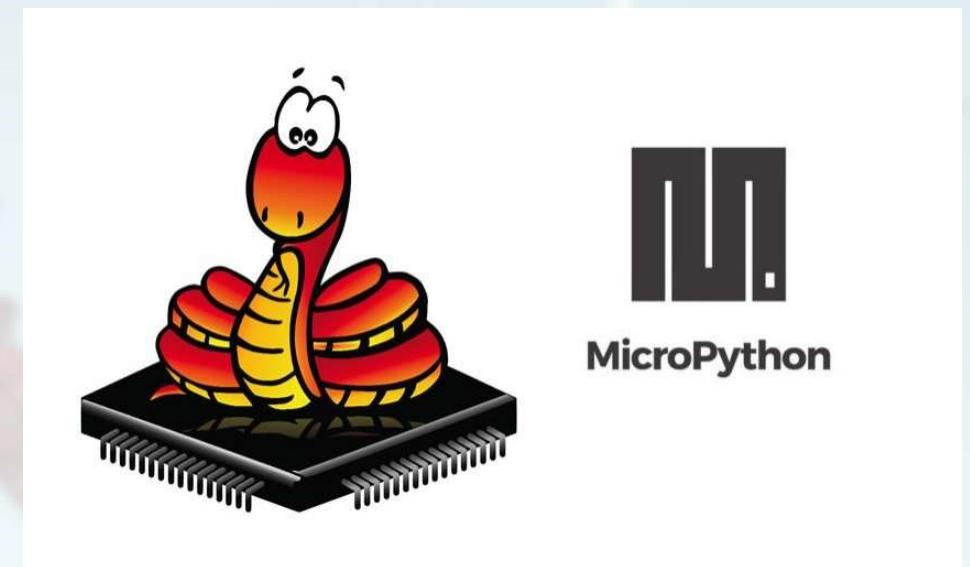
Preparing Driver for JRC Board

- After completing installation, you will be able to upload any program to JRC Board



Micro-Python and ESP32

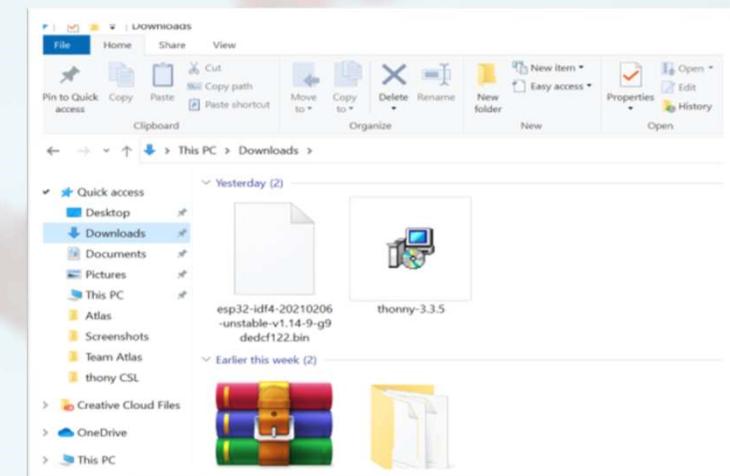
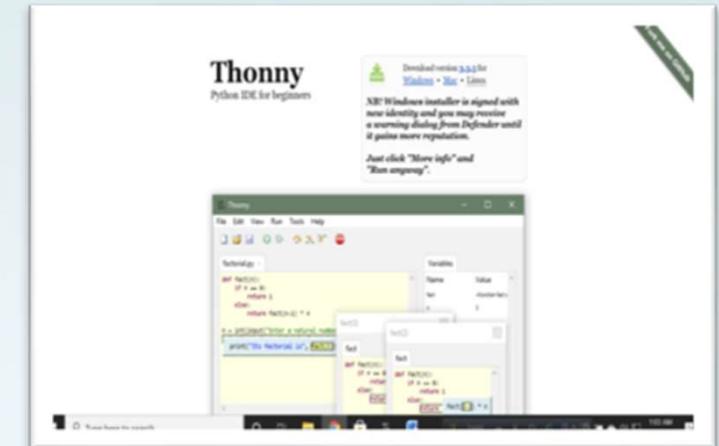
Micro-Python is a tiny open source Python programming language interpreter that runs on small embedded development boards. With Micro-Python you can write clean and simple Python code to control hardware instead of having to use complex low-level languages like C or C++ (what Arduino uses for programming).



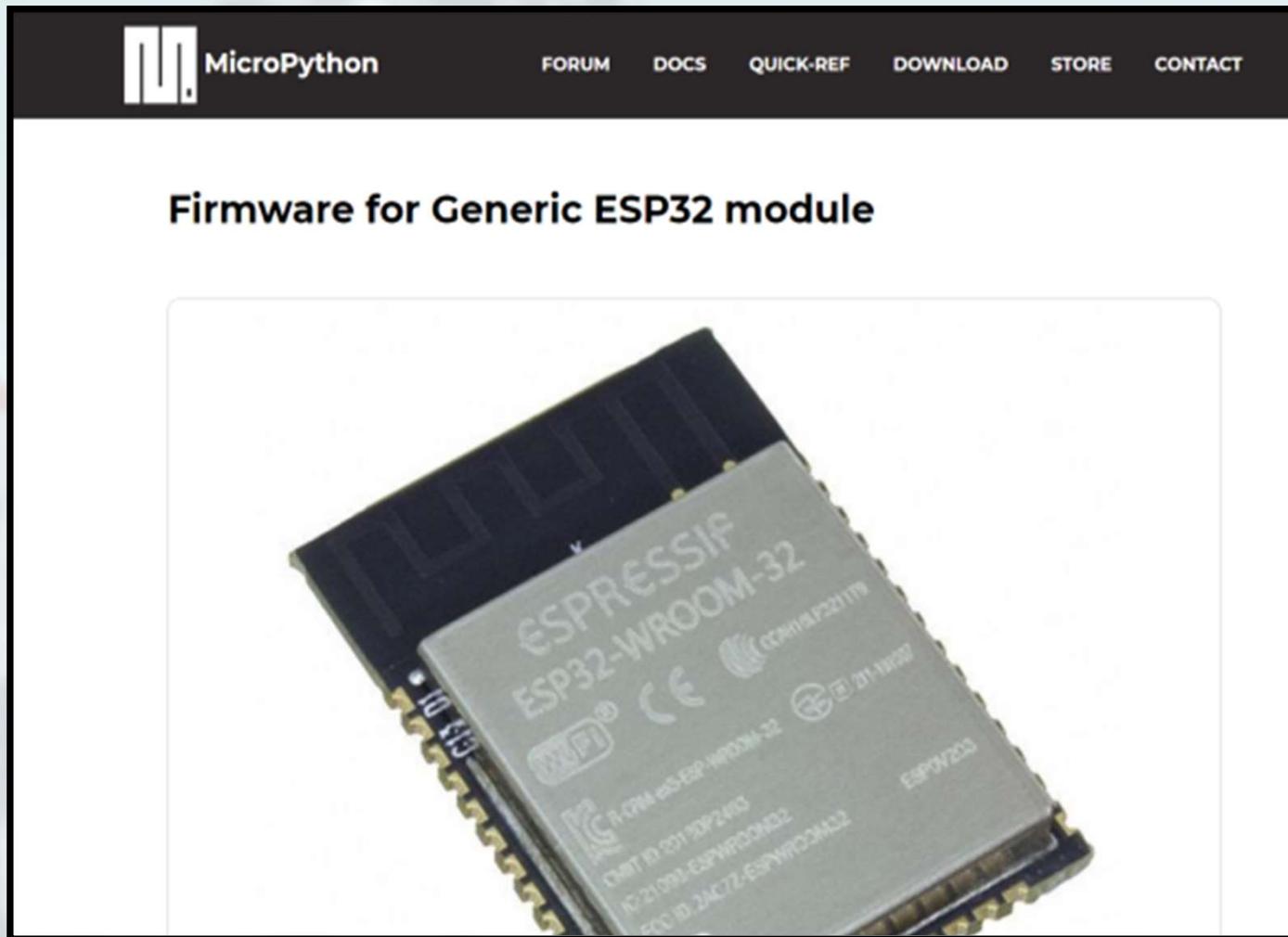
MicroPython and ESP32

The first thing to do with Micro Python is to install a compiler. There is an IDE software called “Thonny” through which the code made with Micro Python can be compiled and input into the microcontrollers. Here's how to download - .

1. First go to the “Thonny.org” website. At the beginning you can see that there are several operating systems including the download version. If you select the operating system as needed, the download process will begin.



Then you have to download - Firmware with ESP-IDF. To download it, go to "micropython.org"



There you have to go to the download option and select the **ESP32** device. Then you have to download a version from Firmware with ESP-IDF inside.

```
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
```

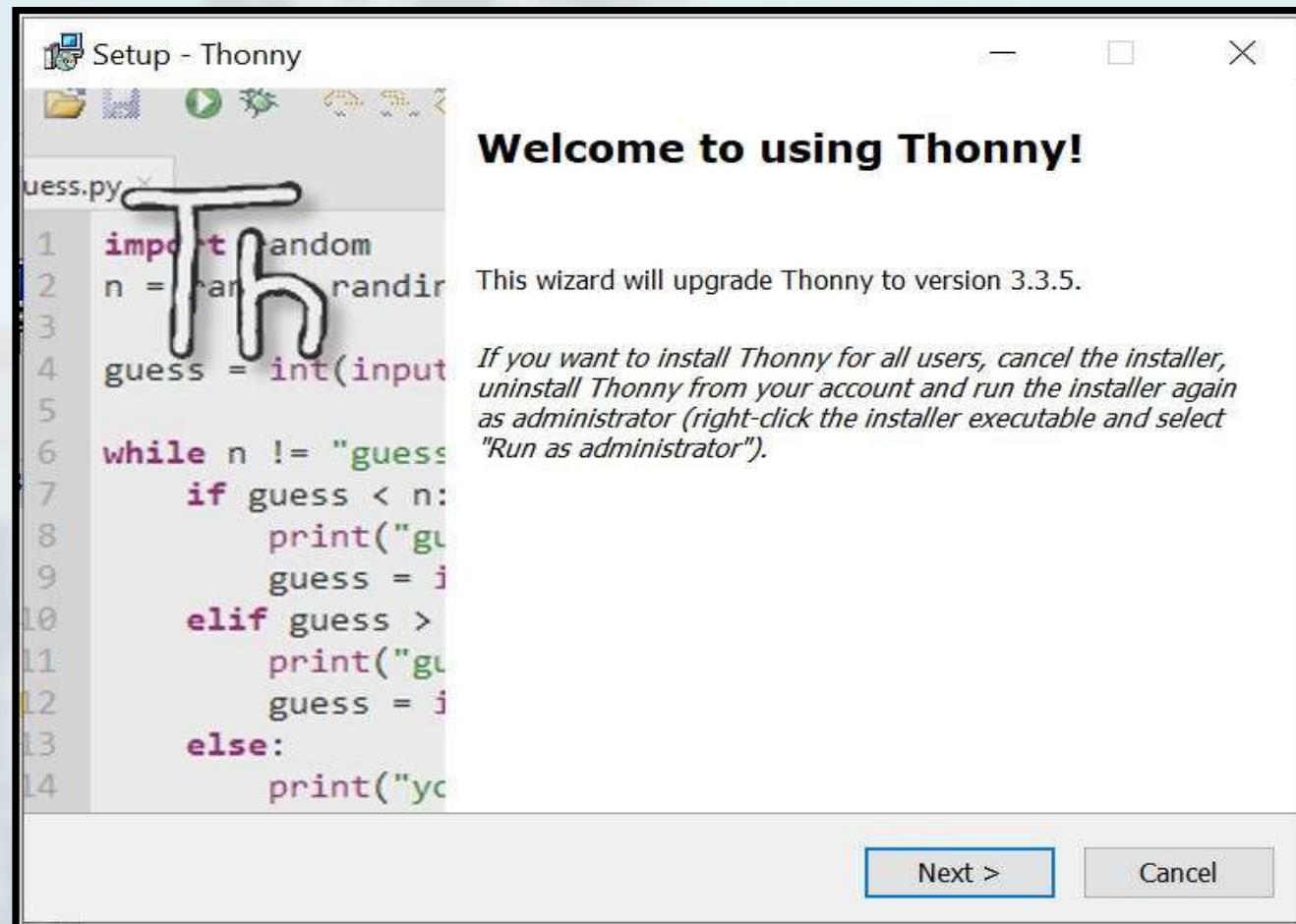
Firmware is provided using either ESP-IDF v3.x or v4.x. If in doubt use v4.x.

Firmware with ESP-IDF v4.x

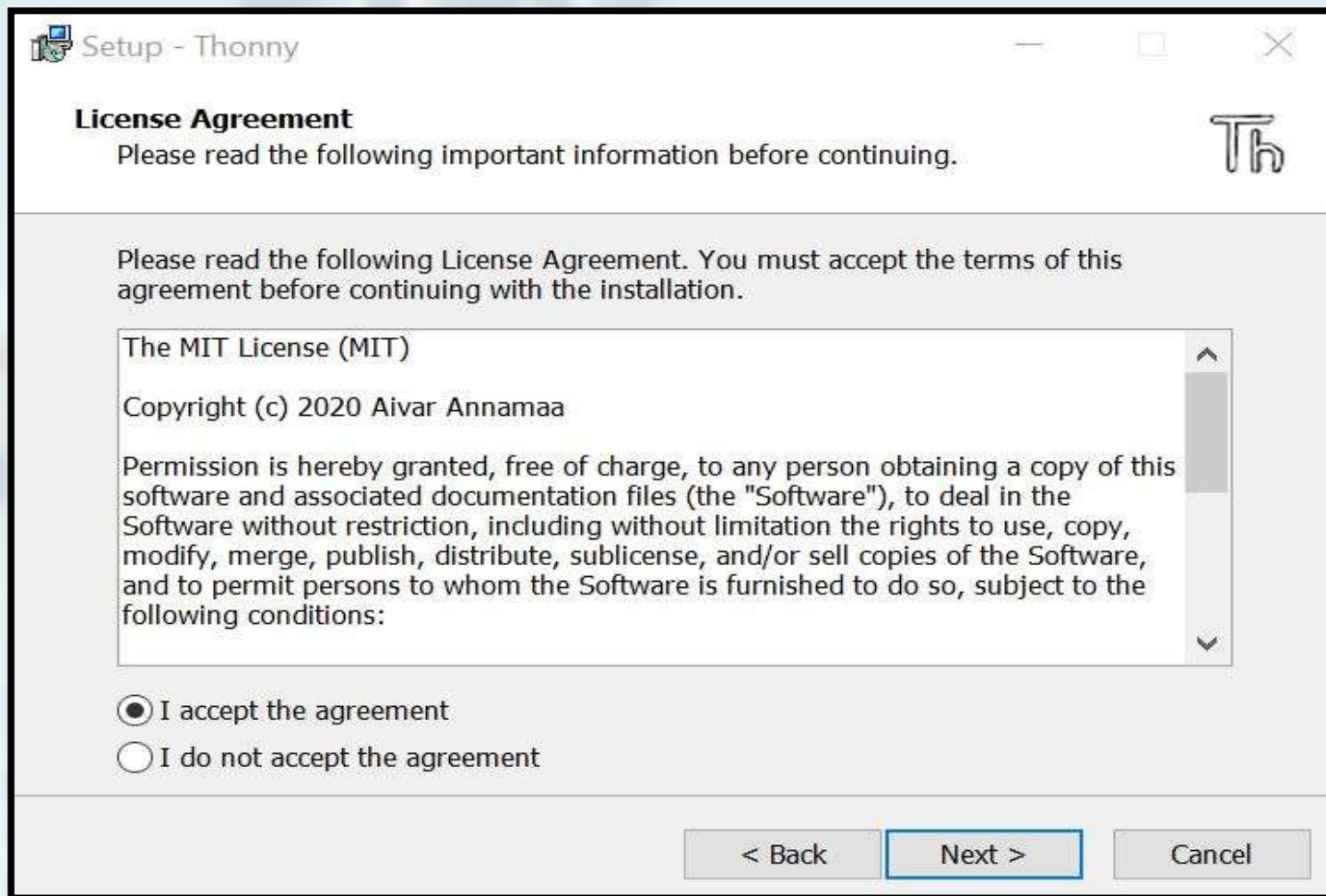
Firmware built with ESP-IDF v4.x, with support for BLE and PPP, but no LAN.

- GENERIC : [esp32-20210224-unstable-v1.14-82-g0daec0dca.bin](#)
- GENERIC : [esp32-20210223-unstable-v1.14-81-g53f5bb05a.bin](#)
- GENERIC : [esp32-20210222-unstable-v1.14-80-g75db0b907.bin](#)
- GENERIC : [esp32-20210221-unstable-v1.14-74-g1342debb9.bin](#)
- GENERIC : [esp32-idf4-20210206-unstable-v1.14-9-g9deddf122.bin](#)
- GENERIC : [esp32-idf4-20210205-unstable-v1.14-8-g1f800cac3.bin](#)
- GENERIC : [esp32-idf4-20210204-unstable-v1.14-3-g7c4435459.bin](#)
- GENERIC : [esp32-idf4-20210204-unstable-v1.14-1-g7f7b4f2bc.bin](#)
- GENERIC : [esp32-idf4-20210202-v1.14.bin](#) ←
- GENERIC : [esp32-idf4-20200902-v1.13.bin](#)
- GENERIC : [esp32-idf4-20191220-v1.12.bin](#)

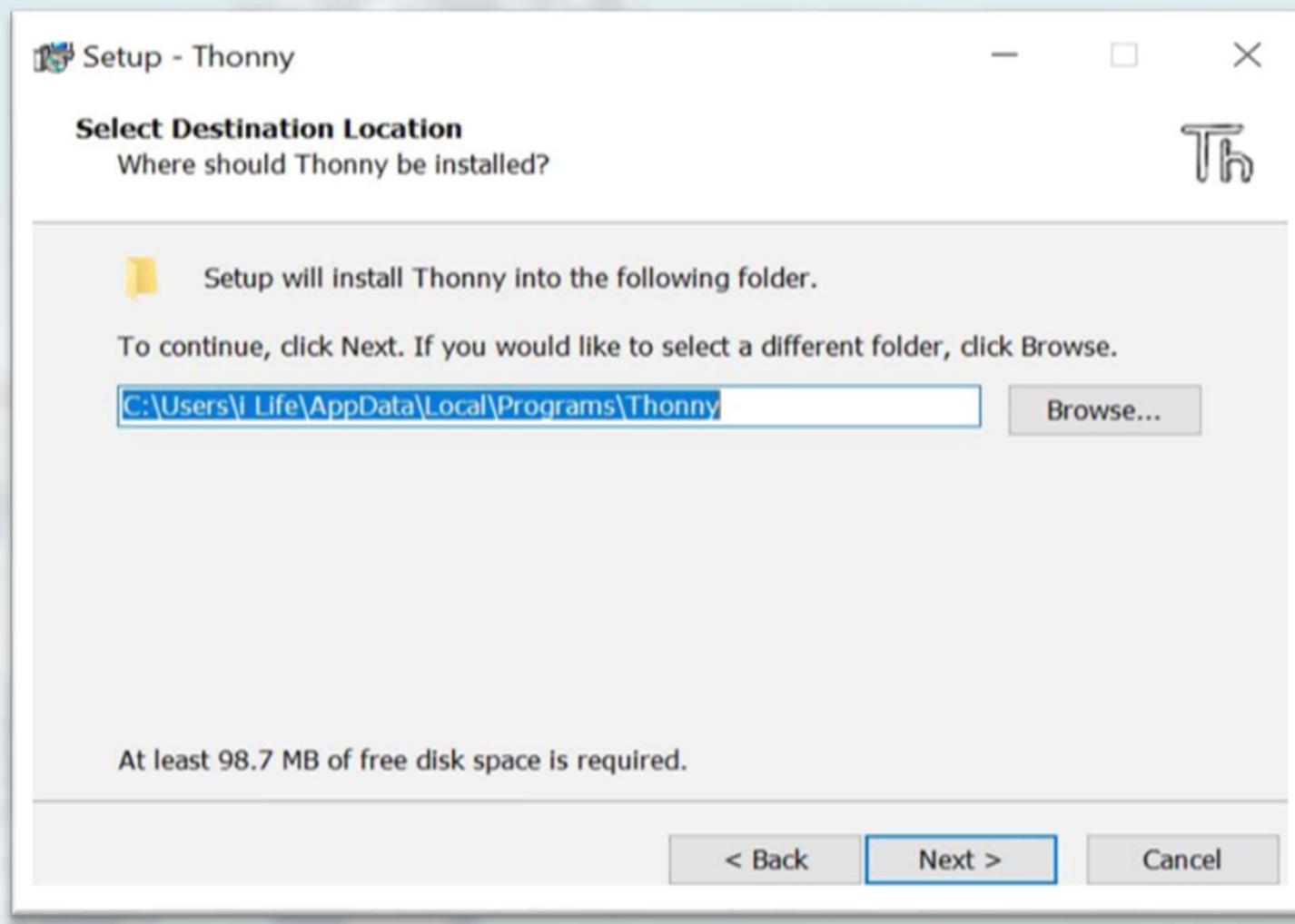
2. Click on the install icon of the software from the download folder (Thonny-3.3.5). Clicking will bring up a page named “Setup-Thonny” like the picture. Click Next.



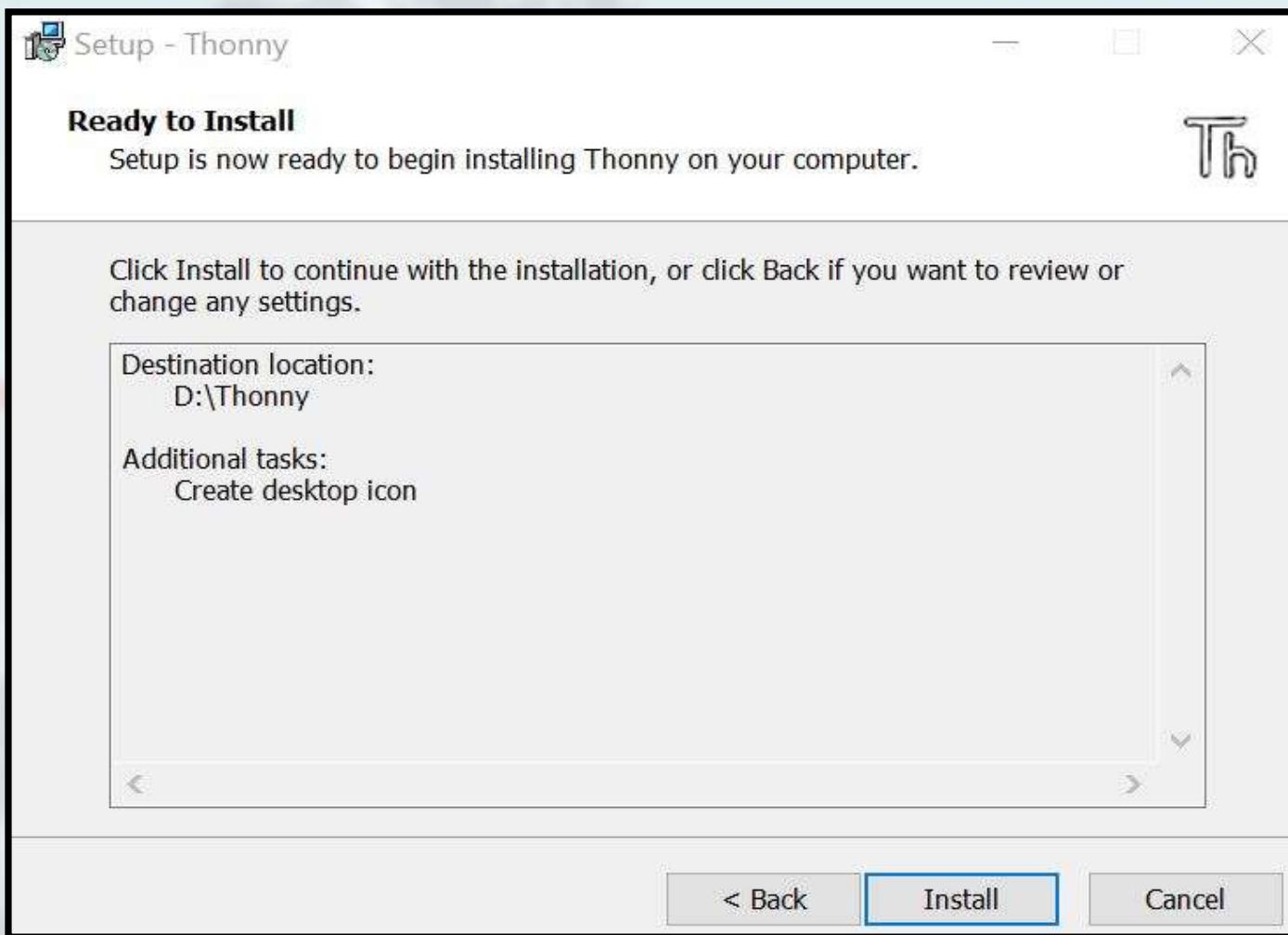
3.Another page called License Agreement will appear. From there I have to select the “I accept the Agreement “ option.



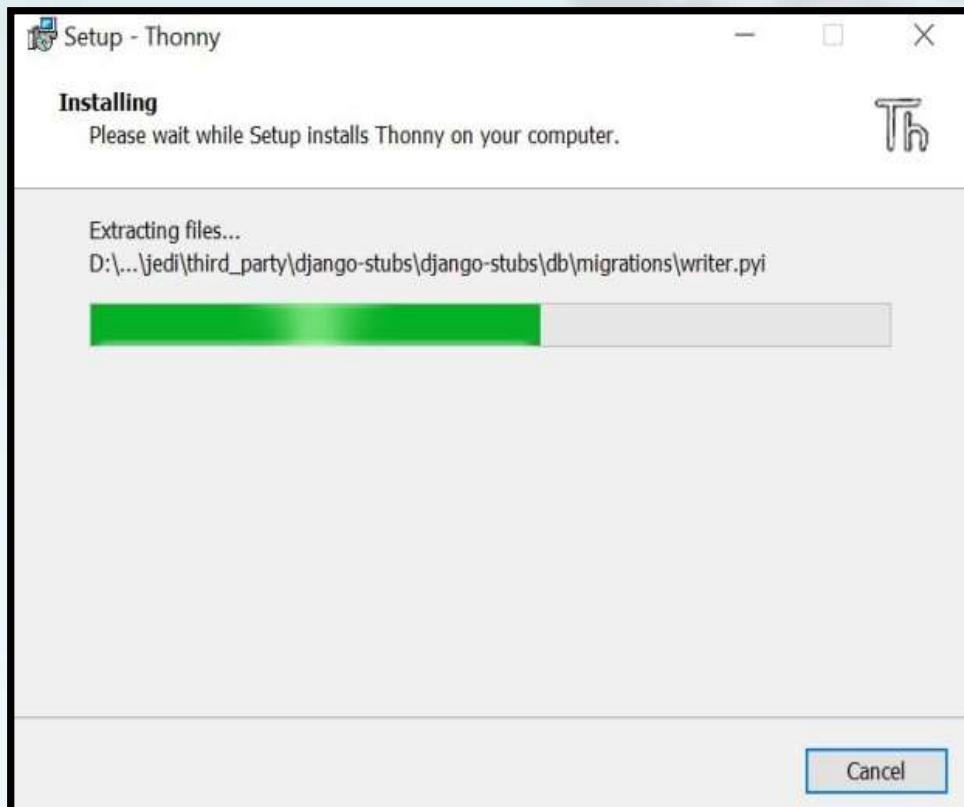
4. On the next page, from the Select Folder option, select the folder specified for the software..



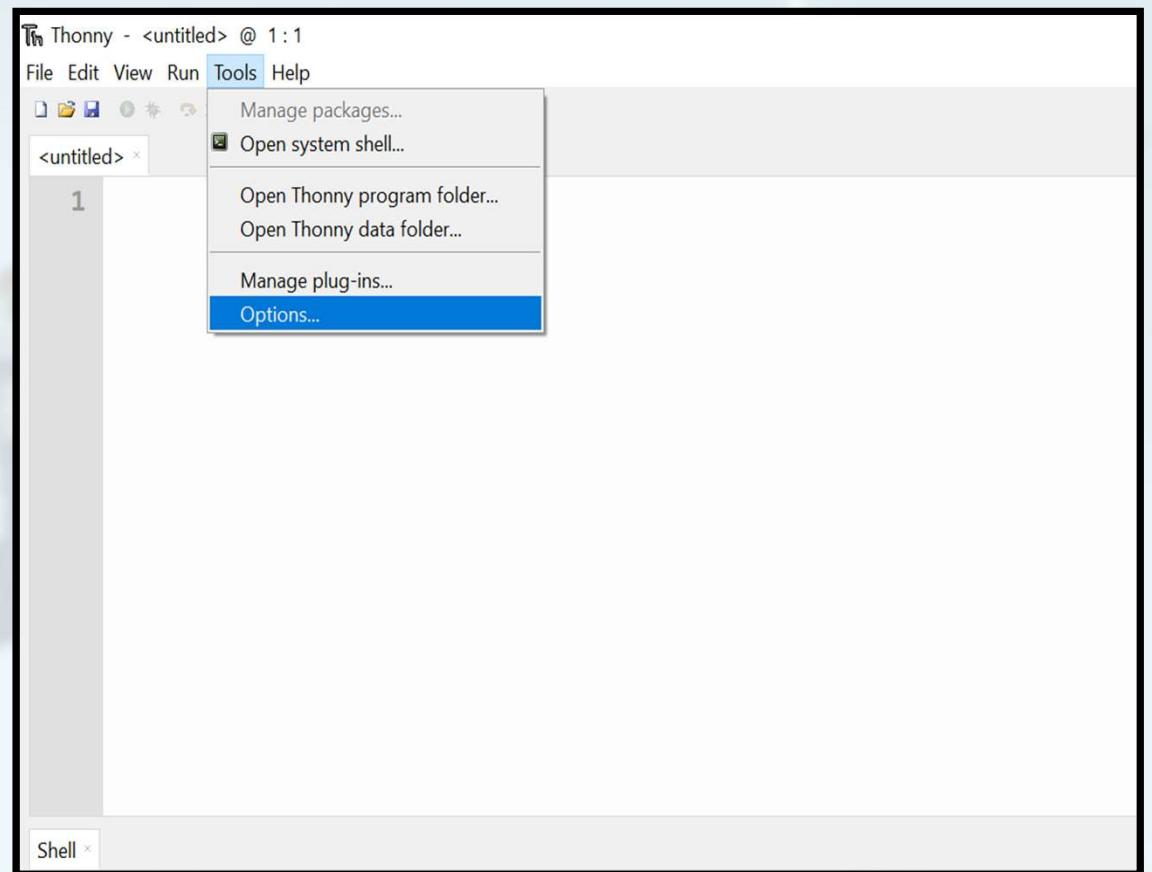
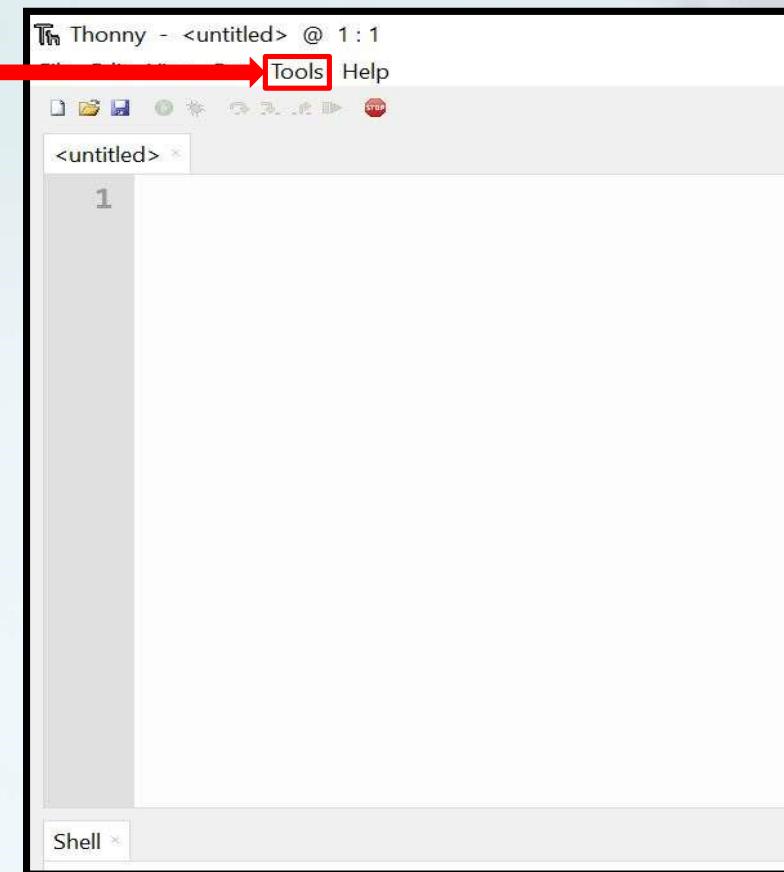
5. A page called Ready to Install will appear. From there you have to click on the Install button.



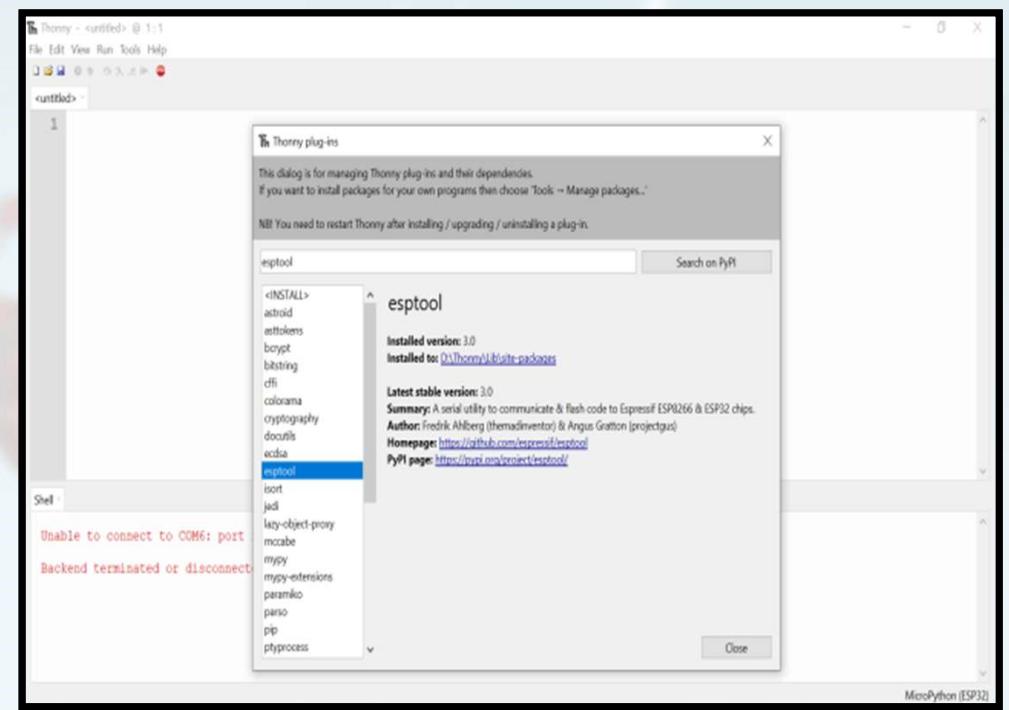
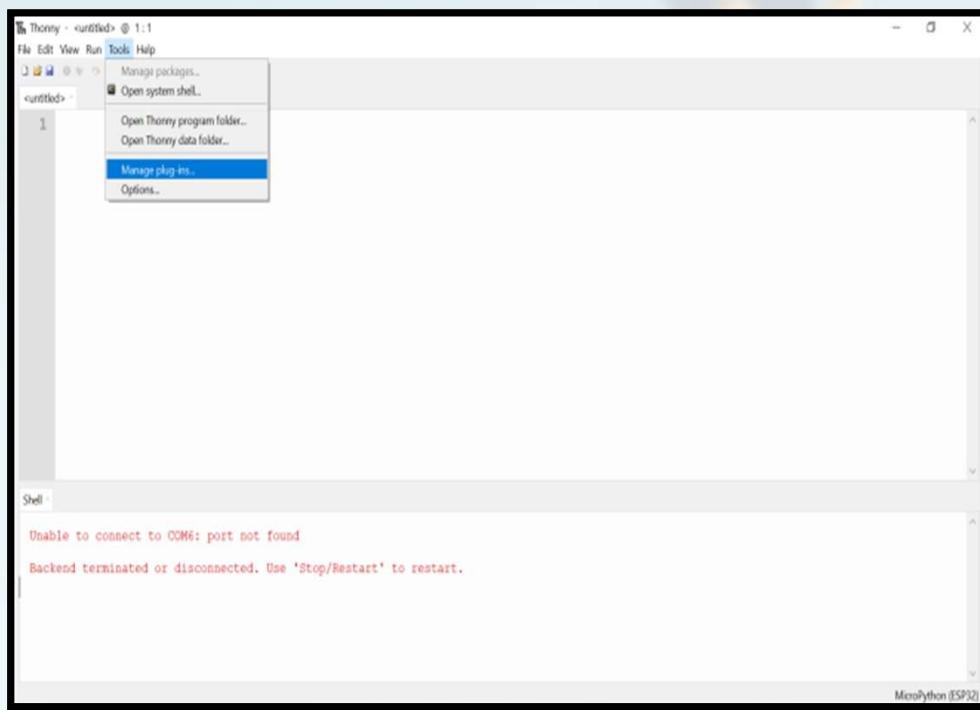
6. Once the installation is complete, the installation will be completed by clicking on the Finish Button.



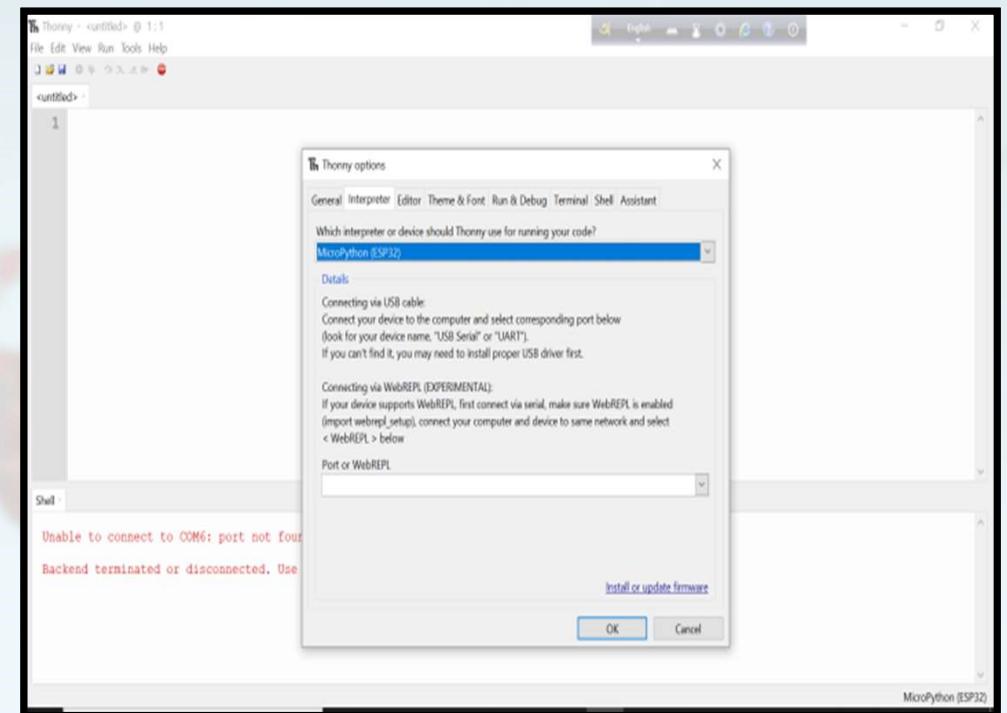
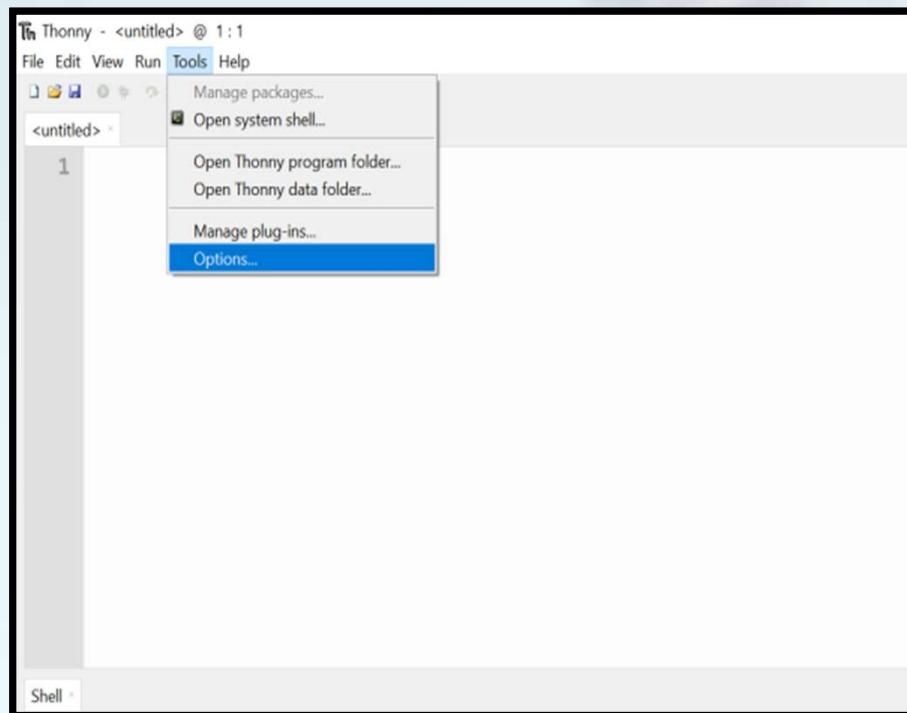
7. When you open the software, a page like the image will appear. There you have to click on the option named Tools.



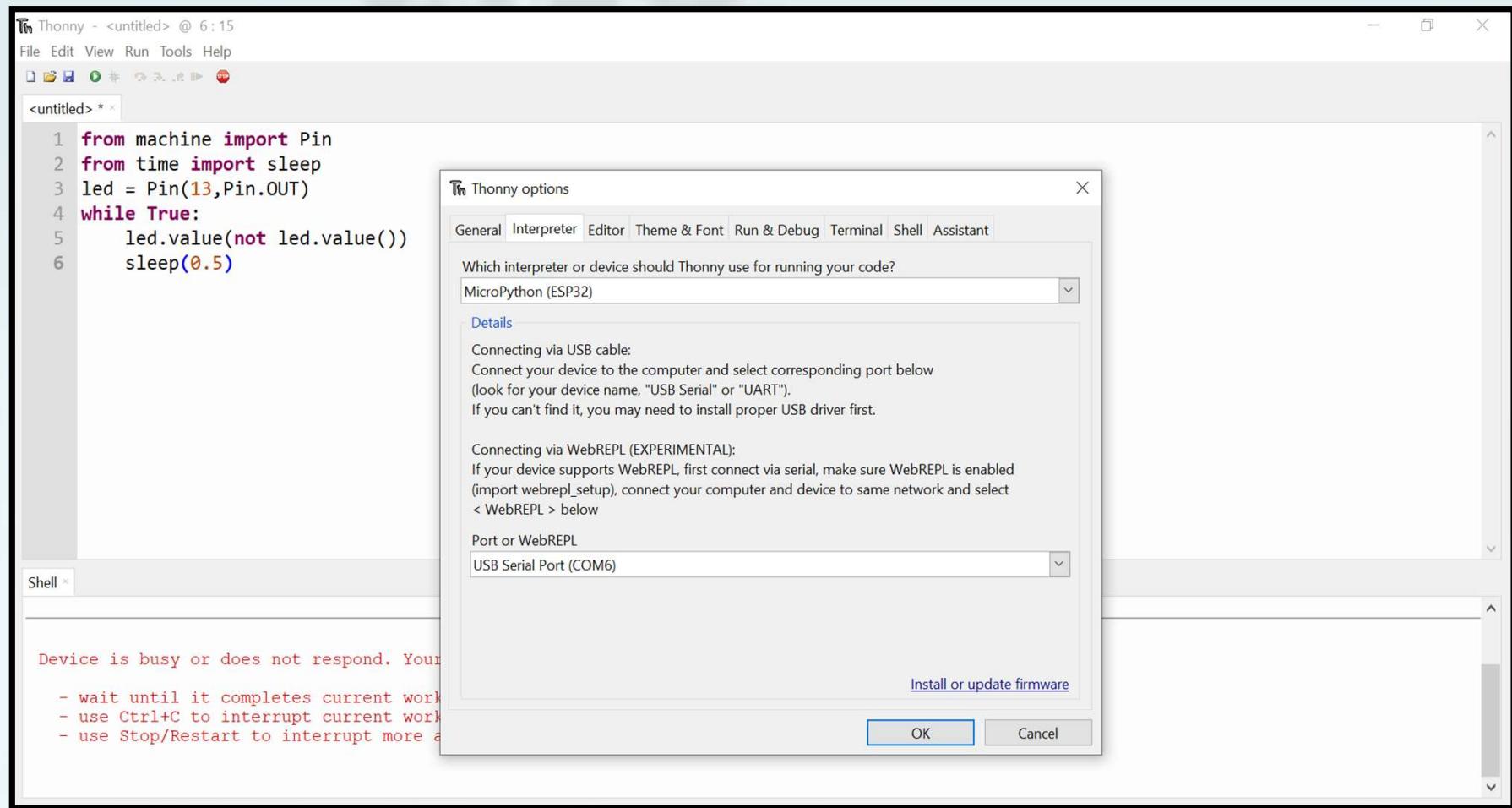
8. You have to select the “Manage plug-ins” option from Tools. Then a page like the picture will come. There you have to search by typing esptool in the search box. Next you need to install the “esptool” package.



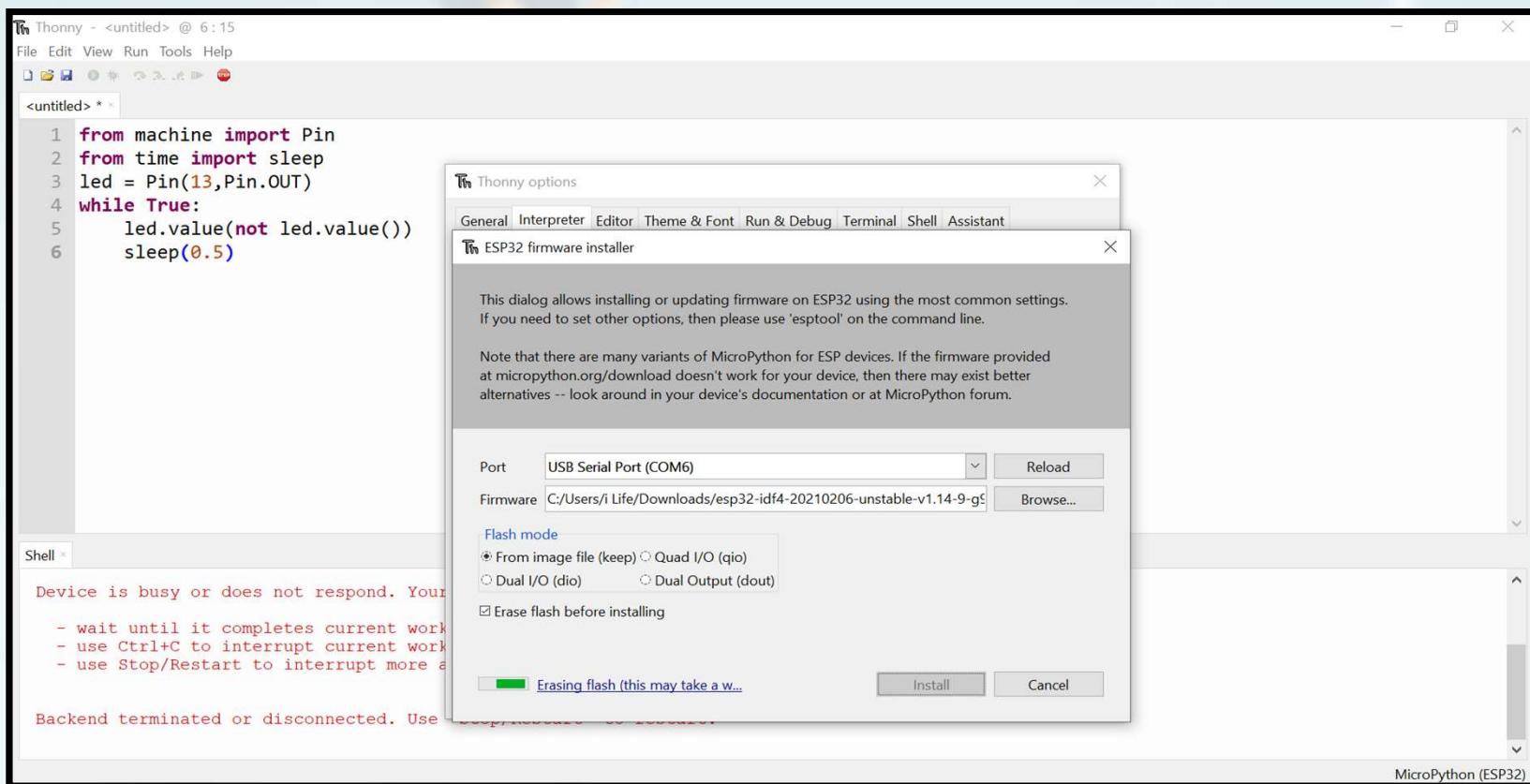
9. Next you have to click on Options. Then you have to click on the Interpreter option (tools -> options -> interpreter). Next you have to select “**MicroPython (ESP32)**” as the device like the picture.



10. Then you have to select the com port from the box named Port and upload it to the specified device (ESP32).



11. Next go to the Install or Update Firmware option. Then you have to make a port selection and add the previously downloaded Firmware with ESP-IDF file to the Firmware box in the Firmware box. After clicking the Install button, Flash will start taking and When 100% complete, you need to close the file by clicking the OK button.



12. A shell will appear as shown in the image. Where you can check if the software installation work is completed. If you type the command >>> print ("Hello World") it will print.



```
Shell x
entry 0x40080630
MicroPython v1.14-9-g9dedcf122 on 2021-02-06; ESP32 module with ESP32
Type "help()" for more information.
MicroPython v1.14-9-g9dedcf122 on 2021-02-06; ESP32 module with ESP32
Type "help()" for more information.
>>> print("Hello World ")
Hello World
>>>
```

13. Next we will do an LED Blinking with MicroPython via ESP32. For this you have to write the following code first. Code -

The screenshot shows the Thonny IDE interface. The top window is titled 'Thonny - <untitled> @ 6:15' and contains a code editor with the following script:

```
1 from machine import Pin
2 from time import sleep
3 led = Pin(13,Pin.OUT)
4 while True:
5     led.value(not led.value())
6     sleep(0.5)
```

Below the code editor is a 'Shell' window with the following output:

```
Shell
>>> help()
Welcome to MicroPython on the ESP32!
For generic online docs please visit http://docs.micropython.org/
For access to the hardware use the 'machine' module:
import machine
pin12 = machine.Pin(12, machine.Pin.OUT)
pin12.value(1)
```

```
from machine import Pin
from time import sleep
led = Pin(13,Pin.OUT)
while True:
    led.value(not led.value())
    sleep(0.5)
```

14. Clicking on the Run Button indicated by the indicator will run the code. And the LED on the ESP32 will blink.

