ESP32-C3 USER GUIDE

SUPPORT

If you require assistance, please feel free to contact us at **tourdeuscs@gmail.com.**

OVERVIEW

The ESP32 development board is designed based on the ESP32-C3 chip from Espressif Systems.

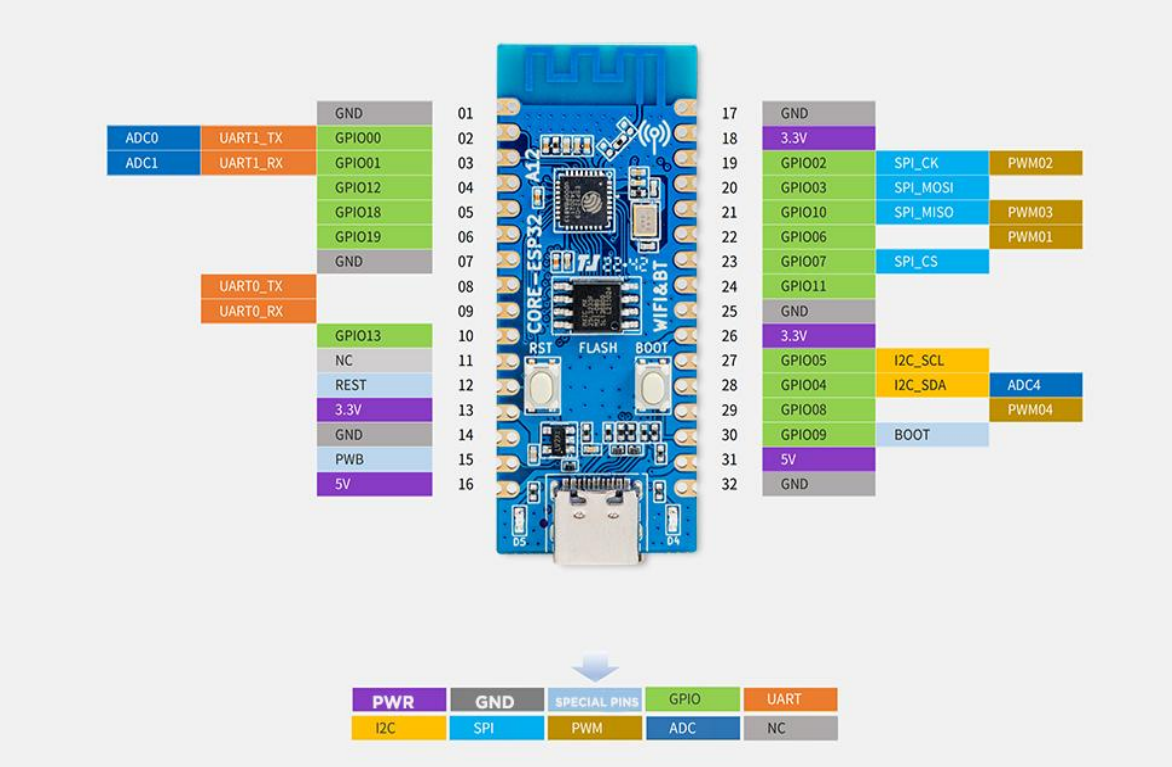
It has a small form factor and stamp hole design, making it convenient for developers to use.The board supports multiple interfaces, including UART, GPIO, SPI, I2C, ADC, and PWM, and is ideal for mobile devices, wearable electronics, and IoT applications with low-power performance.

It can function as a standalone system or peripheral device to the main MCU, providing Wi-Fi and Bluetooth functions through SPI/SDIO or I2C/UART interfaces.

ON BOARD RESOURCE

- This development board has one SPI flash with 4MB storage capacity, which can be expanded up to 16MB.
- It features 2 UART interfaces, UART0 and UART1, with UART0 serving as the download port.
- There is a 5-channel 12-bit ADC on this board, with a maximum sampling rate of 100KSPS.
- A low-speed SPI interface is also included in master mode.
- There is an IIC controller on this board.
- It has 4 PWM interfaces that can use any GPIO.
- There are 15 external GPIO pins that can be multiplexed.
- Additionally, it includes two SMD LED indicators, a reset button, a BOOT button, and a USB to TTL download debug port.

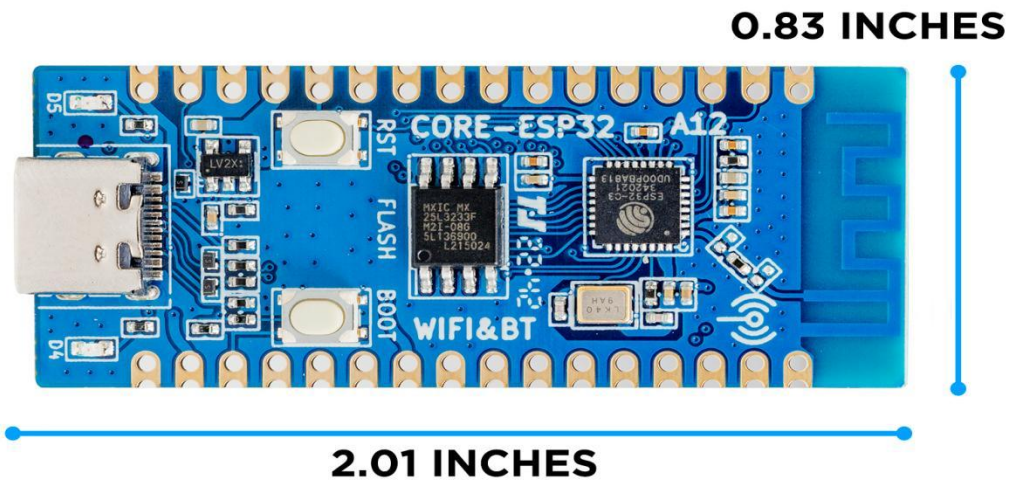
PINOUT DEFINITION



ESP32-C3 PCB

HTTPS://WIKI.LUATOS.COM/_STATIC/BOM/ESP32C3.HTML

DIMENSIONS (CLICK FOR DETAILS)



NOTES ON USE

- To avoid the ESP32 from entering download mode, the BOOT (IO09) pin should not be pulled down before powering up.
- It is not recommended to externally pull down the IO08 pin when designing, as this may prevent downloading via serial port when the pin is low during the download and burning process.
- In QIO mode, IO12 (GPIO12) and IO13 (GPIO13) are multiplexed for SPI signals SPIHD and SPIWP, but for increased GPIO availability, the development board uses 2-wire SPI in DIO mode, and as such, IO12 and IO13 are not connected to flash. When using self-compiled software, flash must be configured to DIO mode accordingly.
- Since the VDD of the external SPI flash is already connected to the 3.3V power supply system, there is no requirement for additional power supply configuration, and it can be accessed using the standard 2-wire SPI communication mode.
- By default, GPIO11 serves as the VDD pin of the SPI flash, and thus requires configuration before it can be utilized as a GPIO.

SCHEMATIC

Please click the following link for reference.
https://cdn.openluat-luatcommunity.openluat.com/attachment/20220609213416069_CORE-ESP32-A12.pdf

DEVELOPMENT ENVIRONMENT CONFIGURATION

Note: The following development system is Windows by default.

ARDUINO

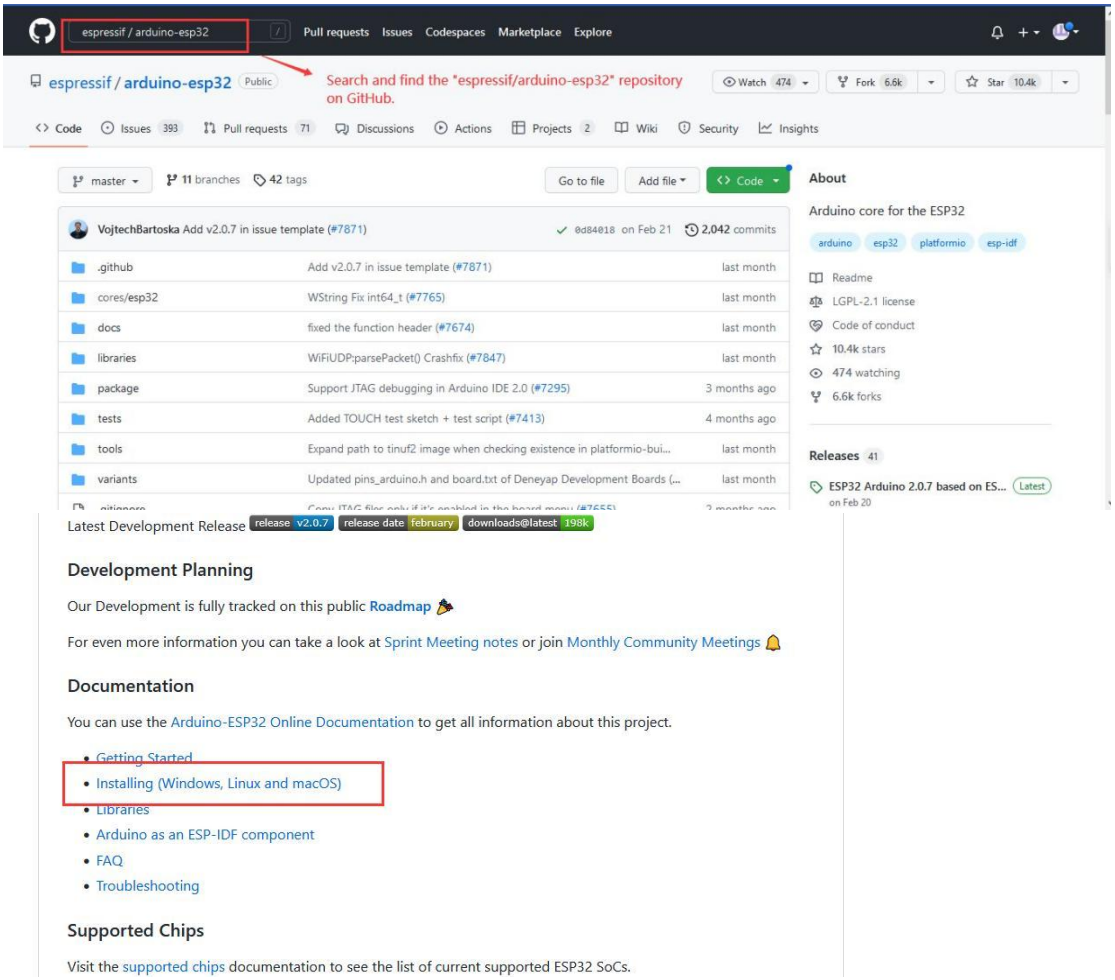
NOTE: Please ensure that you have uninstalled any previous versions of the ESP32 package before using this installation package.
You can do this by navigating to the folder "%LOCALAPPDATA%/Arduino15/packages" in the file manager, and deleting the folder named "esp32".

1. Open the official software download webpage, and choose the corresponding system and system bits to download.

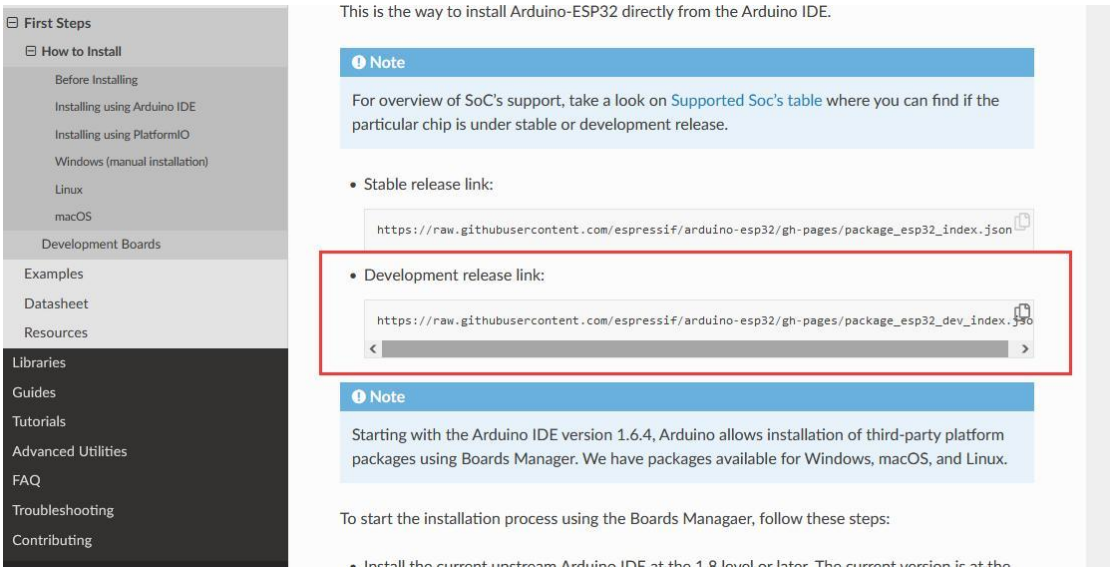
2. You can choose "Just Download", or "Contribute & Download".

3. Run to install the program and install it all by default.
4. Install arduino-esp32

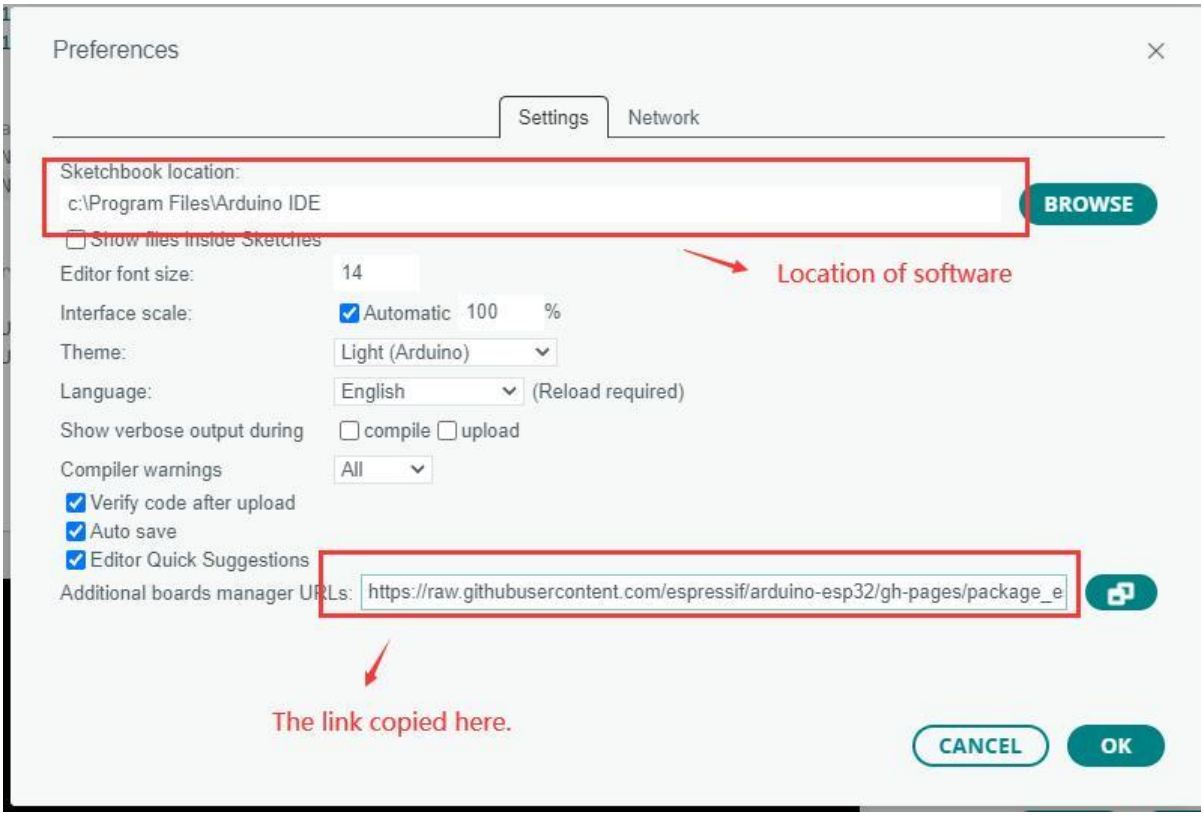
- Find the "espressif/arduino-esp32" repository on GitHub and click on the "Installing" link.



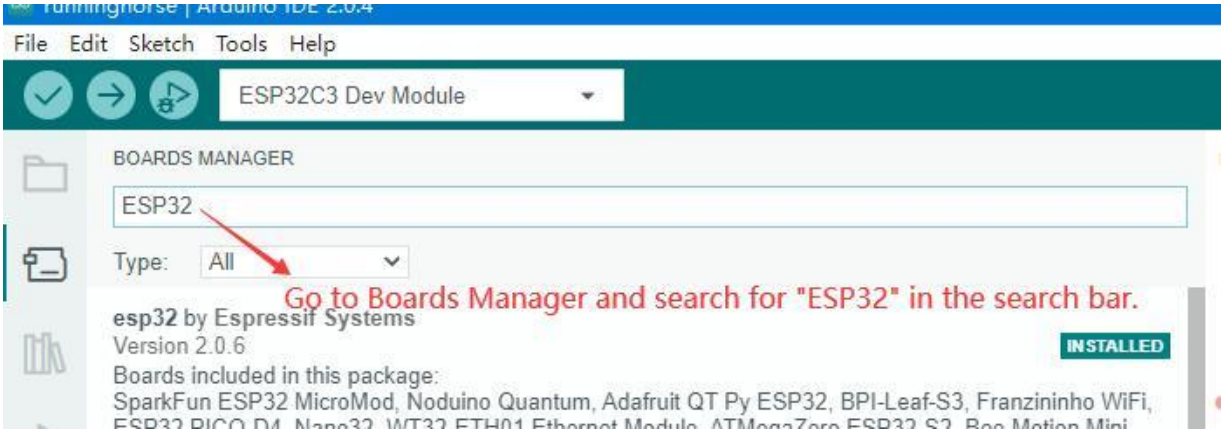
- Look for a URL named development release link and copied.



- In the Arduino IDE, click on File > Preferences > Additional boards manager URLs and add the URL that you found in step 2.



- Now, go back to Boards Manager and install the "ESP32" package.



- After installation, select Tools > Board and choose "ESP32C3 Dev Module" from the list.
- Finally, change the flash mode to DIO by going to Tools > Flash Mode, and change USB CDC on Boot to Enable.

Your ESP32 setup is now ready to go! To test it, you can run a demonstration program to ensure everything is working correctly.