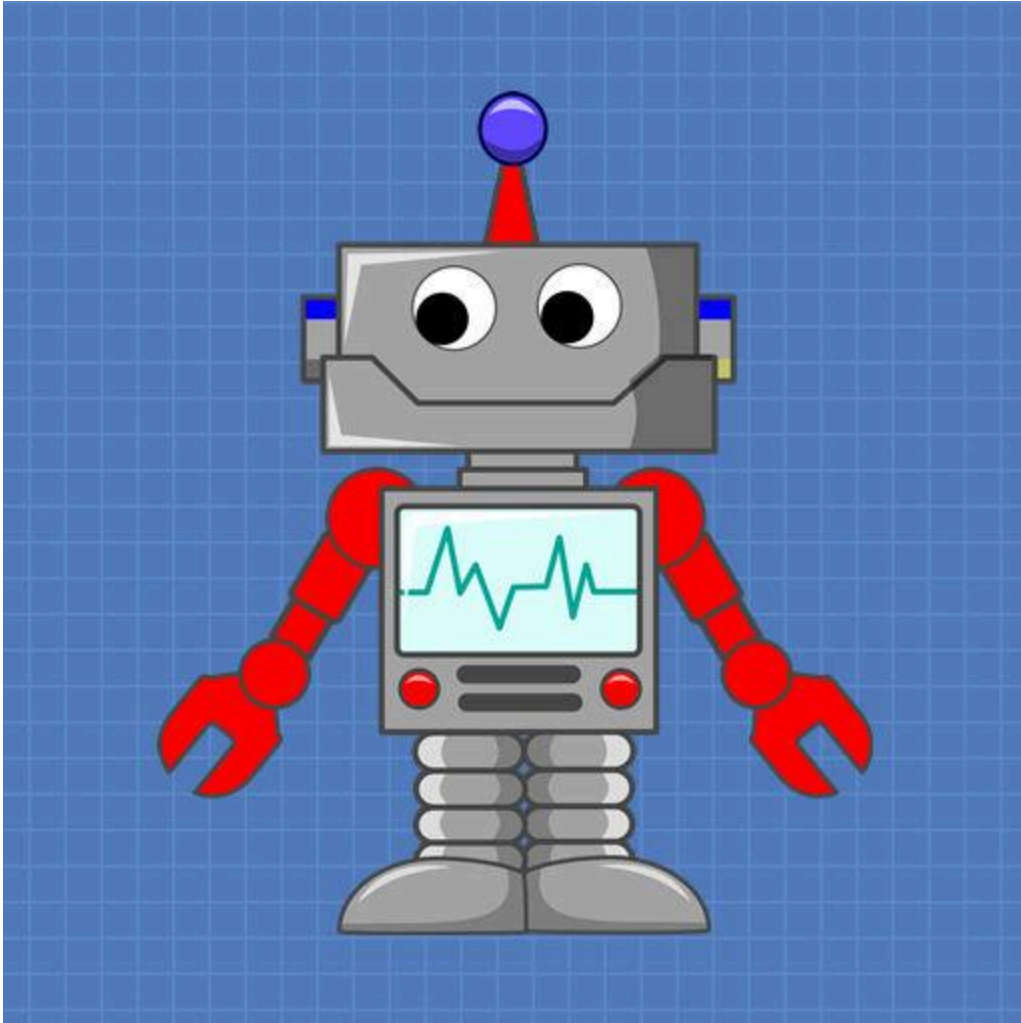


The ESP32 Guide 2024

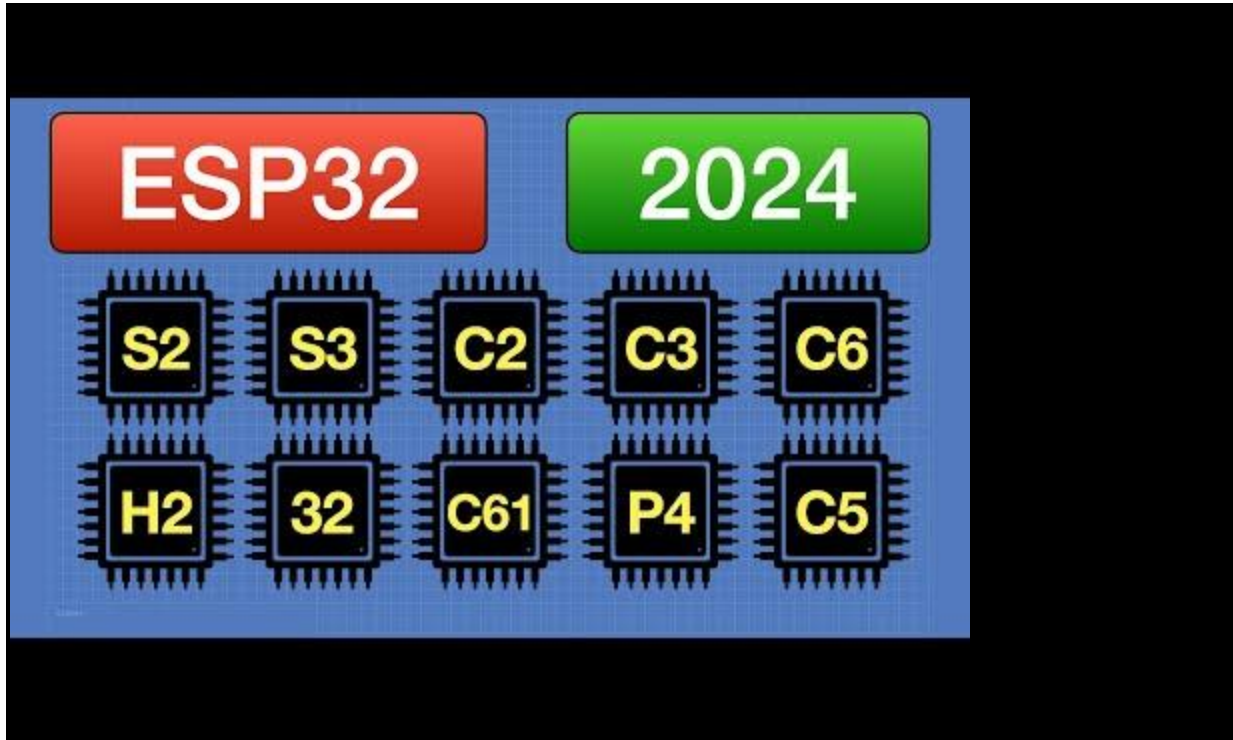


DroneBot Workshop Tutorial

<https://dronebotworkshop.com>

For more projects and tutorials visit the DroneBot Workshop - <https://dronebotworkshop.com>

The Espressif ESP32 has quickly become one of the most popular microcontrollers with hobbyists and developers worldwide. Its features include multiple digital and analog I/O ports, integrated touch switch inputs, support for IIS digital audio, and Wi-Fi and Bluetooth connectivity. And that's just the tip of the iceberg!



The ESP32 family has grown since its introduction and now consists of several SoC (System on a Chip) devices, modules, and development boards. Most of us use development boards, with hundreds available (Digikey lists almost 250 development boards on their website).

That's a lot of boards to choose from! This article will help you navigate this maze of specifications and features to select the best board for your project.

<https://dronebotworkshop.com>

Introduction – History of the ESP32

[Espressif Systems](#), a Shanghai-based company, was founded in 2008 by Teo Swee Ann, a Singaporean entrepreneur and engineer. The company initially focused on producing wireless communication technology solutions.



In its early years, Espressif primarily worked on providing Wi-Fi and Bluetooth chipsets for mainstream consumer electronics. Their first product was the ESP8089, a 2.4-GHz Wi-Fi system-on-chip (SoC) introduced in 2013.

Espressif chipsets were also used in Wi-Fi and Bluetooth adapters for Arduino boards. These adapters acted as a modem, providing a data connection between the Arduino and the network. They didn't have any intelligence of their own.

But that changed with the introduction of the ESP-01 module.

2014 – The ESP8266

The ESP-01 was a module based around an Espressif chipset named the ESP8266. As with the other Espressif Wi-Fi modules, it was used in adapters for microcontrollers like Arduino Uno's. It was introduced in 2014 and became popular due to its low cost.

But the ESP8266 also had a “hidden” feature. It wasn't really hidden at all; it was simply ignored due to the lack of English-language documentation.

The feature that made the ESP8266 so special was that it was not only a Wi-Fi and Bluetooth modem, but it was also a microcontroller. And a capable one, powered by a Tensilica Xtensa L106 32-bit microcontroller.

Once it was realized that the ESP8266 was a microcontroller in its own right, the maker community worked to provide development tools and documentation. Much of that early work was centered around a board made by a third-party manufacturer, AI-Thinker.

Espressif also worked to provide English (and other) language documentation and a development environment called the ESP-IDF (IoT Development Framework). The advent of open-source firmware like NodeMCU and Arduino compatibility made the ESP8266 a popular choice for makers and experimenters.

2016 – The ESP32

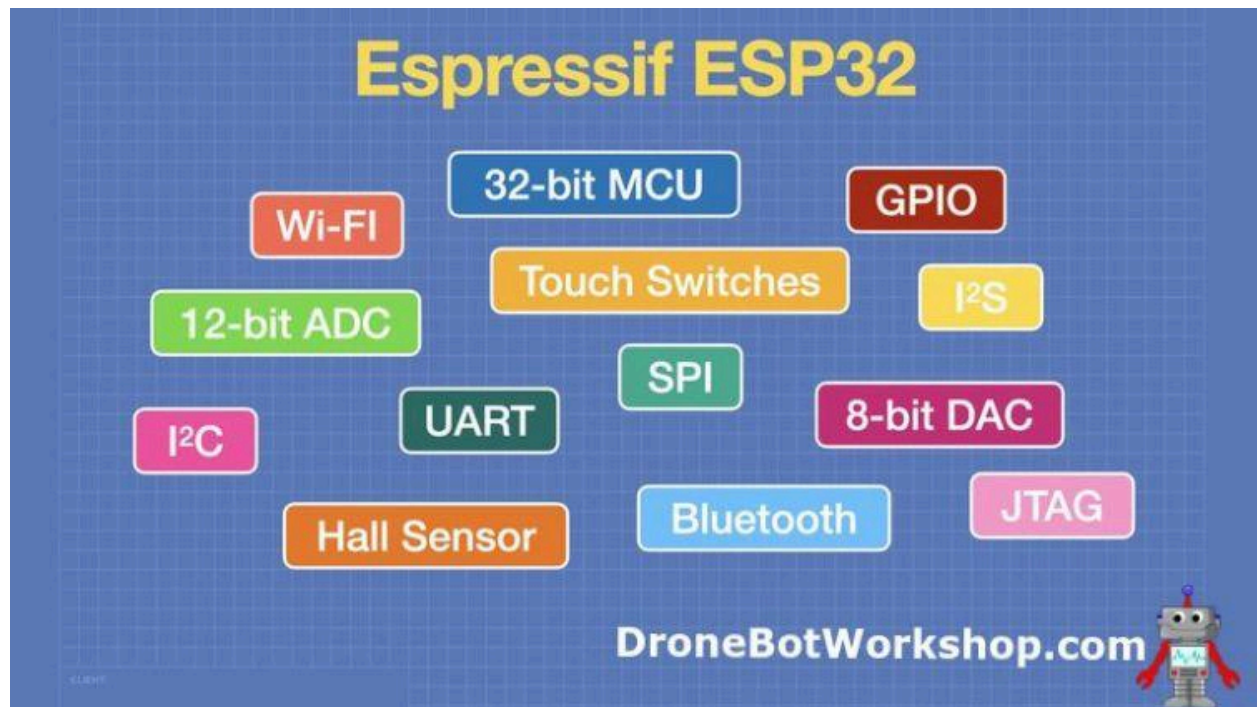
Espressif released the ESP32 in September 2016. This was an upgrade to the ESP8266, adding more I/O ports, memory, processing power, and features. Along with the ESP32 release, Espressif also provided extensive, multilingual documentation, updated development tools, and a board manager for the Arduino IDE with an extensive collection of sample sketches.

For more projects and tutorials visit the DroneBot Workshop - <https://dronebotworkshop.com>

This support and low-cost development boards from Espressif and other manufacturers quickly made the ESP32 a favorite among makers and experimenters.

ESP32 Features

Espressif Systems has made a number of updates to the ESP32 since its launch over seven years ago. The ESP32 is now four product lines, five if you include the still-available ESP8266. The result is dozens of ESP32 SoC devices and modules powering hundreds of development boards from Espressif and other manufacturers, most with slightly different features and specifications.

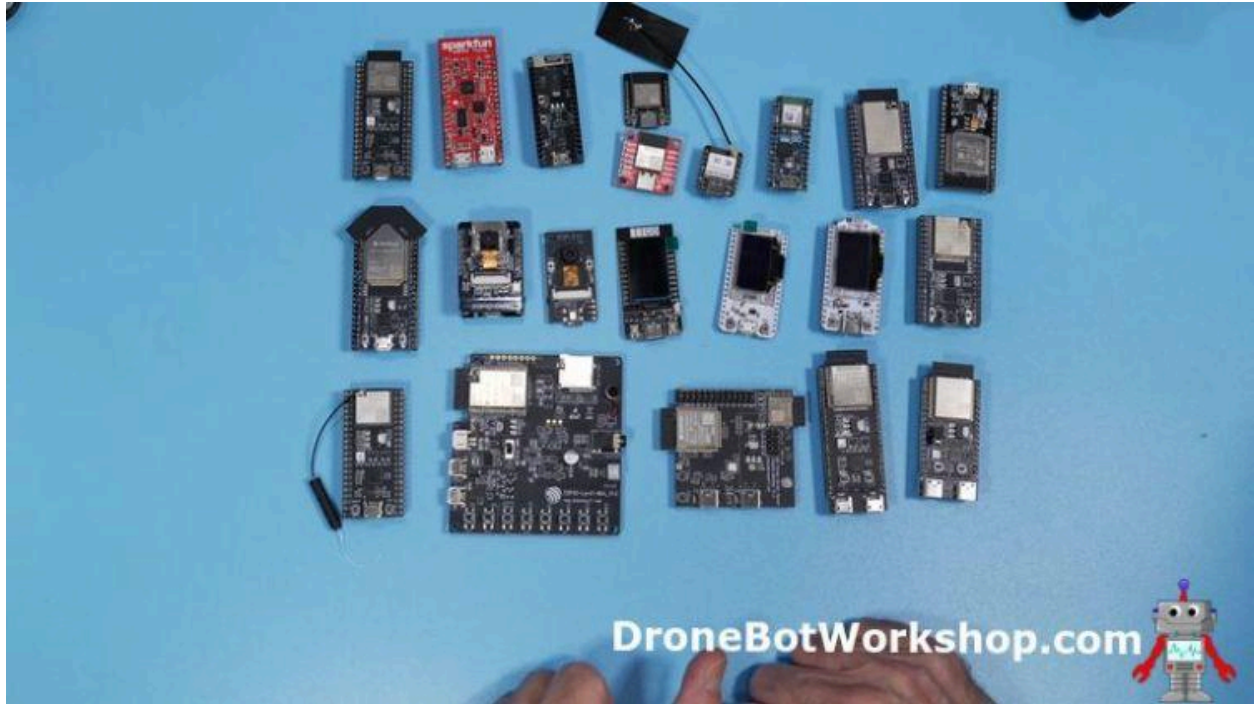


ESP32 SoCs, Modules and DevKits

Espressif Systems manufactures the ESP32 in three different formats:

- **SoCs** – The SoC (System-on-a-Chip) is the most basic format of ESP32, intended for manufacturers capable of designing their own circuitry and getting it FCC-certified if it uses any RF components. While most makers won't work directly with SoCs, knowing their specifications is important as they are at the heart of every design.
- **Modules** – Modules are those square “tin cans” you see on most ESP32 boards, with the ESP32 part number inscribed. Internally, they consist of SoCs prewired to external connections. This is easier to design with and avoids the need for FCC-Certification, as Espressif has already done that. Certification is a long and expensive process that only large manufacturers will willingly absorb, so most third-party manufacturers of ESP32 boards use modules. Some advanced experimenters, who design their own circuit boards, use modules.
- **DevKits** – DevKits are development boards manufactured by Espressif. They can use ESP32 SoCs or modules, often with additional circuitry such as OLED displays, pushbuttons, and LEDs. These are the most common ESP32s used by experimenters and makers.

Of course, you can also buy boards from other manufacturers like Adafruit, Sparkfun, LilyGo, Seeed Studio, Arduino, and others. These boards usually use modules and also include extra features as well. We'll discuss a few of these in detail later, but first, we need to meet the family!



The ESP32 Family

As of this writing, the ESP32 product line has been divided into five tracks (and an upcoming sixth one), which I will refer to as “families,” to give everything a more homey feeling.



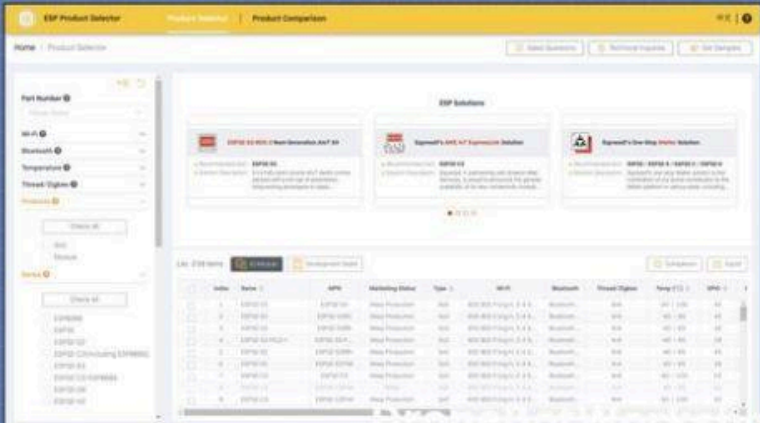
These “families” include the semiretired grandfather, the decade-old ESP8266, which is not recommended for new designs but is still available for legacy circuits.

We’ll let Grandpa have a well-deserved rest while we introduce the other family members and their children.

For more projects and tutorials visit the DroneBot Workshop - <https://dronebotworkshop.com>

Espressif ESP32

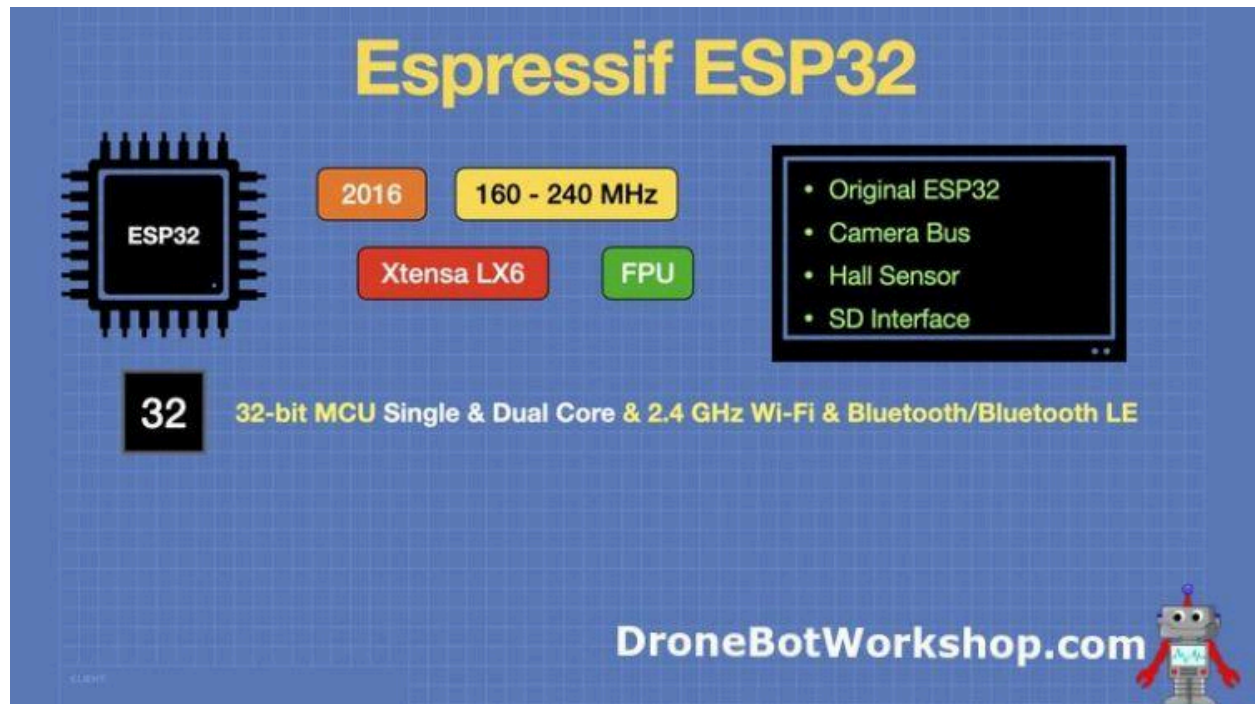
<https://products.espressif.com/#/product-selector>



Index	Part Number	Name	Marketing Status	Type	SKU	Bluetooth	Thread/Output	Price (USD)
1	ESP32-WROOM	ESP32-WROOM	Active Production	Kit	ESP32-WROOM-01	Bluetooth	None	4.00
2	ESP32-WROOM-02	ESP32-WROOM-02	Active Production	Kit	ESP32-WROOM-02	Bluetooth	None	3.50
3	ESP32-WROOM-02L	ESP32-WROOM-02L	Active Production	Kit	ESP32-WROOM-02L	Bluetooth	None	3.50
4	ESP32-WROOM-02L-1	ESP32-WROOM-02L-1	Active Production	Kit	ESP32-WROOM-02L-1	Bluetooth	None	3.50
5	ESP32-WROOM-02L-2	ESP32-WROOM-02L-2	Active Production	Kit	ESP32-WROOM-02L-2	Bluetooth	None	3.50
6	ESP32-WROOM-02L-3	ESP32-WROOM-02L-3	Active Production	Kit	ESP32-WROOM-02L-3	Bluetooth	None	3.50
7	ESP32-WROOM-02L-4	ESP32-WROOM-02L-4	Active Production	Kit	ESP32-WROOM-02L-4	Bluetooth	None	3.50
8	ESP32-WROOM-02L-5	ESP32-WROOM-02L-5	Active Production	Kit	ESP32-WROOM-02L-5	Bluetooth	None	3.50
9	ESP32-WROOM-02L-6	ESP32-WROOM-02L-6	Active Production	Kit	ESP32-WROOM-02L-6	Bluetooth	None	3.50
10	ESP32-WROOM-02L-7	ESP32-WROOM-02L-7	Active Production	Kit	ESP32-WROOM-02L-7	Bluetooth	None	3.50

This list of ESP32 products is current as of January 2024. Check out the [Espressif ESP32 Product Selector](#) for a complete, searchable list.

ESP32 Series



The [original ESP32 series](#) consists of:

- SoCs – 10 (4 are no longer recommended for new designs).
- Modules – 20 (8 are no longer recommended for new designs).
- DevKits – 15.

The following is Espressifs' list of ESP32-32 Series features:

- *ESP32 embedded, two or one Xtensa® 32-bit LX6 microprocessor(s) with adjustable clock frequency, ranging from 80 MHz to 240 MHz*
- *+19.5 dBm output power ensures a good physical range*
- *Classic Bluetooth for legacy connections, also supporting L2CAP, SDP, GAP, SMP, AVDTP, AVCTP, A2DP (SNK) and AVRCP (CT)*
- *Support for Bluetooth Low Energy (Bluetooth LE) profiles including L2CAP, GAP, GATT, SMP, and GATT-based profiles like BluFi, SPP-like, etc*
- *Bluetooth Low Energy (Bluetooth LE) connects to smart phones, broadcasting low-energy beacons for easy detection*

For more projects and tutorials visit the DroneBot Workshop - <https://dronebotworkshop.com>

- *Sleep current is less than 5 μ A, making it suitable for battery-powered and wearable-electronics applications*
- *Peripherals include capacitive touch sensors, Hall sensor, SD card interface, Ethernet, high-speed SPI, UART, I2S and I2C*
- *Fully certified with integrated antenna and software stacks*

ESP32-H Series



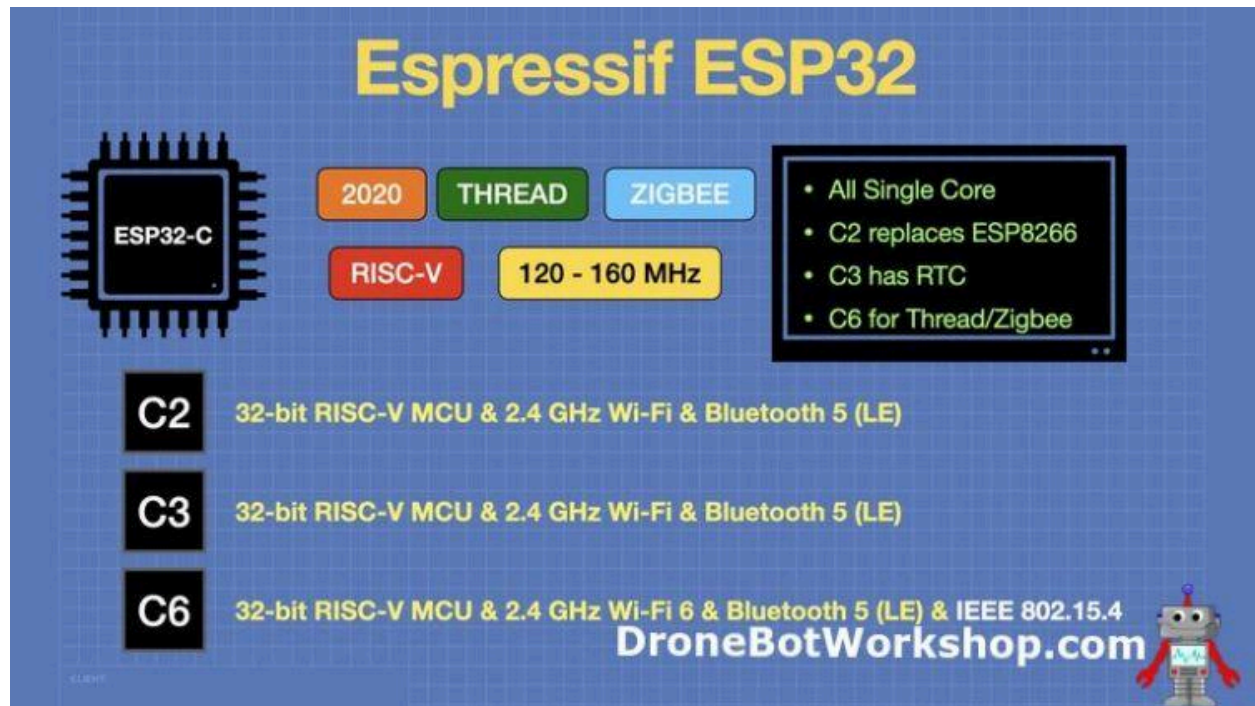
The [ESP32-H series](#) consists of:

- SoCs – 2.
- Modules – 12 (4 are no longer recommended for new designs).
- DevKits – 2.

The following is Espressifs' list of ESP32-H Series features:

- *32-bit RISC-V single-core processor that operates at up to 96 MHz*
- *320 KB SRAM, 128 KB ROM, 4 KB LP Memory, and works with external flash*
- *19 programmable GPIOs, with support for UART, SPI, I2C, I2S, Remote Control Peripheral, LED PWM, Full-speed USB Serial/JTAG Controller, GDMA, MCPWM*
- *Can be used for building Thread end devices, as well as Thread border router and Matter bridge by combining it and ESP Wi-Fi SoC*

ESP32-C Series



The ESP32-C series consists of:

- SoCs – 4.
- Modules – 10.
- DevKits – 2.

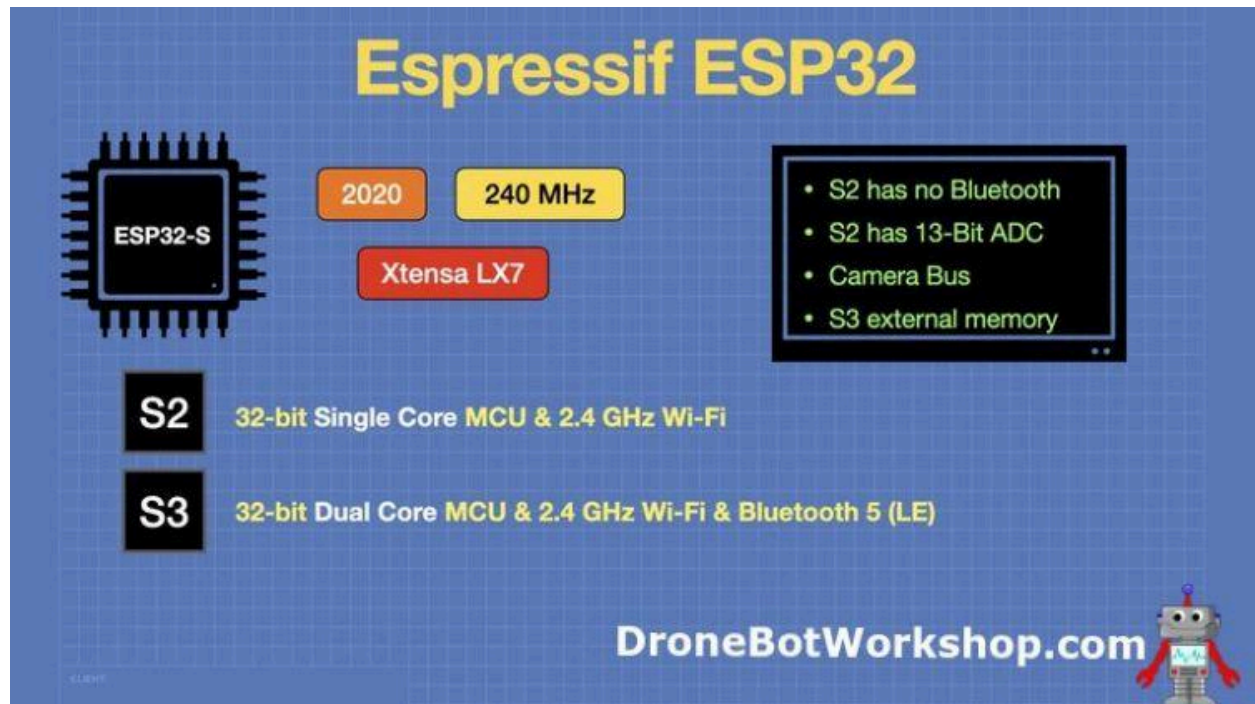
The following is Espressif's list of ESP32-C2 Series features:

- *High-performance 240 MHz single-core CPU*
- *Ultra-low-power performance: fine-grained clock gating, dynamic voltage and frequency scaling*
- *Security features: eFuse, flash encryption, secure boot, signature verification, integrated AES, SHA and RSA algorithms*
- *Peripherals include 43 GPIOs, 1 full-speed USB OTG interface, SPI, I2S, UART, I2C, LED PWM, LCD interface, camera interface, ADC, DAC, touch sensor, temperature sensor*

For more projects and tutorials visit the DroneBot Workshop - <https://dronebotworkshop.com>

- *Availability of common cloud connectivity agents and common product features shortens the time to market*

ESP32-S Series



The ESP32-S series consists of:

- SoCs – 4.
- Modules – 12 (4 are no longer recommended for new designs).
- DevKits – 2.

Here is Espressif's list of features for the ESP32-S3 Series:

- *Xtensa® 32-bit LX7 dual-core processor that operates at up to 240 MHz*
- *512 KB of SRAM and 384 KB of ROM on the chip, and SPI, Dual SPI, Quad SPI, Octal SPI, QPI, and OPI interfaces that allow connection to flash and external RAM*
- *Additional support for vector instructions in the MCU, which provides acceleration for neural network computing and signal processing workloads*
- *Peripherals include 45 programmable GPIOs, SPI, I2S, I2C, PWM, RMT, ADC and UART, SD/MMC host and TWAITM*

- *Reliable security features are ensured by RSA-based secure boot, AES-XTS-based flash encryption, the innovative digital signature, and the HMAC peripheral, “World Controller.”*

Working with the ESP32

If you purchase an Espressif ESP32 DevKit, or a development board from another manufacturer, you'll need some software to program it. What software you use depends upon the programming language you intend to use and, to some degree, your level of expertise.

Powering the ESP32

The most common method of powering an ESP32 development board is to use the USB connector. Some ESP32 DevKits have two USB ports, and either one can power the board.

The ESP32 is a 3.3-volt microcontroller, but most boards also have a 5-volt power supply input, along with an internal buck converter or charge pump. It's preferable to use the 5-volt power input instead of the 3.3-volt one on most boards.

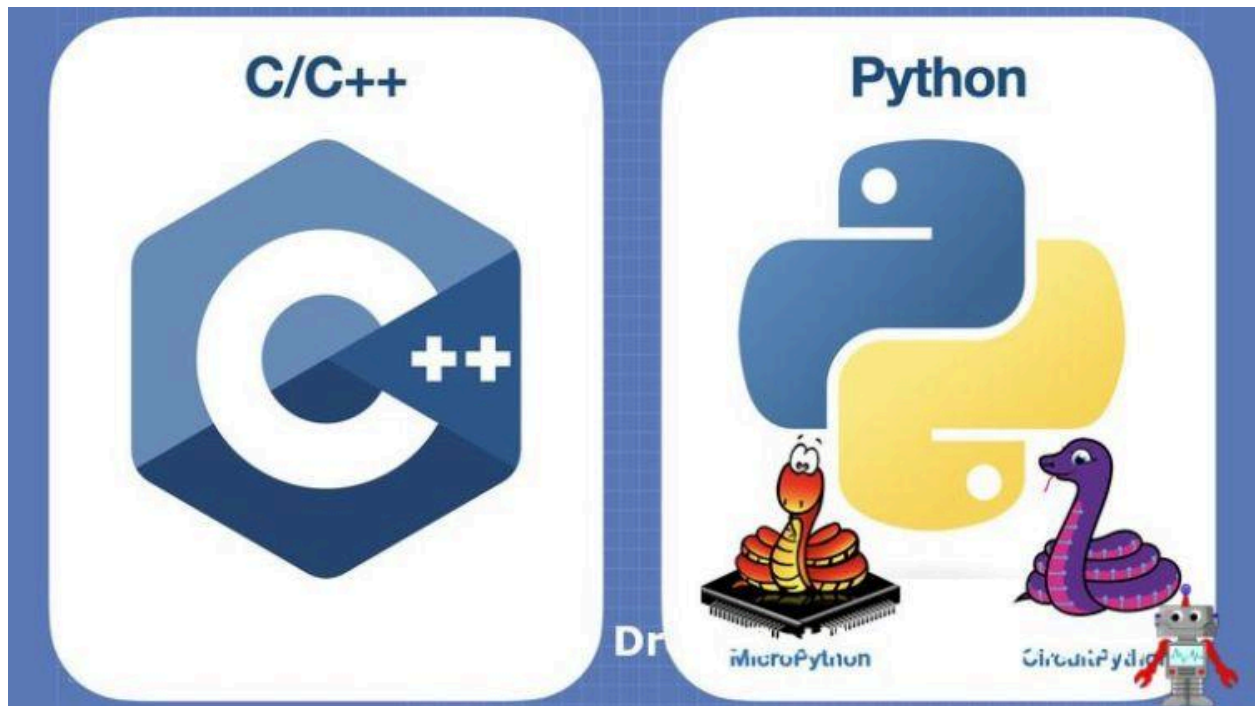
Although the ESP32 is a low-powered device, you will need a bit of current capability on your power supply as the radio (Wi-Fi, Bluetooth etc.) can consume a bit when initializing. How much current capability depends upon several factors, such as the number of I/O ports you use as outputs.

Some ESP32 boards also have connectors for rechargeable batteries; many will charge the battery when used with USB. If you use one of these boards, be sure to find out the

For more projects and tutorials visit the DroneBot Workshop - <https://dronebotworkshop.com>

specifications for the battery, as many of the charging systems are set for a particular type of battery.

ESP32 Programming Languages



When it comes to programming an ESP32, you have a few choices:

- **C/C++** – Probably the most popular language used for ESP32 programming.
- **Python** – Both MicroPython and CircuitPython interpreters can run on the ESP32.
- **JavaScript** – [Espruino](#) is a JavaScript interpreter for the ESP32.
- **LUA** – [LUA for NodeMCU](#) can also run on some ESP32 boards.
- **Rust** – Not officially supported, but [some folks have got it to run](#).

For more projects and tutorials visit the DroneBot Workshop - <https://dronebotworkshop.com>

For most of us, the choices will be between C++ and MicroPython or CircuitPython. Performance will be better with C++, as it's a compiled language instead of an interpreted one like the two Python choices.

ESP32 Development Environments

C/C++

There are three C/C++ development environments commonly used for working with the ESP32:

- **Arduino IDE** – Most ESP32 boards have corresponding entries in the *ESP32 Boards Manager for Arduino*. This is the most common environment for hobbyists.
- **PlatformIO** – Offers an improved interface and better library control but is not as beginner-friendly as the Arduino IDE. Most developers run this under the [Microsoft VS Code IDE](#).
- **Espressif ESP IDF** – The *Espressif IoT Development Framework*. This is the official and probably the best way to program an ESP32. It requires more programming expertise, but every ESP device is supported. It can also be run in the VS Code IDE.

MicroPython/CircuitPython

There are several environments for developing in Python that can be used with either MicroPython or CircuitPython. Here are a few:

- **Mu Editor** – A simple Python editor that is great for beginners.
- **Thonny IDE** – Another simple editor for beginners.
- **Pymakr** – A more advanced Python IDE. Runs under the VS Code IDE

For more projects and tutorials visit the DroneBot Workshop - <https://dronebotworkshop.com>

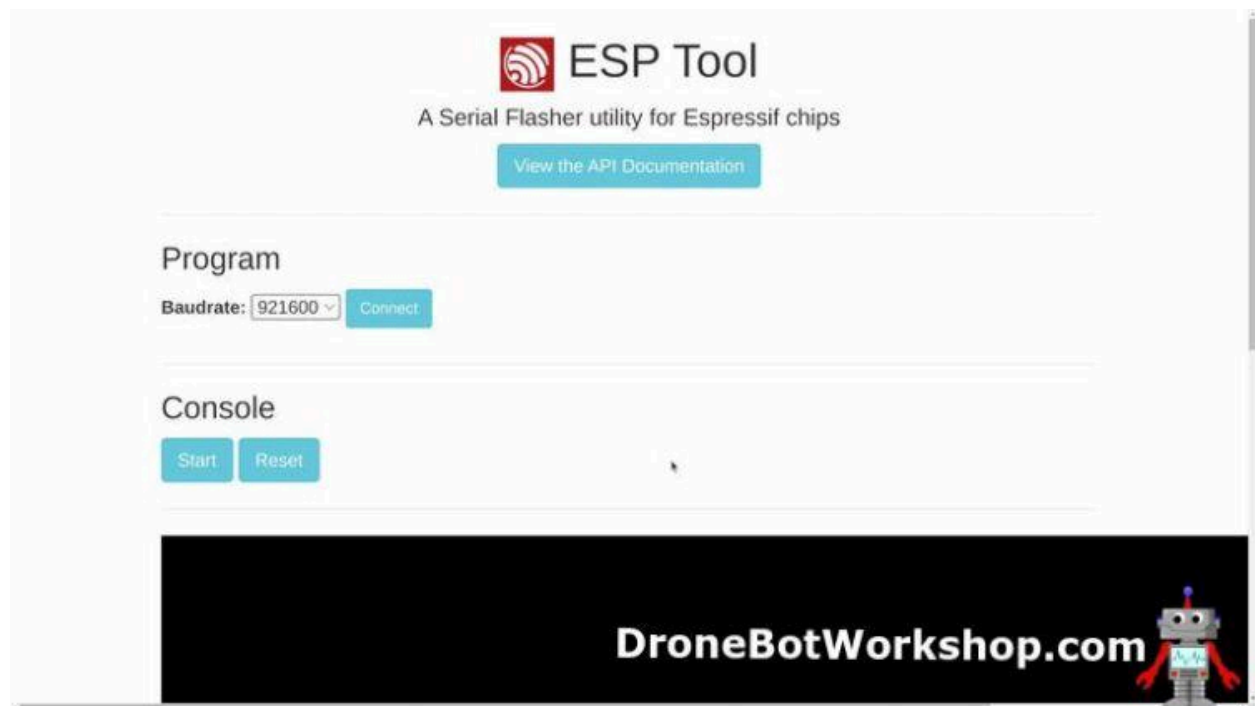
ESP Tool (Online Version)

The online version of the popular [ESP Tool Python script](#) can be used to identify “mystery” ESP32 and ESP8266 boards. It also has many other uses, such as reloading the bootloader.

This version of ESP Tool is written in JavaScript and can be downloaded from GitHub if you wish to run it on your own server. For most of us, though, we can run it online using this link – [ESP Tools Online version](#).

You must use Google Chrome or another Chrome-based browser such as Opera or Microsoft Edge. Browsers like Firefox and Safari won’t work with this utility, as they don’t provide access to your USB port like Chrome-based browsers do.

On the ESP Tool page, you’ll see a dropdown for selecting the Baud rate. You should set this to “115200” for most ESP32 boards.

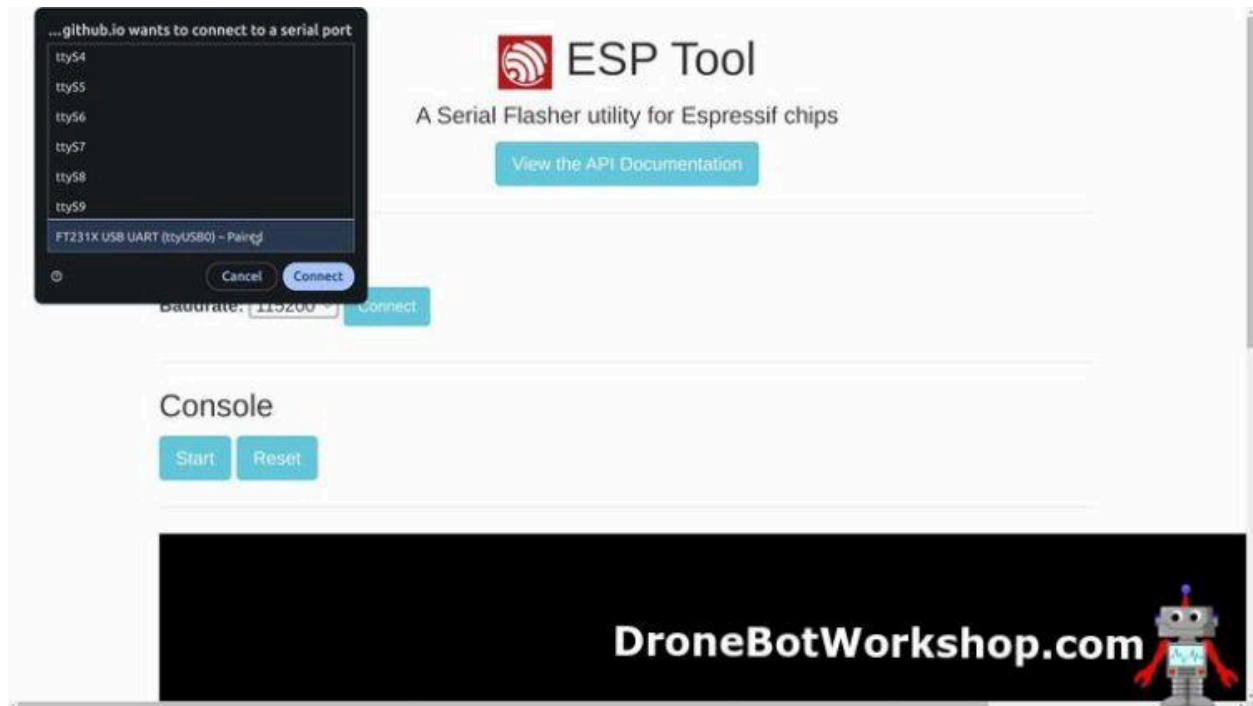


<https://dronebotworkshop.com>

For more projects and tutorials visit the DroneBot Workshop - <https://dronebotworkshop.com>

Now, connect an ESP32 board to the USB port on your computer. Once it is connected and powered up, click the *Connect* button on the ESP Tool page.

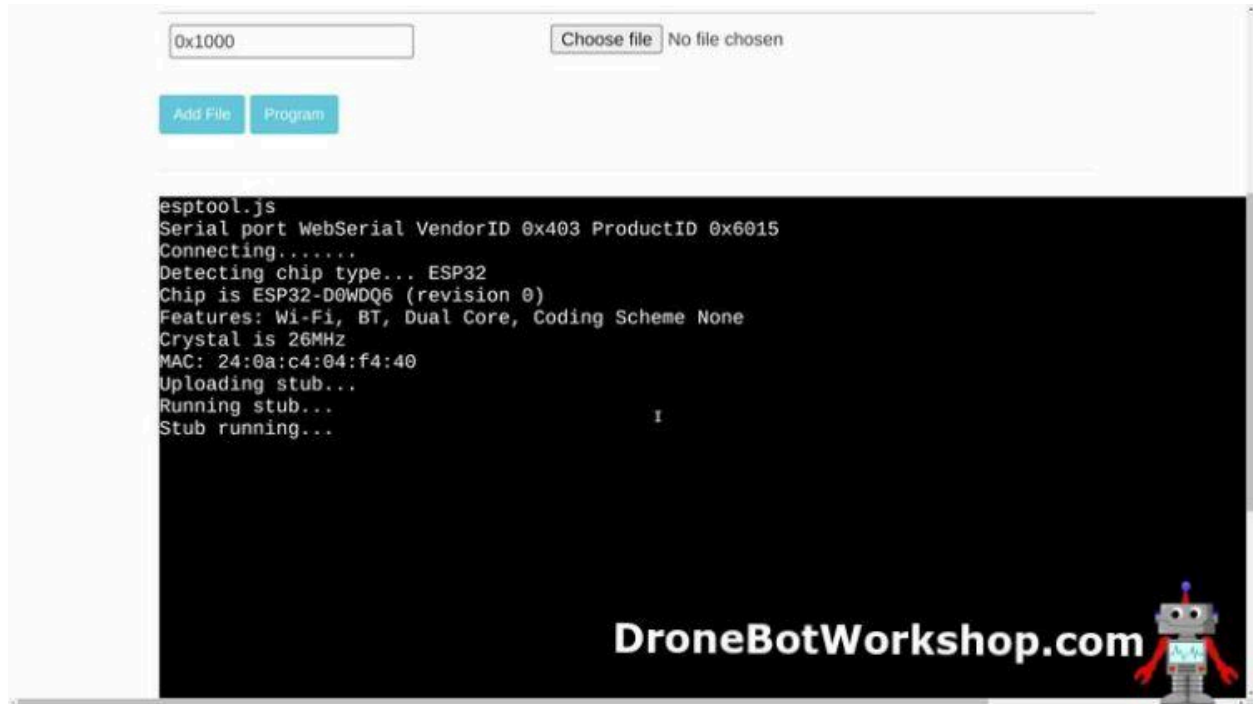
A dialog box will open with a multitude of serial ports. Look through it until you find your ESP32, it should stand out from all the rest and be pretty easy to spot. Select the board and click the *Connect* button within the dialog box.



Observe the output in the large dialog box; this is your terminal box. You should see some information about the ESP32 processor type, the crystal speed, and other assorted bits of wisdom.

<https://dronebotworkshop.com>

For more projects and tutorials visit the DroneBot Workshop - <https://dronebotworkshop.com>



ESP Tool, both the online version and the Python script, is a very useful utility. We'll be seeing more of it in future videos and articles.

Espressif ESP32 DevKits

DevKits are boards manufactured by Espressif Systems and used by hobbyists and developers. They range from boards with just an ESP32 module, USB interface, and not much more to boards with onboard peripherals like displays, interfaces, and audio circuitry.

All DevKit boards are supported with online multilingual documentation. They are excellent choices for experimenters and hobbyists.

Here are a couple of DevKits that you might want to add to your own parts drawer:

<https://dronebotworkshop.com>

Espressif ESP32-S3 DevKit-C

This is one of my favorite ESP32 development boards, if you could only have one board, this is the one I would recommend.



The reason I like this board so much is that it makes all the I/O pins and features of the ESP32-S3 available through its 44 pins. The board also has two USB-C connectors; you can use one to program and communicate with the board, while the other one can be used when the ESP32 emulates a USB device.

You can find [more information about using this board](#) from Espressif.

Espressif ESP32-C6 DevKit-C

If you'd like to start experimenting with Thread, Zigbee, or Matter, this development board will get you going at a meager cost.



This 32-pin module exposes 25 of the ESP32-C6 I/O pins. It also has dual USC-C connectors and an additional 8MB of Flash memory for nonvolatile data storage.

[Check the Espressif documentation](#) for more information about using this DevKit.

Third-Party ESP32 Development Boards

Espressif is not the only company that manufactures ESP32 development boards. There are literally hundreds of ESP32 boards made by dozens of manufacturers, a testament to the popularity and versatility of this amazing microcontroller.

For more projects and tutorials visit the DroneBot Workshop - <https://dronebotworkshop.com>

Creating an exhaustive list of development boards would be a huge task, well beyond what I'm willing to undertake! However, I can list a few third-party ESP32 development boards that you might find useful for your experiments.

Arduino Nano ESP32

We have looked at the Nano ESP32 in great detail before. It's another member of the growing Arduino Nano family, with the same pinout and form factor as the other Nano boards.



This common pinout, along with support from Arduino and a huge maker community, makes the Nano ESP32 an ideal first ESP32 board when you are starting out. The Nano pinout is very breadboard-friendly and allows you to reuse thousands of circuit diagrams intended for other Nano boards. Just remember that the Nano ESP32, like all other ESP32 boards, is a 3.3-volt logic device and is not 5-volt tolerant.

<https://dronebotworkshop.com>

For more projects and tutorials visit the DroneBot Workshop - <https://dronebotworkshop.com>

The design only exposes 24 of the ESP32-S3 pins, so if you need a lot of discrete I/O pins, this may not be the board for you. Most pins have two or three functions, so this lack of I/O pins won't be an issue for many designs.

LilyGo TTGO T-Display S3

I'm recommending this board due to the praise it receives from other users; I don't own one (yet) myself. I own several of the TTGO T-Display ESP32 boards, which I will review afterward.

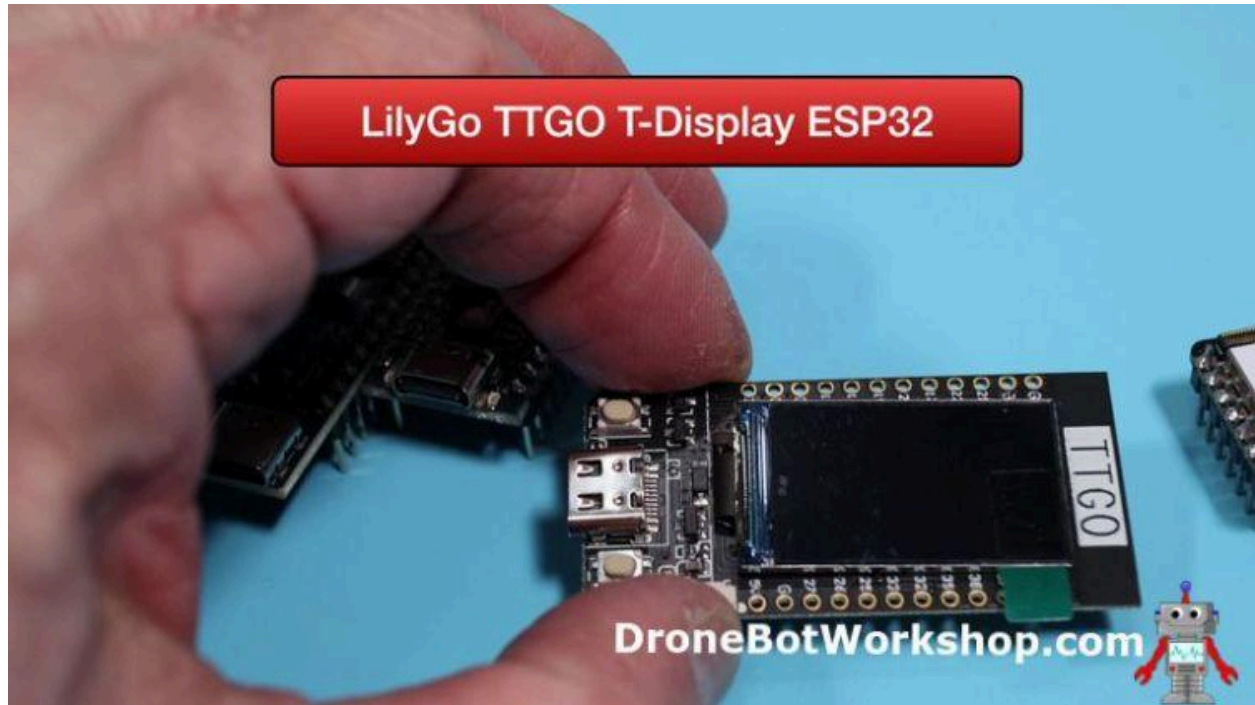
This board has many features that would make it a great general-purpose ESP32 developers board.

- 1.9-inch LCD TFT Touch Display
- 16 MB Flash memory
- LiPo battery connector with internal charging
- USB-C connector

The board has a 24-pin, breadboard-friendly connector. Only 13 of the I/O pins are exposed, but since it does have an internal touchscreen display, you probably won't need that many I/O pins for most designs.

LilyGo TTGO T-Display ESP32

This is the LilyGo board I own, and it is a great value at only eight dollars.



This is the only board on this list that uses the original ESP32. It also has a 1.14" TFT LCD (not a touchscreen) and supports an external LiPo battery. The pinout is similar to the previous LilyGo board, so if you require a lot of I/O pins, you might need to look elsewhere.

I used this board as a remote control in my Mecanum Wheel Robot project. It was simple to program, and the built-in support for a rechargeable LiPo battery was useful.

At such a low price, this is a board I recommend, especially if you require an LCD for your project. It's much simpler using this than it is to wire up your own display, and the quality of the screen is very good – sharp and easy to view at an angle.

Seeeduino XIAO ESP32S3

Here is another ESP32-S3 board that we have already looked at in detail.



The Seeeduino XIAO ESP32S3 is a tiny board with the same form factor as the other XIAO boards. This is one of the smallest ESP32 development boards you can get, but it has many features and a meager price tag.

The board is also available with an optional daughter card with a video camera. The card fits onto a small connector on the top of the board, and there is an option for a higher-quality camera.

Since it has the same XIAO form factor, you can use existing prototype boards and tools, so it has the same advantage as the Arduino Nano ESP32. It also has the same disadvantage, with only 11 I/O pins exposed.

For more projects and tutorials visit the DroneBot Workshop - <https://dronebotworkshop.com>

But if you don't require a lot of I/O pins and you need a small board (perhaps with camera capabilities), this is an excellent value.

Boards Summary

The following table summarizes the features of the ESP32 development boards we just looked at:

	ESP32-S3 DevKit	ESP32-C6 DevKit	Nano ESP32	T-Display-S3	T-Display ESP32	XIAO-ESP32S3
Price USD*	\$15.00	\$9.00	\$20.00	\$21.98	\$8.00	\$7.49
ESP32 Series	ESP32-S3	ESP32-C6	ESP32-S3	ESP32-S3	ESP32	ESP32-S3
I/O Pins Exposed	37	25	24	13	13	11
USB Type	2 x MicroUSB	2 x USB-C	USB-C	USB-C	USB-C	USB-C
Display Type	NONE	NONE	NONE	1.9" LCD Touch	1.14" TFT LCD	NONE
Package	44-Pin	32-Pin	30-Pin	24-Pin	24	14-Pin
LiPo Connector	No	No	No	Yes	Yes	Capable
Antenna	Internal	Internal	Internal	Internal	Internal	External
RTC	Yes	No	No	No	No	No
PSRAM	8MB	8MB	8MB	8MB	16MB	8MB
FLASH		8MB		16MB		8MB

SUBTOTAL * Price as of 2024-01-12

Choosing the best development board comes down to your personal requirements.

If you want as many I/O pins as possible, one of the Espressif DevKits is a good choice. If a small size is a requirement, then the Seeeduino XIAO is a great little board with a low price as well. The two LilyGo boards should satisfy the requirements for a board with a display.

For more projects and tutorials visit the DroneBot Workshop - <https://dronebotworkshop.com>

For those with an investment in Arduino Nano or Seeeduino XIAO prototyping tools, the board offerings from both companies are attractive.

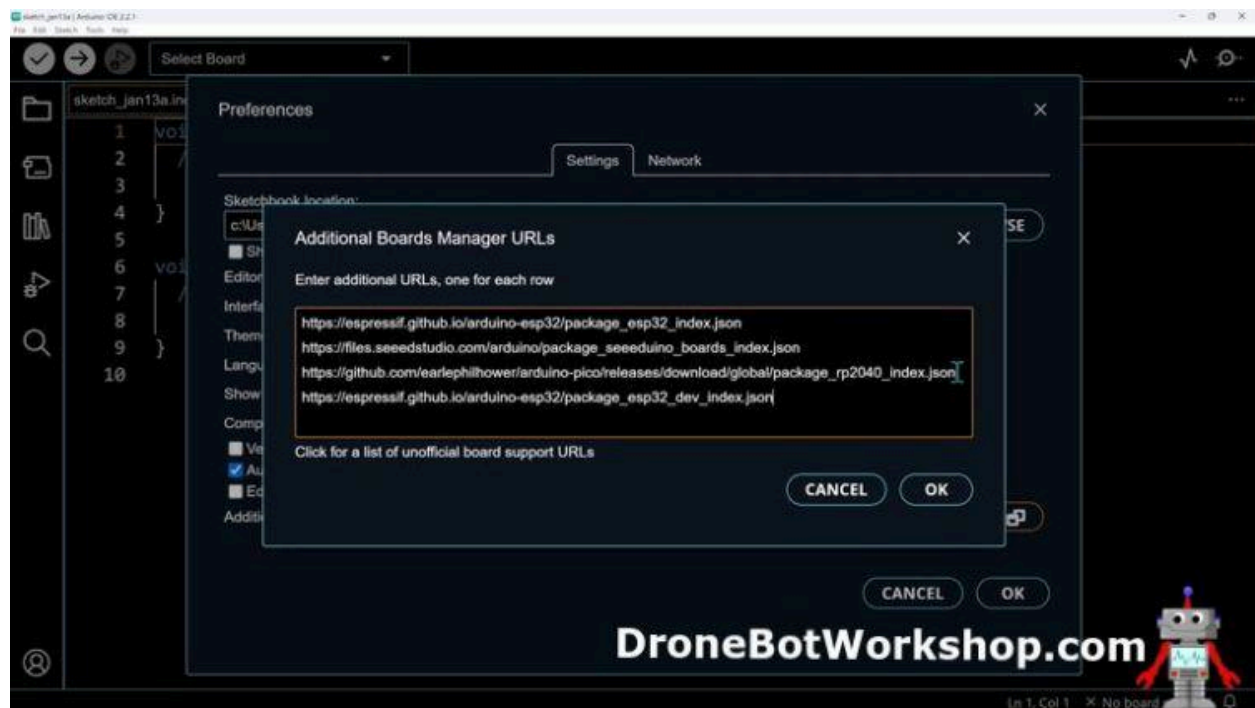
Of course, these six boards only scratch the surface, and there are hundreds more boards to choose from. In the video accompanying this article, I cover three more that might be worth your consideration.

ESP32 Boards Manager 3.0 Alpha

The Arduino IDE is the easiest way of working with the ESP32; at least, it is if you are coding in C++. The Espressif ESP Boards Manager is the key component in using the ESP32 with the Arduino IDE.

The Boards Manager provides all the drivers required to let the IDE work with various ESP32 boards. It also installs a collection of libraries and example sketches. However, the current version of the ESP Boards Manager does not support the newest ESP32 chips, like the C6 and H5.

To provide support for those boards, Espressif has released an Alpha of the new Boards Manager version 3. If you don't mind running alpha software, you can try it in your IDE.



Follow these steps to install the new boards manager:

- Open the Arduino IDE.

For more projects and tutorials visit the DroneBot Workshop - <https://dronebotworkshop.com>

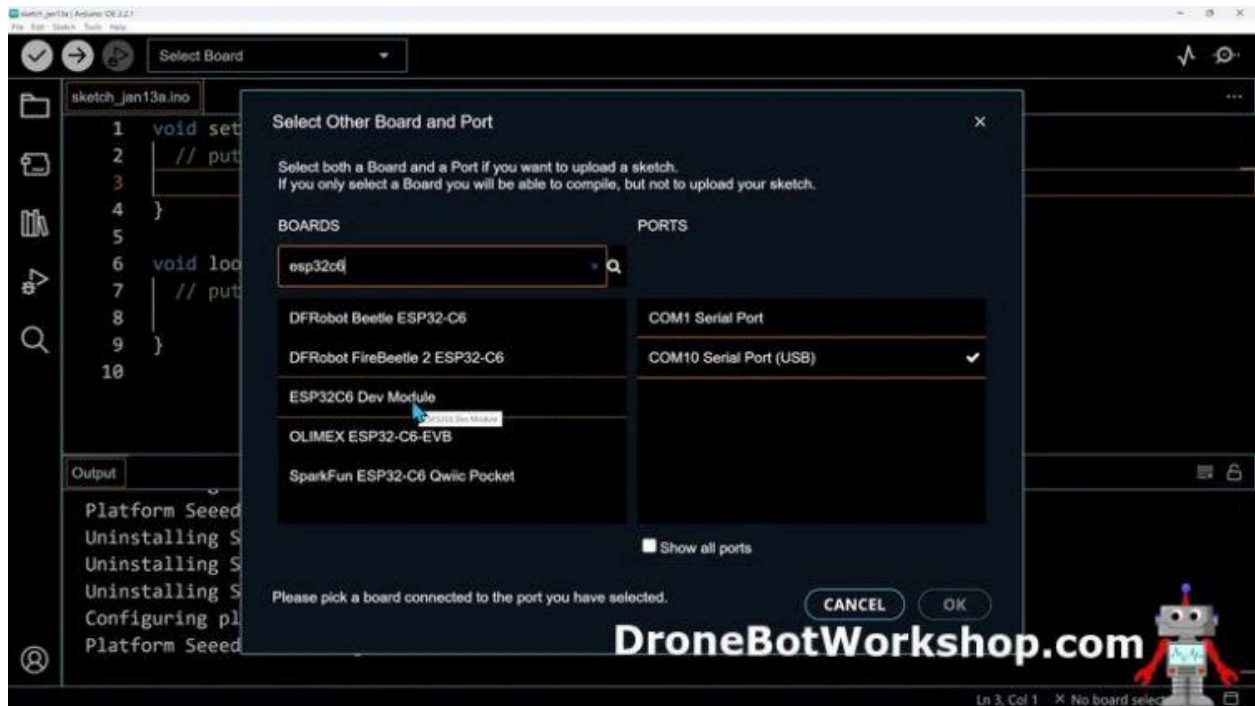
- Click on the *File* menu
- Select *Preferences*. A dialog box will open.
- Look for the text box labeled *Additional boards manager URLs*. Next to the text box, on the right side, is a button. Click on it.
- A modal dialog box will open with a large text box. The text box may be empty, or you may already have a list of JSON files in it.
- Add the following JSON URL to the list:

```
1 https://espressif.github.io/arduino-esp32/package\_esp32\_index.json
```

- Now click *OK* to save the entry and close the modal dialog box.
- Back in the Preference dialog box, click *OK* again to exit.
- Close the Arduino IDE (this step is important; don't omit it)
- Now open the IDE again.
- You should be prompted for updates to your Boards Manager (and possibly Libraries). Click *OK* to the prompt(s) to install the board manager files.
- Wait for the files to install; it may take several minutes.
- Once the files are installed, the operation is complete.

You can test the installation by searching for one of the ESP32-C6 boards. You should now display several of them.

For more projects and tutorials visit the DroneBot Workshop - <https://dronebotworkshop.com>



If, for some reason, you no longer wish to use the Alpha, then go into your Boards Manager and select the ESP32. Use the drop-down to select the older boards manager version. Once again, you should reboot and perform any updates.

Conclusion

The ESP32 has certainly grown since its introduction over seven years ago. The availability of multiple series and the wealth of communications and other features make the ESP32 one of the most powerful yet cost-effective microcontrollers today.

By picking up a few development boards and a few external components and sensors, you can start exploring, designing, and building your own advanced IoT applications, even on a restricted budget.

<https://dronebotworkshop.com>

For more projects and tutorials visit the DroneBot Workshop - <https://dronebotworkshop.com>

Hopefully, this article and its associated video have inspired you to pick up a DevKit or development board and start working with the ESP32 yourself. Who knows what you'll design with this magic microcontroller from Espressif?

Parts List

Here are some components you might need to complete the experiments in this article. Please note that some of these links may be affiliate links, and the DroneBot Workshop may receive a commission on your purchases. This does not increase the cost to you and is a method of supporting this ad-free website.

Espressif ESP32-S3 DevKit-C — [Digikey](#) — [Mouser](#)

Espressif ESP32-S3 DevKit-C — [Digikey](#) — [Mouser](#)

Arduino Nano ESP32 — [Arduino](#) — [Digikey](#)

Seeeduino XIAO ESP32S3 — [Seeed Studio](#) — [Mouser](#)

LilyGo TTGO T-Display-S3 — [LilyGo](#) — [Amazon](#)

LilyGo TTGO T-Display-ESP32 — [LilyGo](#) — [Amazon](#)

Resources

Article PDF – A PDF version of this article in a ZIP file.

[ESP Tool Online](#) – The online version of the ESP Tool utility. You must use a Chrome-based browser.

[Espressif ESP32](#) – ESP32 series from Espressif.