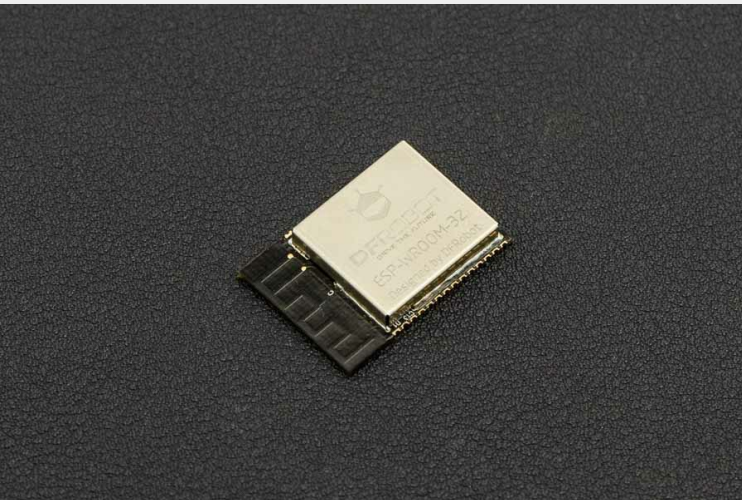


[Modules](#) / [Communications](#) / [WiFi](#) / [ESP32 \(ESP WROOM 32\) WiFi & Bluetooth Dual-Core MCU Module](#)

## ESP32 (ESP WROOM 32) WiFi & Bluetooth Dual-Core MCU Module

SKU:TEL0111

Brand:DFRobot

Reward Points: 65

### Quantity Based

| QTY | DISCOUNT |
|-----|----------|
|-----|----------|

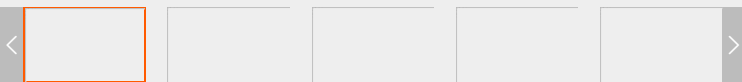
5-9

10-49

50+

Quantity: 

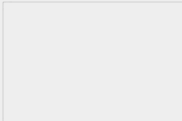
ADD TO CART



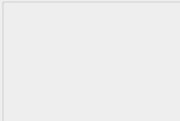
### Frequently Bought Together



+



+



>

ESP32 (ESP WROOM 32) WiFi & Bluetooth Dual-Core MCU Module

FireBeetle-3 WiFi & Bluetooth Module

2x ESP32 Module for ESP32



You have chosen:0  
Total amount:

BUY IT NOW

## INTRODUCTION

The ESP32 WiFi and Bluetooth chip is the latest generation of Espressif products. It has a dual-core 32-bit MCU, which integrates WiFi HT40 and Bluetooth/BLE 4.2 technology inside.

ESP32 WiFi & Bluetooth Dual-Core MCU Module

Compared to the [arduino ESP8266](#) (the previous generation), the ESP32 wifi and bluetooth chip (also known as ESP wroom 32) has a significant performance improvement. It is equipped with a high-performance dual-core Tensilica LX6 MCU. One core handles high speed connection and the other for standalone application development. The dual-core MCU has a 240 MHz frequency and a computing power of 600 DMIPS.

In addition, it supports [Wi-Fi](#) HT40, Classic [Bluetooth/BLE](#) 4.2, and more GPIO resources.

ESP32 WiFi & Bluetooth Dual-Core MCU Module Functionality

ESP32 chip (ESP wroom 32) integrates a wealth of hardware peripherals, including [capacitive touch sensors](#), [Hall sensors](#), low noise sensor amplifiers, [SD card](#) interfaces, Ethernet interfaces, high-speed SDIO / SPI, UART, I2S and I2C, etc.

ESP32 WiFi & Bluetooth Dual-Core MCU Module Pins

Support Arduino Programming

Engineered for mobile devices, [wearable](#) electronics and [Internet of Things \(IoT\) applications](#), the ESP32 wroom 32 achieves ultra-low power consumption with a combination of several proprietary software applications. The state-of-the-art power saving features include fine resolution clock gating, power modes, and dynamic power scaling.

ESP32 WiFi & Bluetooth Dual-Core MCU Module Ultra Low Power Management

ESP32 WiFi & Bluetooth Dual-Core MCU Module Schematic

DFRobot introduces the ESP-WROOM-32 module, based on the Espressif ESP32 chip. It has integrated electronic peripherals, external expansion of all GPIO pins, an onboard antenna, stamp hole design and easy mass production. It can be used for product development, integrated design and many more applications.

## Arduino IDE + Minimal HW Setup for ESP-WROOM-32



Revisión módulo wifi Bee de Dfrobot, encendiendo 12 led por wifi



## ESP32 Arduino Tutorials List:

- [Arduino IDE for Building Compiling Environment of ESP32](#)
- [FireBeetle Board-ESP32 Projects - Let your cigarette lighter wearable](#)
- [How to Make a Low-Power-Consumption Fruit Battery By Use Of A Firebeetle Board-ESP32](#)
- [ESP32 Tutorial: MicroPython support](#)
- [How to Make a Smart Colorful Light Based On FireBeetle ESP32 IOT Microcontroller](#)
- [ESP32 MicroPython Tutorial: Parsing JSON](#)
- [FireBeetle Board-ESP32 Tutorials: Reforming a Small electric Fan](#)
- [ESP32 MicroPython Tutorial: Encoding JSON](#)
- [Application Tutorials of FireBeetle Board - ESP32 BLE: Smarten a Plugboard](#)
- [ESP32 / ESP8266 MicroPython Tutorial: HTTP GET Requests](#)
- [ESP32 / ESP8266 MicroPython Tutorial: Automatic connection to WiFi](#)
- [ESP32 MicroPython Tutorial: Using SHA-256](#)
- [ESP32 / ESP8266 MicroPython: HTTP POST Requests](#)
- [ESP32 MicroPython Tutorial: Executing scripts with uPyCraft](#)
- [ESP32 MicroPython Tutorial: Getting started with the uPyCraft IDE](#)
- [ESP32 / ESP8266 MicroPython Tutorial: Working with lists](#)
- [ESP32 / ESP8266 MicroPython Tutorial: Lambda functions](#)
- [ESP32 / ESP8266 MicroPython Tutorial: Applying filter function to lists](#)
- [ESP32 / ESP8266 MicroPython Tutorial: Working with dictionaries](#)
- [ESP32 / ESP8266 MicroPython Tutorial: Applying map function to lists](#)
- [ESP32 MicroPython Tutorial: Changing the HTTP response content-type of Picoweb route](#)
- [ESP32 MicroPython Tutorial: HTTP Webserver with Picoweb](#)
- [ESP32 Arduino Tutorial: FreeRTOS queues performance test](#)
- [ESP32 MicroPython Tutorial: Serving HTML from the file system in Picoweb](#)
- [ESP32 Picoweb Tutorial: Obtaining the HTTP Method of the request](#)
- [ESP32 MicroPython Tutorial: Developing a simple URL query string parser](#)

[ESP32 / ESP8266 MicroPython Tutorial: String split method](#)

[ESP32 MicroPython Tutorial: Passing arguments to a thread function](#)

[ESP32 MicroPython Tutorial: Timer interrupts](#)

[ESP32 MicroPython Tutorial: Getting the query parameters on a Picoweb app](#)

[ESP32 MicroPython Tutorial: Creating a thread](#)

[ESP32 Arduino tutorial: Websocket server over soft AP](#)

[ESP32 Arduino Tutorial: Websocket server](#)

[ESP32 Tutorial \(Arduino Websocket server\): Receiving and parsing JSON content](#)

[ESP32 MicroPython Tutorial: External interrupts](#)

[ESP32 Arduino Tutorial: Asynchronous HTTP webserver](#)

[ESP32 RFID Tutorial: Printing the MFRC522 firmware version](#)

[ESP32 Arduino Tutorial: Websocket client](#)

[ESP32 Arduino SSD1306 OLED: Redrawing a string](#)

[ESP32 Arduino Tutorial: Software reset](#)

[ESP32 Arduino Tutorial: Interacting with a SSD1306 OLED display](#)

[ESP32 / ESP8266 Arduino: Serial communication with Python](#)

[ESP32 Arduino SSD1306 OLED: Drawing a QR Code](#)

[ESP32 Arduino Tutorial: Base64 encoding](#)

[ESP32 Arduino async HTTP server: Serving a HTML page from PROGMEM](#)

[ESP32 Arduino Tutorial async HTTP server: Serving HTML](#)

[ESP32 Arduino Tutorial HTTP server: Running multiple server instances](#)

[ESP32 Arduino Tutorial HTTP server: Getting query parameters](#)

[ESP32 Arduino Tutorial: Random Number Generation](#)

[ESP32 Arduino Tutorial: Getting the Free Heap](#)

[ESP-WROOM-32 Tutorial: Uploading a program with Arduino IDE](#)

[ESP32 Arduino Tutorial: External interrupts](#)

[ESP32 Arduino Tutorial: Using the pthreads library](#)

[ESP32 Tutorial Getting started with the aREST library](#)

[ESP32 Arduino HTTP server: external and internal redirects](#)

[ESP32 Tutorial: ESP32 Arduino HTTP server: route not found handling](#)

[ESP32 Arduino: HTTP server over soft AP](#)

[ESP8266 Arduino Tutorial: Software restart](#)

[ESP32 Arduino Tutorial: Basic Authentication](#)

[ESP32 Projects: Mail Alarm](#)

[ESP32 Arduino async server Tutorial: Controlling HTTP methods allowed](#)

[ESP32 Arduino HTTP Server: Serving HTML and JavaScript](#)

[ESP32 Arduino Tutorial: Serial communication over Bluetooth Hello World](#)

[ESP32 Arduino Bluetooth Classic: Setting the device name](#)

[ESP32 Arduino Bluetooth over Serial: Receiving data](#)

[ESP32 Arduino: Getting the Bluetooth Device Address](#)

[ESP32 HTTP server: Multiple methods allowed on same route](#)

[ESP32 Arduino Tutorial: Variable length arrays](#)

[ESP32 Arduino web server: getting client IP](#)

[ESP32 Arduino Tutorial: Controlling a relay](#)

[ESP32 Arduino: Using structs as items in FreeRTOS queues](#)

[Mongoose OS on ESPer/Bare ESP32](#)

[ESP32 Arduino: Encryption using AES-128 in ECB mode](#)

[ESP32 Arduino: Getting temperature from a DHT22 sensor](#)

[ESP32 Arduino Bluetooth: Finding the device with Python](#)

[ESP32 / ESP8266 Arduino: Initializer list](#)

[ESP32 / ESP8266 Arduino: Bitwise shift operators](#)

[ESP32 / ESP8266 Tutorial: Class constructors](#)

[ESP32 Arduino web server: Get free heap and reset device remotely](#)

[ESP32 Arduino Tutorial: Getting weather data from API](#)

[ESP32 / ESP8266 Arduino Tutorial: Using structs](#)

[ESP32 / ESP8266 Arduino: The typedef keyword](#)

ESP32 Espruino: convert number to string in different bases

ESP32 Arduino: Applying the HMAC SHA-256 mechanism

ESP32 Espruino Tutorial: Array every method

ESP32 Espruino Tutorial: Creating simple JavaScript objects

ESP32 Espruino: Array map method

ESP32 Espruino Tutorial: Applying the SHA1 hash algorithm

ESP32 Espruino Tutorial: Scanning WiFi Networks

ESP32 Picoweb Tutorial: Serving JSON content

ESP32 / ESP8266 Arduino: The auto keyword

ESP32 Arduino Tutorial: Communication between tasks using FreeRTOS queues

ESP32 / ESP8366 Arduino: Creating a simple class

ESP32 Arduino Tutorial: FreeRTOS Queues

ESP32 Arduino Tutorial: Getting available messages and free spaces of FreeRTOS queue

ESP32 IDF Tutorial: Setting a soft AP

ESP32 Arduino Tutorial: Inserting in front and back of FreeRTOS queues

ESP32 Arduino tutorial: Internal Hall sensor measurements

ESP32 Arduino Tutorial: Setting a soft AP

ESP32 Bluetooth Tutorial: Receiving data through RFCOMM

ESP32 Bluetooth tutorial: Advertising a SPP service with SDP

ESP32 Arduino Tutorial: HTTP POST Requests to Bottle application

ESP32 Bluetooth Tutorial: Finding the device with Python and BTStack

ESP32 Arduino Tutorial: Controlling a buzzer with PWM

ESP32 Bluetooth Tutorial: Using the BTstack library

FireBeetle ESP32 Tutorial: Blinking the on board LED

ESP32 Arduino Tutorial: Getting started with WiFi

FireBeetle ESP32 Arduino Tutorial: Hello World

ESP32 Arduino Tutorial: LED PWM fading

ESP32 Arduino tutorial: Getting humidity measurements from a DHT22 sensor

ESP32 / ESP8266 MicroPython Tutorial: Running a script from the file system

ESP32 Arduino: DHT22 Minimum Sampling Period

ESP32 / ESP8266 MicroPython Tutorial: Uploading files to the file system

ESP32 / ESP8266 MicroPython Tutorial: Writing a file

ESP32 / ESP8266 MicroPython Tutorial: Reading a file

ESP32 Arduino Tutorial: Setting a socket server

ESP32 Arduino: Using an infrared CO2 sensor

ESP32 Arduino: Getting the SDK version

ESP32 Socket Server: Connecting from a Putty socket Client

ESP32 Arduino: Temperature, humidity and CO2 concentration web server

ESP32 Arduino Tutorial: Decrypt AES-128 in ECB mode

ESP32 Arduino Tutorial: Sending data with socket client

ESP32 Arduino Tutorial Socket server: Getting remote client IP

ESP32 Arduino Tutorial mbed TLS: using the SHA-256 algorithm

ESP32 Espruino Tutorial: Listing property names of objects

ESP32 Espruino Tutorial: Converting an object to a JSON string

ESP32 Espruino Tutorial: Deserializing JSON

ESP32 Arduino Tutorial: Obtaining MAC address of stations connected to soft AP

ESP32 Espruino Tutorial: Get SDK version, free heap and software reset

ESP32 Espruino Tutorial: Adding properties to previously created object

ESP32 / ESP8266 Arduino Tutorial: Bitwise AND operator

ESP32 Espruino DHT22 Tutorial: Getting temperature and humidity

ESP32 / ESP8266 Arduino Tutorial: Conditional operator

Make an HTTP request to get weather info with Zerynth

ESP32 JavaScript Espruino Tutorial: String interpolation

Raspberry Pi Flask Tutorial: Receiving HTTP GET Request from ESP32

Raspberry Pi 3 Flask Tutorial: Receiving HTTP POST Request from ESP32

ESP32 Arduino Tutorial: PIR motion sensor

## SPECIFICATION

### CPU and Memory

- Xtensa® Dual-Core 32-bit LX6 microprocessors, up to 600 DMIPS
- 448 KByte ROM
- 520 KByte SRAM
- 16 KByte SRAM in RTC
- QSPI Flash/SRAM, up to 4 x 16 MBytes
- Power supply: 2.2 V to 3.6 V

### Clocks and Timers

- Internal 8 MHz oscillator with calibration
- Internal RC oscillator with calibration
- External 2 MHz to 40 MHz crystal oscillator
- External 32 kHz crystal oscillator for RTC with calibration
- Two timer groups, including 2 x 64-bit timers and 1 x main watchdog in each group
- RTC timer with sub-second accuracy
- RTC watchdog

### Advanced Peripheral Interfaces

- 12-bit SAR ADC up to 18 channels
- 2 × 8-bit D/A converters
- 10 × touch sensors
- Temperature sensor
- 4 × SPI
- 2 × I2S
- 2 × I2C
- 3 × UART
- 1 host (SD/eMMC/SDIO)
- 1 slave (SDIO/SPI)
- Ethernet MAC interface with dedicated DMA and IEEE 1588 support
- CAN 2.0
- IR (TX/RX)
- Motor PWM
- LED PWM up to 16 channels
- Hall sensor
- Ultra low power analog pre-amplifier

### Development Support

- SDK Firmware for fast on-line programming
- Open source toolchains based on GCC

**Operating Temperature Range: -40 ° C to + 85 ° C**

**Package size: 18mm x 25mm x 3mm**

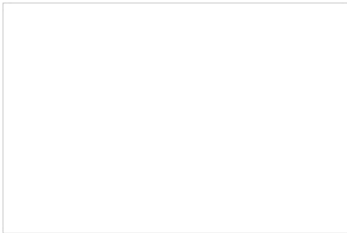
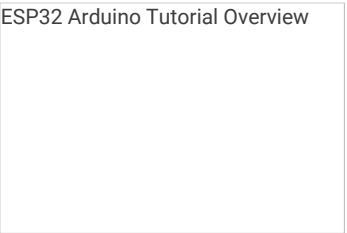
## DOCUMENTS

- [DFRobot ESP32 Module Schematic](#)
- [DFRobot ESP32Module layout](#)
- [ESP32 Arduino Program](#)
- [ESP\\_WROOM\\_32 Datasheet](#)
- [ESP32 Technical Reference Manual](#)
- [ESP32 Resource Download](#)

## SHIPPING LIST

- ESP32 WiFi & Bluetooth Dual-Core MCU Module x1

## TUTORIAL

|                                                                                    |                                                                                     |                                                                                      |
|------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------|
|  |  |  |
| FireBeetle Board-ESP32 Projects - ...<br>ESP32                                     | The Best ESP32 Projects Overview<br>ESP32                                           | ESP32 Arduino Tutorial Overview<br>ESP32                                             |
| 2017-06-07 01:50:46                                                                | 2017-08-10 08:12:01                                                                 | 2018-04-27 08:15:22                                                                  |

## REVIEW

Downloaded from [Arrow.com](#)

# FAQ

22 Comments   DFRobot    Disqus' Privacy Policy

 Login

 Recommend

 Tweet

 Share

Sort by Best

 n the discussion...



**mellott124** • a year ago

Where's the datasheet for this specific module? Your images are blurry and I cannot see the IO.

^ • Reply • Share >



**DFRobot Support**  mellott124 • a year ago

You could browse the SPECIFICATION.

^ • Reply • Share >



**mellott124** → DFRobot Support • a year ago

No, the specification is not sufficient. I need a datasheet with pin definitions, electrical characteristics, etc. You can't expect people to integrate this into their designs without a proper datasheet.

1 ^ • Reply • Share >



**DFRobot Support**  mellott124 • a year ago

Hi, we will put the datasheet on the web, please wait for a few days.

^ • Reply • Share >



**Marshal Horn** • 3 years ago

Do you have a recommended pad layout for this module?

It's difficult to tell the pin spacing from the "module layout" pdf. Is it 1mm pads with 1.3mm spacing?

^ • Reply • Share >



**Marshal Horn** → Marshal Horn • 3 years ago

In case anyone is still wondering, I pulled measurements off the PDF using Inkscape.

The pads are 0.04" wide, with 0.05" spacing. In millimeters, that's 1.02mm pads with 1.27mm spacing.

This means you can connect standard 0.1" header by skipping every other pin, as shown in this blog post

<https://techtutorialsx.com/>...



see more

^ • Reply • Share >



**DFRobot Support**  Marshal Horn • 3 years ago

We don't have this kind of spacing, it measured like 1mm pads and 1.3mm spacing.

^ • Reply • Share >

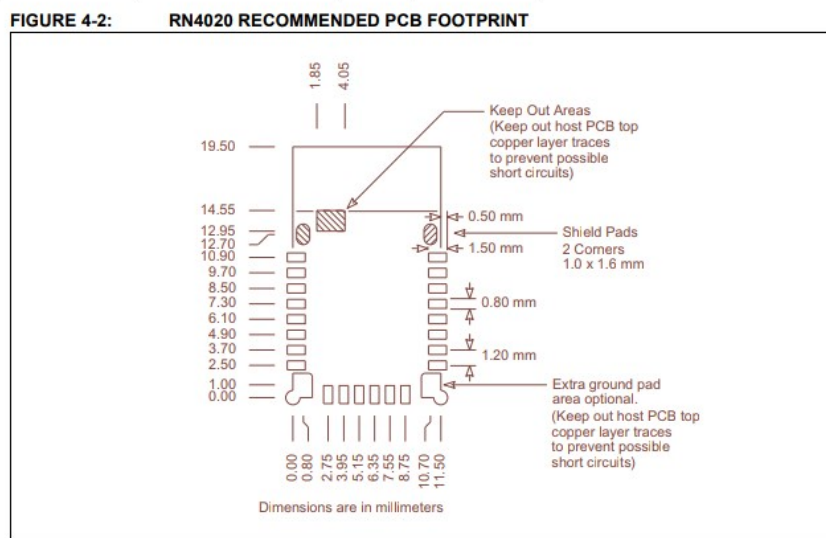




Marshal Horn → DFRobot Support • 3 years ago

Can you provide a diagram?

Here's an example of the info I'm looking for. Do you have a diagram like this for the TEL0111 module?

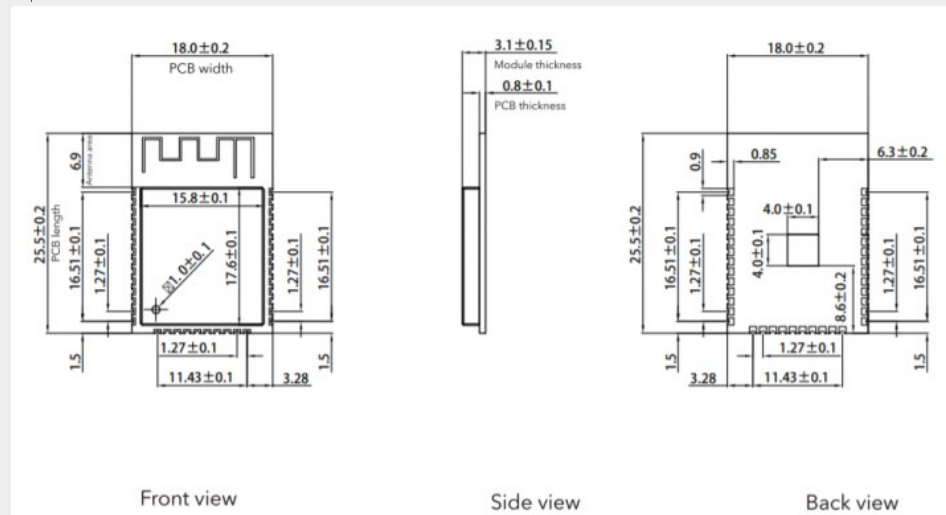


↑ • Reply • Share •



DFRobot Support **Mod** → Marshal Horn • 3 years ago

Hi, I found a dimension here



↑ • Reply • Share •



omar rincon • 3 years ago

Lamento el hecho que DFRobot de la opción de pre-compra pero no envíen las unidades después de pagar, ha pasado ya casi un año desde que pague por este producto y nunca llegó nada a mi domicilio.

↑ • Reply • Share •



DFRobot Support **Mod** → omar rincon • 3 years ago

Could you translate it to English?

↑ • Reply • Share •



omar rincon → DFRobot Support • 3 years ago

I regret the fact that DFRobot of the pre-purchase option but do not send the units after payment, it has been almost a year since I paid for this product and nothing ever came to my home.

↑ • Reply • Share •



DFRobot Support **Mod** → omar rincon • 3 years ago

Sorry, we cannot find your order according to your name or this email address. Please tell us your order ID (only five digits) first. Thanks!

↑ • Reply • Share •





omar rincon → DFRobot Support • 3 years ago

# 81573

Thank you !!

^ • Reply • Share >



DFRobot Support **Mod** → omar rincon • 3 years ago

Hi, we found your order detail. Please contact our sales by email: store@dfrobot.com, they will help your with your order, thank you!

^ • Reply • Share >



mark • 4 years ago

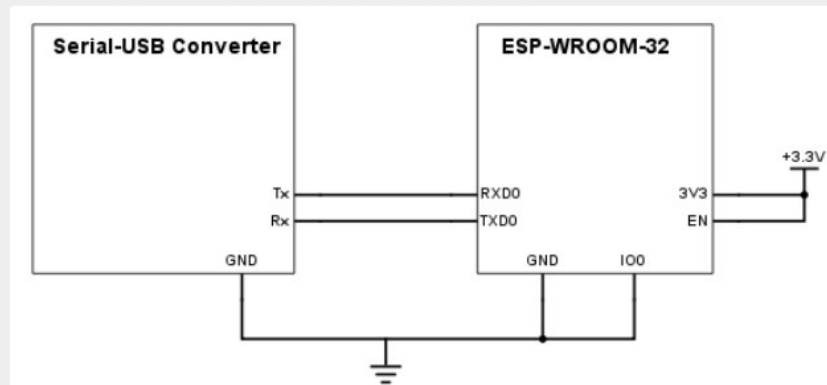
How is this ESP32 arduino support developing?

^ • Reply • Share >



DFRobot Support **Mod** → mark • 4 years ago

The ESP32's working voltage is 3.3V, and it has a microcontroller mounted in the module. You just need a serial-USB converter and choose FireBeetle-ESP32 board in the Arduino IDE, then you can develop in the Arduino environment. The connection between the converter and ESP32 looks like this:



You can have more information from the link: [https://techtutorialsx.com/...](https://techtutorialsx.com/)

^ • Reply • Share >



mark • 4 years ago

Is there any ESP32 tutorial for us to study?

^ • Reply • Share >



DFRobot Support **Mod** → mark • 4 years ago

Check out this website: [https://www.dfrobot.com/blog/...](https://www.dfrobot.com/blog/)

It's about using Arduino IDE to compile ESP32, for further study you can read the ESP32 Technical Reference Manual and ESP32 Arduino Program

^ • Reply • Share >



haha • 4 years ago

What are some software development tools for esp32 (successor to ESP8266)?

^ • Reply • Share >



CherudimLeaf → haha • 4 years ago

See this [https://techtutorialsx.com/...](https://techtutorialsx.com/)

^ • Reply • Share >



haha → CherudimLeaf • 4 years ago

thank you, it's useful for me

^ • Reply • Share >





## INFORMATION

---

[About Us](#)  
[Warranty](#)  
[Privacy Policy](#)  
[Shipping](#)  
[Payment](#)  
[FAQ](#)

## CUSTOMER SERVICE

---

[DFRobot Distributors](#)  
[Contact us](#)  
[Site Map](#)

## MY ACCOUNT

---

[Affiliates](#)  
[Specials](#)  
[Coupon](#)