

ELEGOO

PART 01

First Look ESP32-English

LESSON 02

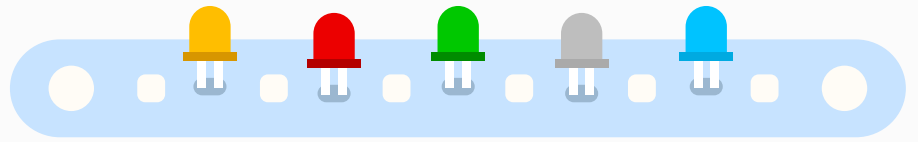


SCAN QR CODE ▶▶

GET THE TUTORIAL



01



What Is the ESP32?

The ESP32 is a family of highly integrated Wi-Fi (802.11 b/g/n) and Bluetooth (Classic + BLE 4.2/5.x) System-on-Chip (SoC) solutions developed by Espressif Systems (Shanghai, China).



ESP32-based modules and development boards excel at reading a wide variety of inputs:

- 1 Environmental data (via sensors connected to ADC or digital bus interfaces)
- 2 Biometric and motion signals (via I²C/SPI sensors such as heart-rate or IMU modules)
- 3 Audio and image data (via dedicated peripherals or expansion boards)
- 4 Remote network commands (via Wi-Fi or BLE)
- 5 Driving motors, servos, and actuators (using PWM or dedicated driver interfaces)
- 6 Uploading data to cloud platforms (MQTT/HTTP with AWS IoT, Aliyun, Azure IoT Hub, etc.)
- 7 Rendering information on displays (I²C OLED displays, SPI TFT screens)
- 8 Performing audio playback and simple voice-based interaction (via integrated DAC)

- Common development frameworks include Arduino Core for ESP32, MicroPython, and ESP-IDF (the official Espressif development framework). For beginners, the Arduino IDE with the ESP32 core provides the simplest entry point while retaining access to advanced ESP32 peripherals.
- Thanks to a strong global developer community, the ESP32 ecosystem provides abundant open-source libraries, tutorials, and sample projects—such as ESPAsyncWebServer for network applications or Adafruit SSD1306 for OLED displays—making rapid prototyping easier than ever.

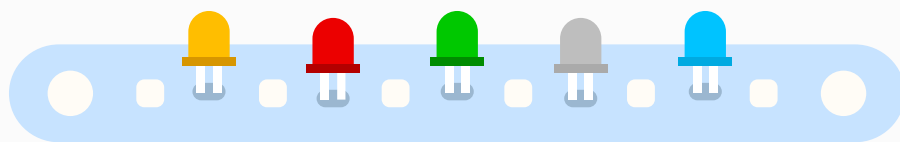
WHETHER YOU WANT TO BUILD:



- A portable weather station that syncs real-time data to your phone via Wi-Fi.
- A smart-home controller with BLE and voice commands.
- A battery-powered sensor node that runs for months.
- A Wi-Fi-controlled obstacle-avoidance robot.
- A wearable device for health monitoring and BLE notifications.

The ESP32 gives you all the hardware foundations to turn ideas into working prototypes.

02



Evolution of the ESP32

Espressif Systems, founded in 2008, released the ESP8266 in 2014—a breakthrough, low-cost Wi-Fi SoC that reshaped the IoT landscape. To address demand for multi-protocol connectivity and higher processing performance, Espressif officially introduced the ESP32 series in 2016.

The first mass-produced model, ESP32-D0WDQ6, featured a dual-core Tensilica Xtensa LX6 32-bit processor with integrated Wi-Fi and Bluetooth, filling functional gaps left by the ESP8266.

THE ESP32 PRODUCT FAMILY :



- **ESP32-S Series**

Optimized for low-power applications (e.g., ESP32-S3 with a RISC-V co-processor), ideal for wearables and battery-powered devices.

- **ESP32-C Series**

Cost-optimized and available with integrated flash, suitable for mass-production designs.

- **ESP32-H Series**

High-performance multimedia support (e.g., BLE 5.3-capable ESP32-H2), targeting cameras, displays, and edge-AI devices.

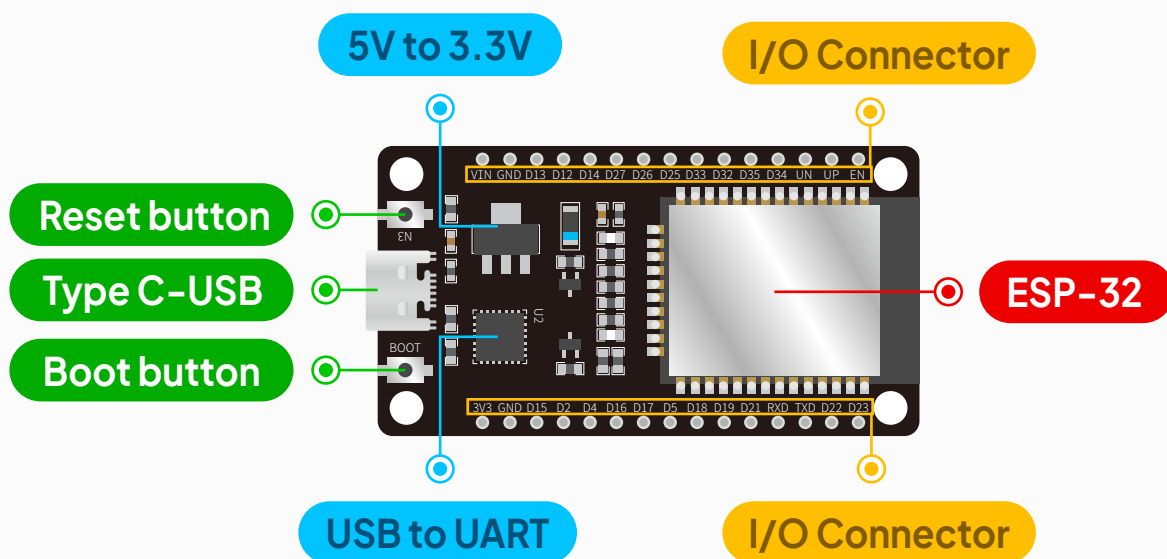
- A key milestone occurred in 2017 with the release of **the Arduino Core for ESP32**, enabling millions of Arduino developers to adopt ESP32 without learning a new programming environment. Today, the ESP32 is widely regarded as the de-facto standard for mid-range IoT prototyping.

03

ESP32 Board Overview

This section uses the **ESP32 DevKit V1** as an example to explain the core components and essential pin functions.

MAIN COMPONENTS:



(ESP32 board display image)

- **ESP32 Module**

Dual-core 32-bit processor with Wi-Fi and Bluetooth capability.

- **Power Regulation Circuitry**

Converts USB or Vin supply to stable 3.3V.

- **USB-to-UART Bridge**

Provides programming and serial communication via USB.

POWER SPECIFICATIONS:



The ESP32 DevKit V1 supports two primary power-input methods:

- **USB Type-C / Micro-USB**

Ideal for development; supplies 5V input and enables code uploading.

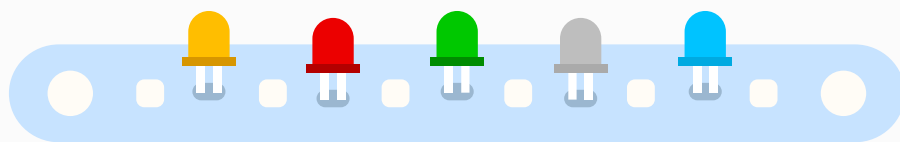
- **Vin pin (5–12V)**

Accepts external DC input; the onboard LDO regulator converts it to 3.3V for the ESP32.

Important Notes

1. Do NOT exceed 12V on Vin, or the regulator and ESP32 module may be damaged.
2. All ESP32 GPIO pins operate at 3.3V logic. Do not directly connect 5V signals—use a level shifter when necessary.

04



Important Pin Functions

ESP32 pins are configurable through software (multiplexed peripherals). Below is an organized:

(1) POWER & GROUND PINS



- **3V3 – 3.3V Regulated Output**

3.3V regulated output; useful for powering low-current sensors and modules (up to ~500mA).

- **GND – Multiple Ground Points**

All external modules must share GND.

- **Vin**

External 5–12V power input.

(2) GENERAL-PURPOSE GPIO PINS



- **Examples:**

GPIO15 (D15)	Supports PWM, SPI functions.
GPIO2 (D2)	ADC-capable; must remain high at boot to avoid startup issues.
GPIO4 (D4)	GPIO with touch and ADC support.
GPIO5 (D5)	Often used as VSPI clock.

GPIO13/12/14	Versatile GPIOs supporting PWM, SPI, and touch (but note GPIO12 affects boot voltage settings).
--------------	---

(3) UART PINS



- **GPIO16 (RX2) / GPIO17 (TX2)**

UART2 communication or general I/O.

- **GPIO3 (RX0) / GPIO1 (TX0)**

UART0 (USB serial) used for programming and debugging.

(4) ADC / DAC / TOUCH PINS



- **GPIO27 / GPIO32 / GPIO33**

ADC inputs, some with touch sensing.

- **GPIO34 / GPIO35 / GPIO36 / GPIO39 (VP/VN)**

ADC-only, input-only pins.

- **GPIO25 / GPIO26**

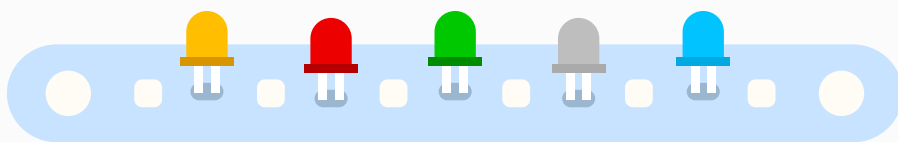
Support both ADC input and DAC output.

(5) BUS INTERFACE PINS



I ² C	GPIO21 (SDA), GPIO22 (SCL)
SPI (VSPI)	GPIO18 – SCK GPIO19 – MISO GPIO23 – MOSI GPIO5 – CS
EN pin	Chip-enable; Active-high.

05



Advantages of the ESP32 DevKit

ESP32 pins are configurable through software (multiplexed peripherals). Below is an organized:

(1) POWER & GROUND PINS



- **Extensive hardware expandability**

Multiple GPIOs and peripheral interfaces suitable for diverse applications.

- **Rich software ecosystem**

supports Arduino, MicroPython, and ESP-IDF; large library availability improves development efficiency.

- **Strong community & documentation**

Numerous open-source examples and tutorials accelerate learning and project development.

Further Reading

For more information, refer to Espressif's official documentation:

<https://docs.espressif.com/projects/esp-dev-kits/en/latest/esp32/>



FCC Warning

This device complies with part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) This device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

Any Changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment.

Note: This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

This equipment complies with FCC radiation exposure limits set forth for an uncontrolled environment. This equipment should be installed and operated with minimum distance 20cm between the radiator & your body.