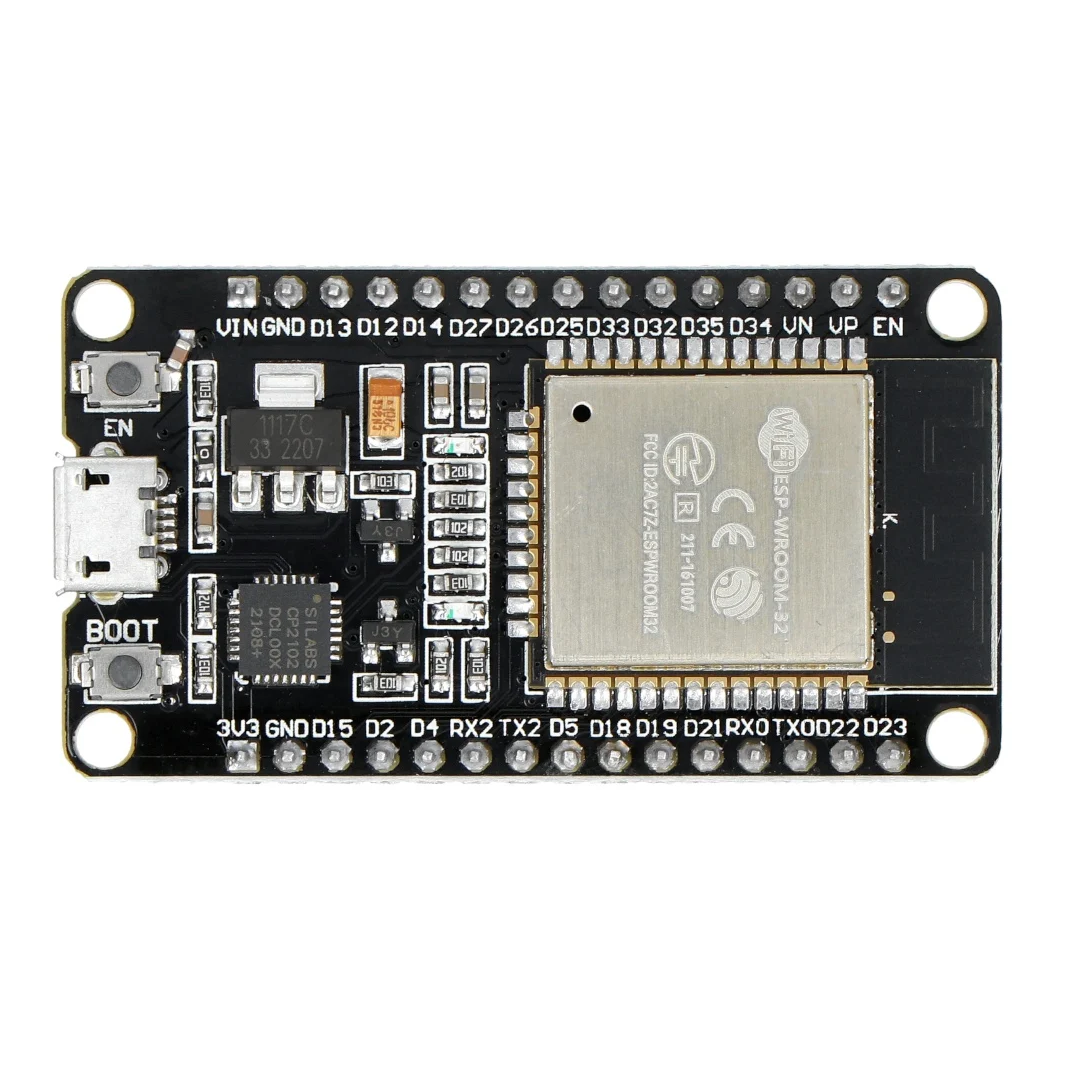
ESP32 Devkit v1

Introduction

The **ESP32 DevKit V1** is an advanced development board powered by the **ESP32 chip**, which is produced by Espressif Systems. It’s a dual-core microcontroller designed for a variety of IoT (Internet of Things) applications, offering Wi-Fi and Bluetooth connectivity in a single chip. The ESP32 is known for its high performance, low-power consumption, and rich set of features, making it one of the most popular development platforms for embedded systems.

At the heart of the ESP32 DevKit V1 is the ESP32 chip, which combines a **dual-core processor** running at up to 240 MHz, with **Wi-Fi** (802.11 b/g/n) and **Bluetooth** (classic and BLE) capabilities. These features allow developers to easily build wireless communication systems for a wide range of applications. Whether you’re creating smart home devices, building an IoT sensor network, or designing a robot, the ESP32 DevKit V1 provides the tools and connectivity needed to make your project a reality.



**Key Features of the ESP32 DevKit V1:**

1. **Dual-Core Processing**: The ESP32 features two CPU cores that run at speeds up to 240 MHz, enabling the efficient execution of complex tasks, especially in real-time applications.
2. **Wireless Connectivity**: Built-in **Wi-Fi** and **Bluetooth** (both classic and low energy) make it easy to integrate your device into a network or pair it with other Bluetooth-enabled devices. This flexibility is key for applications that require both local and remote communication.
3. **GPIO Pins and Peripherals**: The DevKit V1 provides a wide array of **GPIO pins** (up to 34 pins), which can be configured as inputs or outputs to interface with sensors, actuators, and other hardware. It also supports several communication protocols like **I2C**, **SPI**, **UART**, and **PWM**, allowing for easy integration with various external devices.
4. **Analog and Digital I/O**: With **analog-to-digital converters (ADC)** and **digital-to-analog converters (DAC)**, the ESP32 DevKit V1 allows developers to read analog signals (such as from sensors) and output analog signals to control devices like motors or speakers.
5. **Low Power Consumption**: The ESP32 is designed to operate efficiently with low power consumption, making it an excellent choice for battery-powered devices or systems where energy efficiency is critical.
6. **Integrated Flash Storage**: The board typically comes with **4MB of flash memory**, which is used to store your application code, data, and any necessary libraries. The onboard memory allows for complex projects and applications without requiring external storage.

**Development Environments and Flexibility:**

The ESP32 DevKit V1 is compatible with a variety of development environments, making it highly versatile for both beginners and professionals. Some of the most commonly used development platforms include:

* **Arduino IDE**: The ESP32 DevKit V1 can be programmed using the popular Arduino IDE, making it an excellent choice for those already familiar with Arduino. Libraries and examples are readily available to simplify development.
* **Espressif IDF**: For more advanced users, Espressif’s own **IoT Development Framework (IDF)** provides greater control over the hardware and is ideal for building more complex applications.
* **PlatformIO**: A cross-platform IDE that supports the ESP32, providing features like code completion, debugging, and easy integration with version control systems.

**Applications of the ESP32 DevKit V1:**

The ESP32 DevKit V1 is designed to be used in a wide range of applications. Some common use cases include:

* **Smart Home Devices**: From controlling lights and appliances to creating advanced security systems, the ESP32 enables wireless control and monitoring of devices in a home automation system.
* **IoT Sensors and Networks**: The board is perfect for building IoT sensors that collect data (such as temperature, humidity, or motion) and send it wirelessly to a server or cloud platform for analysis and storage.
* **Wearable Devices**: With its Bluetooth capabilities, the ESP32 is commonly used in wearable devices like fitness trackers, heart rate monitors, and smartwatches.
* **Robotics**: The ESP32 can be used to control robots, providing communication for remote control, autonomous operation, or both.

**Getting Started with the ESP32 DevKit V1:**

Getting started with the ESP32 DevKit V1 is easy. The board is supported by a large community of developers and has extensive documentation, tutorials, and example projects available online. These resources make it straightforward for beginners to start building projects without requiring deep expertise in embedded systems or electronics.

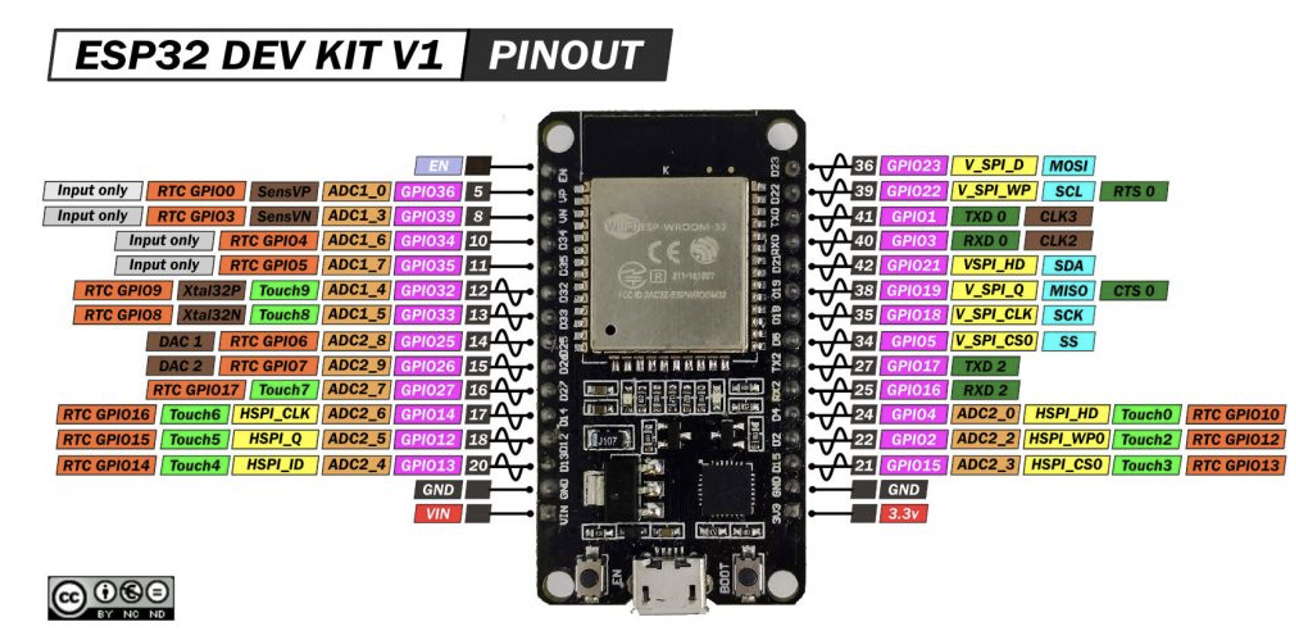
The Arduino IDE offers a simple setup process, where users can download and install the ESP32 board support package, select the correct board, and begin coding. With various libraries available for sensors, displays, and wireless communication, the development process becomes even more streamlined.

ESP32 Peripherals and I/O

Although the ESP32 has 48 GPIO pins in total, only 25 of them are broken out to the pin headers on both sides of the development board. These pins can be assigned a variety of peripheral duties, including:

|  |  |
| --- | --- |
| 15 ADC channels | 15 channels of 12-bit SAR ADC with selectable ranges of 0-1V, 0-1.4V, 0-2V, or 0-4V |
| 2 UART interfaces | 2 UART interfaces with flow control and IrDA support |
| 25 PWM outputs | 25 PWM pins to control things like motor speed or LED brightness |
| 2 DAC channels | Two 8-bit DACs to generate true analog voltages |
| SPI, I2C and I2S interface | Three SPI and one I2C interfaces for connecting various sensors and peripherals, as well as two I2S interfaces for adding sound to your project |
| 9 Touch Pads | 9 GPIOs with capacitive touch sensing |

The ESP32 DevKit V1 development board has 30 pins in total. For convenience, pins with similar functionality are grouped together. The pinout is as follows:



Pinout description

The DoIt ESP32 DevKit V1 features a comprehensive pinout that caters to a wide range of functionalities. Here’s a brief overview of its pinout:

1. **GPIO Pins:** The board provides numerous General Purpose Input/Output (GPIO) pins which can be used for various digital input/output functionalities. These pins also support functionalities like PWM, I2C, SPI, and more.
2. **Analog Inputs:** Several pins on the ESP32 DevKit V1 are capable of reading analog signals, making them suitable for interfacing with analog sensors.
3. **3.3V and GND Pins:** These are used to power external components or sensors.
4. **5V and GND:** The board can also provide a 5V output, which is useful for powering external modules that require more power.
5. **VIN:** This is the input voltage pin, which can be used to power the board when not using the USB connection.
6. **EN:** This is the enable pin. It’s used to reset the microcontroller.
7. **TX/RX:** These pins are used for serial communication.
8. **SPI Interface:** The board has pins for SPI communication, enabling fast data transfer with peripherals like displays or flash memory.
9. **I2C Interface:** The ESP32 DevKit V1 supports I2C communication, which is widely used for interfacing with sensors and other peripherals.
10. **Touch Sensor Pins:** Some GPIOs can be used as capacitive touch inputs, offering an interface for touch-based input devices.
11. **VP/VN:** These are the pins for the internal hall effect sensor.
12. **USB-to-UART Bridge:** This feature is crucial for programming the ESP32 using a USB cable and also for serial communication with a computer or other USB host devices.

Specifications

* Processors:
  + CPU: Xtensa dual-core (or single-core) 32-bit LX6 microprocessor, operating at 160 or 240 MHz and performing at up to 600 [DMIPS](https://en.wikipedia.org/wiki/Dhrystone)
  + Ultra low power (ULP) co-processor
* Memory: 520 KiB SRAM
* Wireless connectivity:
  + Wi-Fi: [802.11](https://en.wikipedia.org/wiki/IEEE_802.11) b/g/n
  + Bluetooth: v4.2 BR/EDR and BLE (shares the radio with Wi-Fi)
* Peripheral interfaces:
  + 12-bit [SAR ADC](https://en.wikipedia.org/wiki/Successive_approximation_ADC) up to 18 channels
  + 2 × 8-bit [DACs](https://en.wikipedia.org/wiki/Digital-to-analog_converter)
  + 10 × touch sensors ([capacitive sensing](https://en.wikipedia.org/wiki/Capacitive_sensing) GPIOs)
  + 4 × [SPI](https://en.wikipedia.org/wiki/Serial_Peripheral_Interface_Bus)
  + 2 × [I²S](https://en.wikipedia.org/wiki/I%C2%B2S) interfaces
  + 2 × [I²C](https://en.wikipedia.org/wiki/I%C2%B2C) interfaces
  + 3 × [UART](https://en.wikipedia.org/wiki/Universal_asynchronous_receiver/transmitter)
  + [SD](https://en.wikipedia.org/wiki/Secure_Digital)/[SDIO](https://en.wikipedia.org/wiki/Secure_Digital#SDIO_cards)/[CE-ATA](https://en.wikipedia.org/wiki/CE-ATA)/[MMC](https://en.wikipedia.org/wiki/MultiMediaCard)/[eMMC](https://en.wikipedia.org/wiki/MultiMediaCard#eMMC) host controller
  + SDIO/SPI slave controller
  + [Ethernet](https://en.wikipedia.org/wiki/Ethernet) MAC interface with dedicated DMA and [IEEE 1588 Precision Time Protocol](https://en.wikipedia.org/wiki/Precision_Time_Protocol) support
  + [CAN bus](https://en.wikipedia.org/wiki/CAN_bus) 2.0
  + Infrared remote controller (TX/RX, up to 8 channels)
  + Motor [PWM](https://en.wikipedia.org/wiki/Pulse-width_modulation)
  + LED [PWM](https://en.wikipedia.org/wiki/Pulse-width_modulation) (up to 16 channels)
  + [Hall effect sensor](https://en.wikipedia.org/wiki/Hall_effect_sensor)
  + Ultra low power analog pre-amplifier
* Security:
  + IEEE 802.11 standard security features all supported, including WFA, WPA/WPA2 and [WAPI](https://en.wikipedia.org/wiki/WLAN_Authentication_and_Privacy_Infrastructure)
  + Secure boot
  + Flash encryption
  + 1024-bit OTP, up to 768-bit for customers
  + Cryptographic hardware acceleration: [AES](https://en.wikipedia.org/wiki/Advanced_Encryption_Standard), [SHA-2](https://en.wikipedia.org/wiki/SHA-2), [RSA](https://en.wikipedia.org/wiki/RSA_(cryptosystem)), [elliptic curve cryptography](https://en.wikipedia.org/wiki/Elliptic_curve_cryptography) (ECC), [random number generator](https://en.wikipedia.org/wiki/Random_number_generator) (RNG)
* Power management:
  + Internal [low-dropout regulator](https://en.wikipedia.org/wiki/Low-dropout_regulator)
  + Individual power domain for RTC
  + 5μA deep sleep current
  + Wake up from GPIO interrupt, timer, ADC measurements, capacitive touch sensor interrupt