***THE ESP32***

This microcontroller was developed due to the increasing need for industrial and commercial IoT as compared to the small scale use of the ESP8266. It was developed from around 2015 to address the shortcomings of the mother microcontroller and act as its powerful successor. The first dev. kit was released into the market towards the end of 2016. The chip gained rapid popularity in professional and hobbyist markets. This chip could:

* Handle more complex applications, e.g. audio streaming and real time control.
* Add secure communication for production IoT devices.
* Support Bluetooth that could be used for local wireless communication
* Improve multitasking and peripheral support.

This among many other functions enabled edge computing and application of the microcontroller to security sensitive applications.

***ESP32 DEVELOPMENT BOARDS***

A development board is a complete ready to use board built for prototyping and development. It normally includes:

1. An ESP32 module
2. A USB interface i.e. a USB to serial chip
3. Voltage Regulators that power the board via USB or an external source
4. Buttons for booting and resetting.
5. Pin headers that allow for easy connection to breadboards or peripherals.

Examples include:

1. **ESP32 DevKit V1**

It is a general purpose ESP32 development board. It is the most popular kit and best for beginners. It uses the ESP32-WROOM-32 module, has USB to Serial chip and full GPIO access. It is simple, well-supported, breadboard-friendly, is great for learning and prototyping. It however has no built in display or cameras.



Figure 1: ESP32 DevKit V1

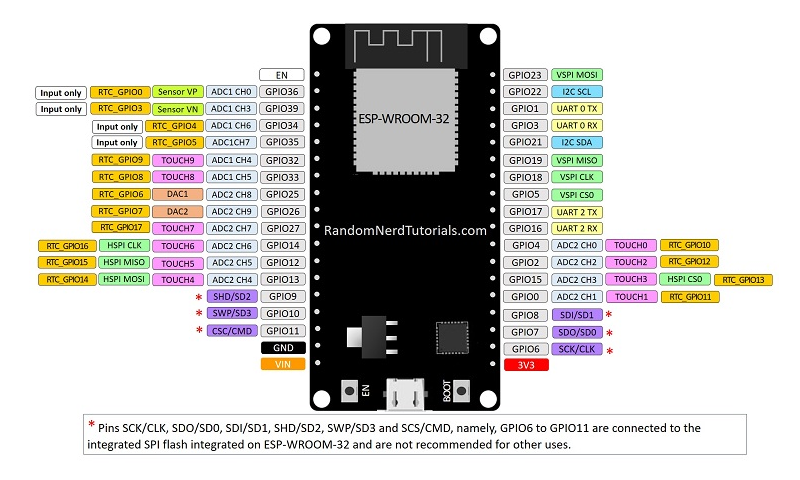


Figure 2: Pin out diagram for ESP32 DEVKIT V1 DOIT board

1. **ESP32 OLED Kit**

It includes a built in 0.96” OLED display making it a good addition for IoT applications. It has provision for connection an external antennae and a battery connector for creating battery operated projects. It uses the ESP32-WROOM-32 or WROVER module. The built in display saves GPIOs and space and is good for live feedback. It however has fewer accessible GPIOs

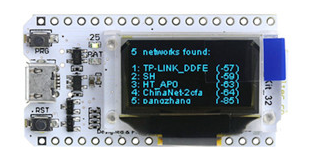


Figure 3:ESP32 OLED Kit

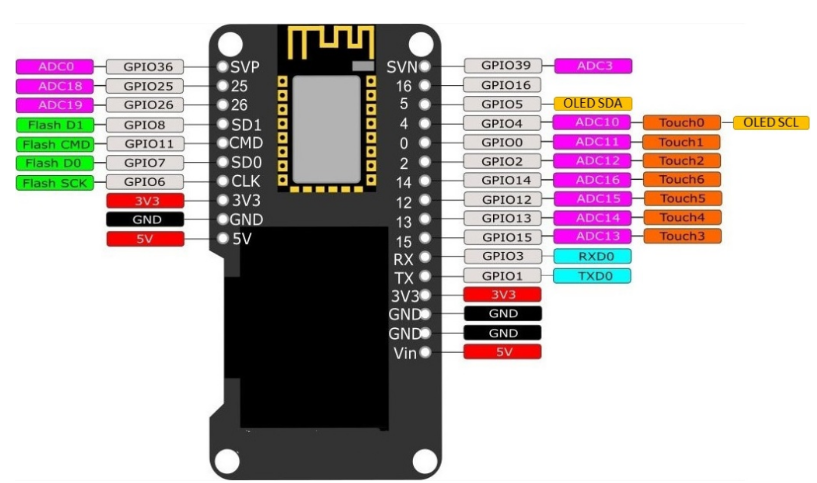


Figure 4: Pin out diagram for ESP32 OLED Kit

1. **ESP32-CAM**

It has a built in 2MP camera and a microSD card slot. It is suitable for projects that require a camera with advanced functions like image tracking and recognition. I t has no USB port and therefore requires external USB to Serial adapter. It uses ESP32-S a variant of the ESP32. It however has less GPOs due to the camera pins.



Figure 5:ESP32-CAM

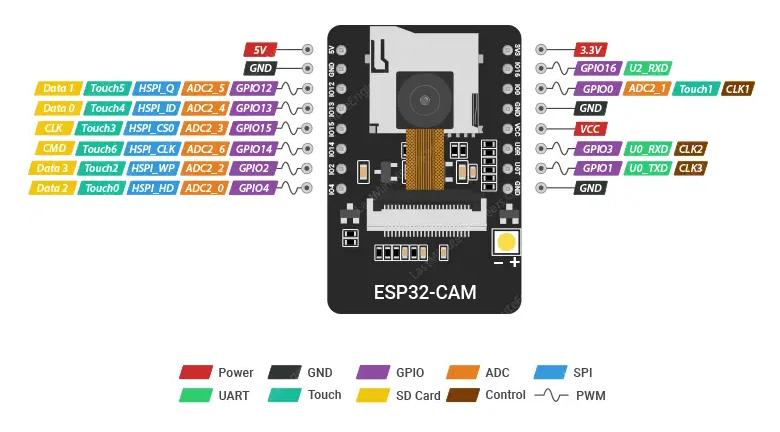


Figure 6: Pin out diagram for ESP32 CAM

1. **ESP32 SIM800L TTGO T-Call**

It allows one to connect to the internet via a SIM card data plan without the need of Wi-Fi. It has a GSM module (SIM800L) to allow cellular connectivity. It supports SMS, GPRS and voice. It is ideal for remote IoT. It can run fully off grid with battery. It however consumes more power and requires good battery management.

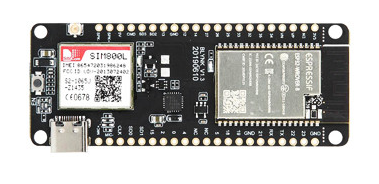


Figure 7:ESP32 SIM800L TTGO T-Call

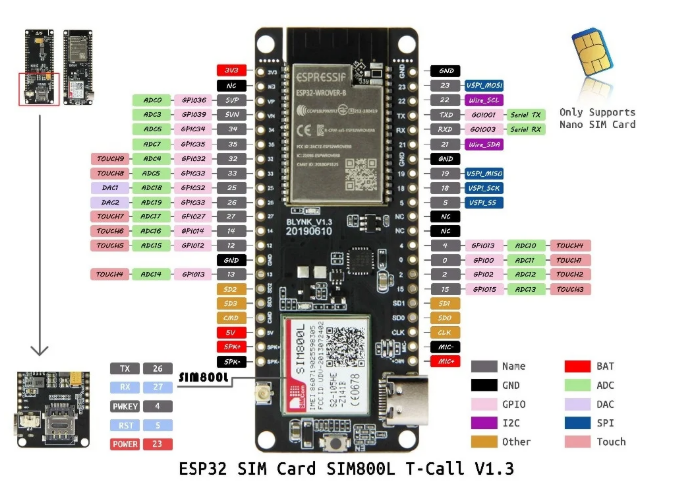


Figure 8: Pin out Diagram for ESP32 SIM800L TTGO T-Call

1. **M5Stack**

This is a stackable modular product development toolkit based on ESP32.it includes a main controller and additional modules that can be stacked to expand the functionality of the project. It is basically a collection of modular stackable ESP32-based mini-computers and has multiple versions. It has a case, screen and battery support. It is hard to customize it at hardware level.



Figure 9:M5Stack

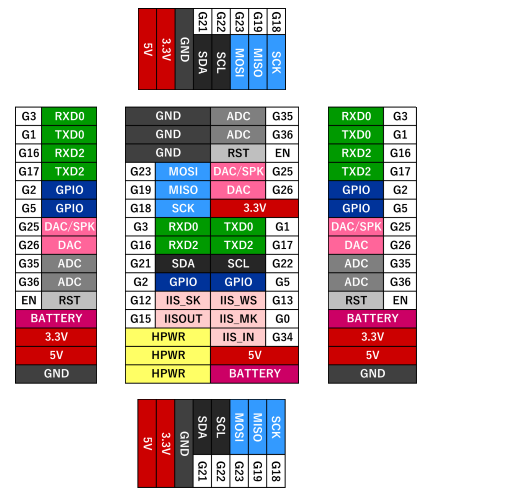


Figure 10: Pin out Diagram for M5Stack

***ESP32 DEVELOPMENT BOARDS PINOUT DIAGRAM***

Due to its popularity, availability and wide range of applications, the discussions below will specifically be tailored to the ESP32 DevKit V1. However, as has been highlighted earlier on, the functionalities of these pins doesn’t change with the development board.

The ESP32 comes with 48 pins but only 25 are broken out to pin headers. Not every development board exposes all the pins that shall be discussed herein. It is also worth noting that the pin location and accessible can change depending on the manufacturer and the development board. The pins, however, work exactly the same regardless of the development board in question. The ESP32 has a pin multiplexing feature. This enables multiple peripherals to share a single GPIO pin. Generally, there are 40 identifiers for GPIO pins including the reserved and internal pins that may not be accessible for general use. The exact number of usable GPIOs is 34 (GPIO0- GPIO39). Of these 34, 6 are strictly for input (GPIO34- GPIO39). The remaining 6 that are not accounted of from the 40 (GPIO6-GPIO11) are tied to internal flash and are not ideal for normal GPIO usage.

In addition to those pins, there are

* Ground pins (GND).
* Power pins - 3V3 and VIN
* Enable pin (EN) - used to enable the ESP32. When HIGH, chip is enabled and when LOW chip operates at low power. The EN pin is also connected to a pushbutton switch that can pull the pin LOW and trigger a reset.

Below is table that shows what pins are best to use as inputs, outputs and which need caution while in use.

*Key:*

Table 1: Key for Table 2 Color Codes

|  |  |
| --- | --- |
| **Color** | **Meaning** |
| Green | Pins are OK to use |
| Yellow | Pins are OK to use but attention needs to be paid due to their unexpected behavior especially at boot. |
| Red | Pins not recommended to be used as inputs or outputs. |

Table 2: Table to guide on the use of different GPIO pins

|  |  |  |  |
| --- | --- | --- | --- |
| **GPIO** | **Input** | **Output** | **Notes** |
| **0** | pulled up | OK | outputs PWM signal at boot, must be LOW to enter flashing mode and during programming. Must be HIGH during boot |
| **1** | TX pin | OK | debug output at boot. Used for flashing and debugging |
| **2** | OK | OK | connected to on-board LED, must be left floating or LOW to enter flashing mode and also low during boot. Is also connected to the on-board LED |
| **3** | OK | RX pin | HIGH at boot. Used for flashing and debugging |
| **4** | OK | OK |  |
| **5** | OK | OK | outputs PWM signal at boot, strapping pin |
| **6** | x | x | connected to the integrated SPI flash |
| **7** | x | x | connected to the integrated SPI flash |
| **8** | x | x | connected to the integrated SPI flash |
| **9** | x | x | connected to the integrated SPI flash |
| **10** | x | x | connected to the integrated SPI flash |
| **11** | x | x | connected to the integrated SPI flash |
| **12** | OK | OK | boot fails if pulled high, strapping pin, must be LOW during boot |
| **13** | OK | OK |  |
| **14** | OK | OK | outputs PWM signal at boot |
| **15** | OK | OK | outputs PWM signal at boot, strapping pin, must be HIGH during boot |
| **16** | OK | OK |  |
| **17** | OK | OK |  |
| **18** | OK | OK |  |
| **19** | OK | OK |  |
| **21** | OK | OK |  |
| **22** | OK | OK |  |
| **23** | OK | OK |  |
| **25** | OK | OK |  |
| **26** | OK | OK |  |
| **27** | OK | OK |  |
| **32** | OK | OK |  |
| **33** | OK | OK |  |
| **34** | OK |  | input only |
| **35** | OK |  | input only |
| **36** | OK |  | input only |
| **39** | OK |  | input only |

***FEATURES OF THE ESP32.***

1. ***Processor and Architecture***

It is a 32 bit Tensilica Xtensa LX6 microcontroller with a dual core. It has a clock speed of up to 240MHz. It has the ability to allow the different cores to run at independent speeds or even be powered down. It has an Ultra-Low Power (ULP) 8 bit Coprocessor to allow sensor monitoring during deep sleep. It can execute up to 600 DMIPS for both cores.

1. ***Internal Memory***

It has 448KB of ROM for booting and core functions. It has an internal SRAM of 520 KB, shared between data RAM, instruction RAM and Cache.

It has an RTC memory that is a small and special low power memory region. It belongs to the RTC domain, a subsystem that remains powered during deep sleep. It can retain data while the main CPU is powered off. It used for low power tasks like waking up from sleep, preserving state or running ultra-low power co-processors.

It has 8 KB of SRAM in RTC, which is called RTC fast memory and can be used for data storage; it is accessed by the main CPU during RTC Boot from the light-sleep mode.

It has 8 KB of SRAM in RTC, which is called RTC slow memory and can be accessed by the co-processor during the deep-sleep mode.

1. ***External Flash and SRAM***

ESP32 can access the external QSPI flash and SRAM through high-speed caches.

* The external flash can be mapped into CPU instruction memory space and read-only memory space simultaneously.
* When external flash is mapped into CPU instruction memory space, up to 11 MB + 248 KB can be mapped at a time. Note that if more than 3 MB + 248 KB are mapped, cache performance will be reduced due to speculative reads by the CPU.
* When external flash is mapped into read-only data memory space, up to 4 MB can be mapped at a time. 8-bit, 16-bit and 32-bit reads are supported.
* External SRAM can be mapped into CPU data memory space. Up to 4 MB can be mapped at a time. 8-bit, 16-bit and 32-bit reads and writes are supported.

ESP32-WROOM-32 integrates a 4 MB SPI flash, which is connected to GPIO6, GPIO7, GPIO8, GPIO9, GPIO10 and GPIO11. These six pins cannot be used as regular GPIOs.

Some modules like the ESP32-WROVER come with a PSRAM that can be quite useful for graphical applications that take up a lot of storage.

1. ***Wireless Connectivity***

It has Wi-Fi capabilities (IEEE 802.11) with 2.4 GHz band. It has a Wi-Fi transceiver allowing it to connect to a Wi-Fi network to access the internet (Station Mode). It can also create its own Wi-Fi wireless network (Soft Access Point mode) to which other devices can connect.

It supports classic Bluetooth (BT) and Bluetooth Low Energy (BLE 4.2/ 5.0). The BLE supports multiple advertising, scanning and GATT services. Both the Wi-Fi and BLE can be used simultaneously. The Bluetooth is mostly applicable for remote controlling purposes. The ESP32 supports

1. ***Peripherals and I/O***

* ***GPIOs****-* it has up to 25 GPIO pins that can be assigned different functions by programming the appropriate registers. This is because GPIOs 0, 37 and 38 are not available on most modules (only bare chips) due to hardware constraints. There are of several kinds: digital only, analog-enabled, capacitive touch enabled etc. Analog- enabled GPIOs and Capacitive touch enabled GPIOs can be configured as digital GPIOs. Most of the digital GPIOs can be configured with internal pull-up or pull-down resistors or set to high impedance.
* ***Input only GPIOs-*** Pins GPIO34, GPIO35, GPIO36 and GPIO39 cannot be configured as outputs. They can be used as digital or analog inputs, or for other purposes. They also lack internal pull-up and pull-down resistors, unlike the other GPIO pins.
* ***ADC channels*** *-* is available on 18 channels/pins. There are 2 ADC units (physical ADC core in the chip), ADC1 and ADC2. ADC1 handles 8 channels (GPIOs 32-39) while ADC2 handles 10 channels (GPIOs 0, 2, 4, 12-15, 25-27). However, due to hardware constraints, not all of them are usable leaving room for only 15 ADC channels.

Each unit can read from multiple GPIOs in general. However, the unit can only read from one channel at a time. ADC2 shares internal resources with the Wi-Fi radio and is therefore not suited for simultaneous use with Wi-Fi.

The ADC input channels have a 12 bit resolution meaning it can detect 4096 discrete analog levels. Analog readings 0-4095 correspond to voltages 0 - 3.3V. This results in a resolution of 0.8mV per unit (3.3/4096). The ADC pins do not have a linear behavior and hence some digital values that are closer to each other will be hardly distinguishable. The ESP32 is designed to measure voltages while in sleep mode.

* ***DAC****-* it has 2 8 bit channels that are used to convert digital signals to true analog voltage signal outputs. It can be used as a ‘digital potentiometer’ to control analog devices. These DACs have an 8-bit resolution, which means that values ranging from 0 to 256 will be converted to an analog voltage ranging from 0 to 3.3V.

The DACs 8 bit resolution may be insufficient for use in audio applications in which case an external DAC with a higher resolution is preferred.

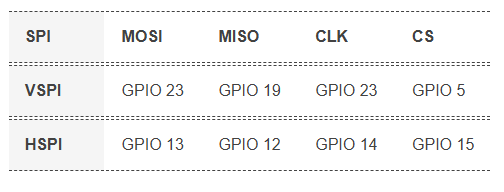
* ***Touch Sensors****-* it has 10 internal capacitive touch sensors that can be used for functions such as wakeup, UI or low power touch activation. They use capacitive touch technology. They can sense variations in anything that holds an electrical charge like the human skin. These pins can be integrated into capacitive touch and replace mechanical buttons. These internal touch sensors are connected to the following GPIOs (GPIO 0, 2, 4, 12-15, 27, 32, 33). Due to hardware constraints, only 9 are usable.
* ***SPI* -**SPI pins support high-speed communication with peripherals, making them indispensable in data-intensive applications.

The SPI uses 4 main signals:

* **MOSI-** Master Out Slave In
* **MISO-** Master In Slave Out
* **SCLK**- Serial Clock
* **CS-** Chip Select (active low)

The ESP32 has four SPI peripheral devices, called SPI0, SPI1, HSPI and VSPI. By default, the pin mapping for SPI is:

Table 3: Pin Mapping for SPI

****

GPIO6 to GPIO9, GPIO10, and GPIO11 are exposed in some ESP32 development boards. However, these pins are connected to the integrated SPI flash on the ESP-WROOM-32 chip and are not recommended for other uses.

* ***Inter-Integrated Circuit (12C)*** *-* has 2 I2C bus interfaces but no dedicated I2C pins. Instead, it allows for flexible pin assignment, meaning any GPIO pin can be configured as I2C SDA (data line) and SCL (clock line).

However, GPIO21 (SDA) and GPIO22 (SCL) are commonly used as the default I2C pins especially when using the Arduino IDE.

* ***UART interfaces***- has 3 hardware supported UARTs that support asynchronous communication at up to 5Mbps. The ESP32 supports up to three UART interfaces: UART0, UART1, and UART2, depending on the ESP32 board model you’re using.
* UART0 is usually reserved for communication with the serial monitor during upload and debugging. However, you can also use it for communication with other devices after uploading the code if the Serial Monitor is not needed.

UART0 pins are connected to the USB-to-Serial converter and are used for flashing and debugging. Therefore, the UART0 pins are not recommended for use.

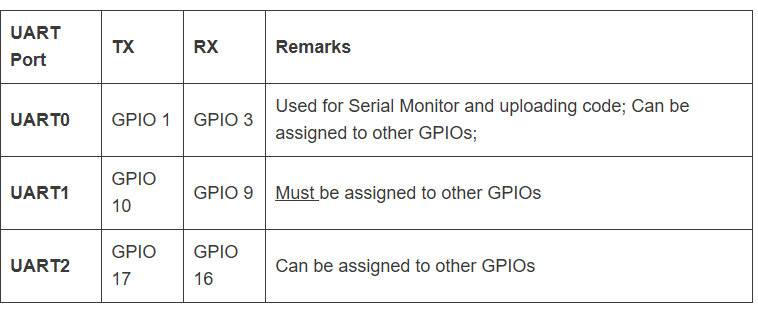
* UART1 and UART2: available to communicate with external devices.

Like I2C and SPI, these UART pins can be mapped to any GPIO pin on the ESP32. However, they have a default pin assignment on most board models.

* UART2 is a safe option for connecting to UART-devices such as GPS, fingerprint sensor, distance sensor, and so on.

For most ESP32 boards the UART pin assignment is as follows:

Table 4: UART pin assignment



About UART1 (GPIO 9 and GPIO10) – these GPIOs are connected to the ESP32 SPI flash memory, so you can’t use them like that. To use UART1 to communicate with other devices, you must define different pins.

* ***Integrated Inter-chip Sound (12S)* -** on 2 pins and is used as an audio interface for microphones and DACs.
* ***PWM****-* it can be assigned to any of the 25 GPIOs but only 16 can be active simultaneously. This is because the ESP32 has 16 channels capable of outputting PWM. Each channel is controlled by the LED PWM controller (LEDC). The PWM controller consists of PWM timers, the PWM operator and a dedicated capture sub-module. Each timer provides timing in synchronous or independent form, and each PWM operator generates a waveform for one PWM channel. The dedicated capture sub-module can accurately capture events with external timing.

To set a PWM signal, you need to define these parameters in the code:

* Signal’s frequency;
* Duty cycle;
* PWM channel;
* GPIO where you want to output the signal.
* ***Strapping Pins –*** the ESP32 has 5 strapping pins:
* GPIO 0 (must be LOW to enter boot mode)
* GPIO 2 (must be floating or LOW during boot)
* GPIO 4
* GPIO 5 (must be HIGH during boot)
* GPIO 12 (must be LOW during boot)
* GPIO 15 (must be HIGH during boot)

These pins are used to put the ESP32 into BOOT mode (to run the program stored in the flash memory) or FLASH mode (to upload the program to the flash memory). Depending on the state of these pins, the ESP32 will enter BOOT mode or FLASH mode at power on.

On most development boards with built-in USB/Serial, you don’t need to worry about the state of these pins, as the board puts them in the correct state for flashing or boot mode.

However, if peripherals are connected to these pins, you may encounter issues when attempting to upload new code or flash the ESP32 with new firmware, as these peripherals prevent the ESP32 from entering the correct mode.

The strapping pins function normally after reset release, but they should still be used with caution.

* ***CAN Bus****-* it is supported on some modules. It is designed for robust and flexible performance in harsh environments and is useful for industrial applications.
* ***Ethernet MAC****-* has been integrated but requires an external PHY chip to enable its use. This physical layer chip will enable physical layer functions and is suitable for very short distance transmission.
  + ***Hall Effect Sensor****-* the ESP32 has a built-in Hall Effect sensor located beneath the metal lid of the ESP32-WROOM-32 module itself.It is used to measure magnetic fields without need to connect external modules. Being integrated into the ESP32 means that you can easily connect the sensor readings with Wi-Fi or Bluetooth functionalities, making remote monitoring and control easier.

While the onboard Hall sensor might not replace dedicated external sensors for precise applications due to its positioning and sensitivity, it still offers a range of potential uses. This includes basic magnetic field detection, triggering specific functions when a magnet is nearby, or even building simple educational projects to understand the Hall Effect.

Because the sensor is located beneath the metal lid, it is less sensitive to weak magnetic fields than standalone Hall sensors, so magnets of significant strength are usually required to obtain noticeable readings.

***NB:*** *The ADC and DAC features are assigned to specific static pins. One can however decide which pins are UART, 12C, SPI or PWM through assigning them in the code. This is made possible by the multiplexing ability of the chip. Despite the fact that one can define pin properties on the software, one has to be careful because there are pins assigned as default.*

1. ***Power Management and Sleep Modes***

The module can safely and efficiently operate within a voltage range of 2.2 -3.6V. The board therefore has a voltage regulator to keep the voltage stable at 3.3.V and can reliably provide up to 600mA of current. The output of the 3.3V regulator is broken out to the header pin labelled 3V3 which can then be used to power external circuitry. This board is typically powered by the on-board MICROB USB connector. If one would rather use a regulated 5V power supply, this can be connected through the Vin pin to directly power the ESP32 and its peripherals.

The ESP32 sleep mode is a power-saving mode. When not in use, the ESP32 can enter this mode, storing all data in RAM. At this point, all unnecessary peripherals are disabled while the RAM receives enough power to retain its data.

It takes up a current of between 5 µA and 240mA depending on the power mode it is operating in at a given moment. These include:

* **Active/Normal mode-** In this mode, all peripherals of the chip remain active.Since everything is always active in this mode (especially the WiFi module, processing core, and Bluetooth module), the chip consumes about 240 mA of power. It has also been observed that the chip draws more than 790 mA at times, particularly when both WiFi and Bluetooth are used simultaneously.
* **Modem Sleep mode** - In modem sleep mode, everything is active except for the WiFi, Bluetooth, and the radio. The CPU remains active, and the clock is configurable. In this mode, the chip consumes approximately 3 mA at slow speed and 20 mA at high speed. To keep the connection alive, Wi-Fi, Bluetooth, and the radio are woken up at predefined intervals. This is referred to as the *Association Sleep Pattern.* During this sleep pattern, ESP32 switches between active mode and modem sleep mode. To accomplish this, the ESP32 connects to the router in station mode using the DTIM beacon mechanism. The Wi-Fi module is disabled between two DTIM beacon intervals and then automatically enabled just before the next beacon arrives. This results in power conservation. The sleeping time is determined by the router’s DTIM beacon interval time, which is typically 100 ms to 1000 ms.

***NB:*** DTIM stands for Delivery Traffic Indication Message. In this mechanism, the access point (AP)/router broadcasts beacon frames periodically. Each frame contains network-related information. It is used to announce the presence of a wireless network as well as to synchronize all connected members.

* **Light Sleep mode** - Light sleep is similar to modem sleep in that the chip follows the Association Sleep Pattern. The only difference is that in light sleep mode, the CPU, most of the RAM, and digital peripherals are clock-gated. During light sleep mode, the CPU is paused by disabling its clock pulse. The RTC and ULP-coprocessor, on the other hand, remain active. This results in a lower power consumption than the modem sleep mode, which is around 0.8 mA. Before entering light sleep mode, the ESP32 stores its internal state in RAM and resumes operation upon waking from sleep. This is referred to as Full RAM Retention.

***NB:*** Clock gating is a popular power management technique for reducing dynamic power dissipation by removing or ignoring the clock signal when the circuit is not in use.

Clock gating reduces power consumption by pruning the clock tree. Pruning the clock disables portions of the circuitry, preventing the flip-flops in them from switching states. Since switching states consumes power, when not switched, the power consumption drops to zero.

* **Deep Sleep mode** - In deep sleep mode, the CPUs, most of the RAM, and all digital peripherals are disabled. Only the following parts of the chip remain operational:
  + ULP Coprocessor
  + RTC Controller
  + RTC Peripherals
  + RTC fast and slow memory

In deep sleep mode, the chip consumes anywhere between 0.15 mA (when the ULP coprocessor is on) and 10 µA.

During deep sleep mode, the primary CPU is turned off, whereas the Ultra-Low-Power (ULP) Coprocessor can take sensor readings and wake up the CPU as needed. This sleep pattern is referred to as the ULP sensor-monitored pattern. This is useful for designing applications where the CPU needs to be woken up by an external event, a timer, or a combination of these events, while maintaining minimal power consumption.

Along with the CPU, the main memory of the chip is also disabled. As a result, everything stored in that memory is erased and cannot be accessed.

Because RTC memory is kept active, its contents are preserved even during deep sleep and can be retrieved once the chip is woken up. This is why the chip stores Wi-Fi and Bluetooth connection data in RTC memory before entering deep sleep. When the chip wakes up from deep sleep, it performs a reset and begins program execution from the beginning.

* **Hibernation mode** - Hibernate mode is very similar to deep sleep. The only difference is that in hibernation mode, the chip disables the internal 8 MHz oscillator as well as the ULP-coprocessor, leaving only one RTC timer (on slow clock) and a few RTC GPIOs to wake the chip up. Because the RTC recovery memory is also turned off, we cannot save any data while in hibernation mode. As a result, the chip’s power consumption is reduced even further; in hibernation mode, it consumes only about 2.5 μA. This mode is especially useful if you’re working on a project that doesn’t need to be active all the time.

1. ***Security Features***

Its security features are upped from the esp8266 microcontroller. It has hardware acceleration for encryption. It also has secure boot that prevents unauthorized firmware from running. Flash encryption protects contents of external flash. It supports WPA/WPA2/Enterprise encryption.

1. ***Development Environment***

It can support multiple programming languages like C, C++, MicroPython, JavaScript etc. It can also be used in development environments like the ESP-IDF and  Arduino IDE which happen to be the most popular. It can also support MQTT, HTTP and Over the Air (OTA) updates.

The ESP32 has multiple variants that are improved versions of the original. Each of these variants has multiple modules based on the particular chip. These variants include:

1. ***ESP32 Series-*** This is the original and most common variant.

* ***Main Features.***

*32-bit MCU & 2.4 GHz Wi-Fi & Bluetooth/Bluetooth Les*

* 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
* Sleep current is less than 5 μA, 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

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **MODULE** | **DESCRIPTION** | **CHIP EMBEDDED** | **DIMENSIONS** | **PINS** | **FLASH (MB)** | **PSRAM (MB)** | **ANETNNAE** | **DEVELOPMNT BOARD** |
| **ESP32-WROOM-32E** | ESP32-WROOM-32E integrates ESP32-D0WD-V3, with higher stability and safety performance. | ESP32-D0WD-V3  ESP32-D0WDR2-V3 | 18x25.5x3.1 | 38 | 4,8,16 | N/A | PCB antenna | ESP32-DevKitC |
| **ESP32-WROOM-32UE** | ESP32-WROOM-32UE integrates ESP32-D0WD-V3, with higher stability and safety performance. | ESP32-D0WD-V3  ESP32-D0WDR2-V3 | 18x19.2x3.2 | 38 | 4,8,16 | N/A | IPEX antenna connector | ESP32-DevKitC |
| **ESP32-WROVER-E** | ESP32-WROVER-E integrates ESP32-D0WD-V3, with higher stability and safety performance. | ESP32-D0WD-V3  ESP32-D0WDR2-V3 | 18x31.4x3.3 | 38 | 4,8,16 | 8 | PCB antenna | ESP32-DevKitC  ESP-WROVER-KIT  ESP32-LyraT  ESP32-LyraT-Mini  ESP32-LyraTD-MSC  ESP32-LyraTD-SYNA  ESP32-Vaquita-DSPG  ESP32-Korvo  ESP32-Ethernet-Kit |
| **ESP32-WROVER-IE** | ESP32-WROVER-IE integrates ESP32-D0WD-V3, with higher stability and safety performance | ESP32-D0WD-V3  ESP32-D0WDR2-V3 | 18x31.4x3.3 | 38 | 4,8,16 | 8 | IPEX antenna connector | ESP32-DevKitC |
| **ESP32-MINI-1** | ESP32-MINI-1 is a highly-integrated, small-sized Wi-Fi+ Bluetooth +Bluetooth® LE MCU module that has a rich set of peripherals | ESP32-U4WDH | 13.2×19×2.4 | 55 | 4 MB embedded in chip | N/A | PCB antenna | ESP32-DevKitM-1 |
| **ESP32-MINI-1U** | ESP32-MINI-1U is a highly-integrated, small-sized Wi-Fi + Bluetooth® + Bluetooth® LE MCU module that has a rich set of peripherals. | ESP32-U4WDH | 13.2×13.5× 2.4 | 55 | 4 MB embedded in chip | N/A | IPEX antenna connector | ESP32-DevKitM-1 |
| **ESP32-PICO-MINI-02** | ESP32-PICO-MINI-02 is a small-sized powerful Wi-Fi+ Bluetooth +Bluetooth® LE MCU module. It is based on ESP32-PICO-V3-02, A System – in –Package device which integrates an 8MB SPI flash, 2MB PSRAM and a 40MHz oscillator. | ESP32-PICO-V3-02 | 13.2×16.6×2.4 | 53 | 8 MB embedded in chip | 2 MB embedded in chip | PCB antenna | ESP32-PICO-DevKitM-2 |
| **ESP32-PICO-MINI-02U** | ESP32-PICO-MINI-02U is a small-sized powerful Wi-Fi+Bluetooth®+Bluetooth® LE MCU module. It is based on ESP32-PICO-V3-02, a System-in-Package (SiP) device, which integrates an 8 MB SPI flash, 2 MB SPI Pseudo static RAM (PSRAM) and 40 MHz oscillator.. | ESP32-PICO-V3-02 | 13.2x11.2x 2.4 | 53 | 8 MB embedded in chip | 2 MB embedded in chip | IPEX antenna connector | ESP32-PICO-DevKitM-2 |
| **ESP32-PICO-V3-ZERO** | ESP32-PICO-V3-ZERO is an Alexa Connect Kit (ACK) module with Espressif. ESP32-PICO-V3-ZERO is based on ESP32-PICO-V3, a System-in-Package (SiP) device. It provides complete Wi-Fi and Bluetooth® functionalities with ultra-small size, robust performance and low energy consumption. | ESP32-PICO-V3 | 16×23×2.3 | 77 | 4 MB embedded in chip | N/A | PCB antenna | ESP32-PICO-V3-ZERO-DevKit |
| **ESP32-WROOM-32D**  **ESP32-WROOM-32E** | ESP32-WROOM-32D integrates ESP32-D0WD. | ESP32-D0WD | 18x25.5x3.1 | 38 | 4,8,16 | N/A | PCB antenna | ESP32-DevKitC |
| **ESP32-WROOM-32U**  **ESP32-WROOM-32UE** | ESP32-WROOM-32U integrates ESP32-D0WD. It integrates a U.FL connector. | ESP32-D0WD | 18x19.2x3.2 | 38 | 4,8,16 | N/A | IPEX antenna connector | ESP32-DevKitC |
| **ESP32-SOLO-1**  **ESP32-WROOM-32E** | ESP32-SOLO-1 is a powerful, generic Wi-Fi+BT+Bluetooth LE MCU module that targets a wide variety of applications, ranging from low-power sensor networks to the most demanding tasks, such as voice encoding, music streaming and MP3 decoding. | ESP32-S0WD | 18x25.5x3.1 | 38 | 4 | N/A | PCB antenna | ESP32-DevKitC |
| **ESP32-WROOM-32**  **ESP32-WROOM-32E** | SP32-WROOM-32 contains the ESP32 SoC, flash memory, high-precision discrete components, and a PCB antenna which provides outstanding RF performance in space-constrained applications. | ESP32-D0WDQ6 | 18x25.5x3.1 | 38 | 4 | N/A | PCB antenna | ESP32-DevKitC |
| **ESP32-WROVER-B**  **ESP32-WROVER-E** | ESP32-WROVER-B is a powerful, generic WiFi-BT-Bluetooth LE MCU module that targets a wide variety of applications, ranging from low-power sensor networks to the most demanding tasks, such as voice encoding, music streaming and MP3 decoding. | ESP32-D0WD | 18x31.4x3.3 | 38 | 4,8,16 | 8 | PCB antenna | ESP32-DevKitC  ESP-WROVER-KIT  ESP32-LyraTD-DSPG |
| **ESP32-WROVER-IB**  **ESP32-WROVER-IE** | ESP32-WROVER-IB is a powerful, generic Wi-Fi+BT+Bluetooth LE MCU module that targets a wide variety of applications | ESP32-D0WD | 18x31.4x3.3 | 38 | 4,8,16 | 8 | IPEX antenna connector | ESP32-DevKitC |
| **ESP32­-WROOM­-DA** | ESP32-WROOM-DA is a powerful Wi-Fi + Bluetooth + Bluetooth LE MCU module, which has the same layout of pins as ESP32-WROOM-32E, facilitating quick and easy migration between these two modules. With two unique complementary PCB antennas in different directions on one single module, | ESP32-D0WD-V3 | 35.6x34.4x3.5 | 41 | 4,8,16 | N/A | Two complementary PCB antennas | ESP32-DevKitC |
| **ESP32-DU1906** | ESP32-DU1906 is an industry-leading AIoT voice module powered by Espressif's flagship chip ESP32-D0WD-V3 and Baidu's HongHu voice chip DU1906. This module integrates Wi-Fi, classic Bluetooth, Bluetooth LE, and voice processing circuits on board, providing first-class user experience for speech wake-up and interaction. | ESP32-D0WD-V3 | 22×42×3.5 | 66 | 8 | 8 | PCB antenna | ESP32-Korvo-DU1906 |
| **ESP32-DU1906-U** | ESP32-DU1906-U is an industry-leading AIoT voice module powered by Espressif's flagship chip ESP32-D0WD-V3 and Baidu's HongHu voice chip DU1906. This module integrates Wi-Fi, classic Bluetooth, Bluetooth LE, and voice processing circuits on board, providing first-class user experience for speech wake-up and interaction. | ESP32-D0WD-V3 | 22×35.5× 3.5 | 66 | 8 | 8 | IPEX antenna connector | ESP32-Korvo-DU1906 |

1. ***ESP32-H Series*** - Optimized for ultra-low power consumption and is designed for IoT applications with stringent energy requirements.

* ***Main Features.***

*32-bit RISC-V MCU & Bluetooth 5 (LE) & IEEE 802.15.4*

* 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

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **MODULE** | **DESCRIPTION** | **CHIP EMBEDDED** | **DIMENSIONS** | **PINS** | **FLASH** | **ANETNNAE** | **DEVELOPMNT BOARD** |
| **ESP32-H2-MINI-1** | ESP32-H2-MINI-1 is a powerful, generic Bluetooth Low Energy and IEEE 802.15.4 combo module with ESP32-H2 chip at its core. | ESP32-H2FH2S  ESP32-H2FH4S | 13.2×16.6×  2.4 | 53 | 1, 2, 4 | PCB antenna | ESP32-H2-DevKitM-1 |
| **ESP32-H2-MINI-1U** | ESP32-H2-MINI-1U is a powerful, generic Bluetooth Low Energy and IEEE 802.15.4 combo module with ESP32-H2 chip at its core. | ESP32-H2FH2S  ESP32-H2FH4S | 13.2×12.5×  2.4 | 53 | 2, 4 | IPEX antenna connector | ESP32-H2-DevKitM-1 |
| **ESP32-H2-WROOM-02C** | The ESP32-H2-WROOM-02C module is based on the ESP32-H2 chip, and can be mounted onto the surface of a PCB board. It has 19 GPIOs and the pin supports UART, LED PWM, 12C, SPI, I2S, and RMT. | ESP32-H2FH2S  ESP32-H2FH4S | 18.0 × 20.0 × 3.2 | 29 | 2, 4 | PCB antenna | ESP32-Module-Prog-1(R)  ESP-Module-Prog-SUB-02 |

1. ***ESP32-C Series*** - Emphasizes cost-effectiveness and integrates Wi-Fi and Bluetooth features, ideal for simple IoT devices

* ***Main Features- ESP32-C6 Series***

***32-bit RISC-V MCU & 2.4 GHz Wi-Fi 6 & Bluetooth 5 (LE) & IEEE 802.15.4***

* 32-bit RISC-V single-core processor that operates at up to 160 MHz
* State-of-the-art power and RF performance
* 320 KB ROM, 512 KB SRAM, 16 KB Low-power SRAM on the chip, and works with external flash
* 30 (QFN40) or 22 (QFN32) programmable GPIOs, with support for SPI, UART, I2C, I2S, RMT, TWAI and PWM

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **MODULE** | **DESCRIPTION** | **CHIP EMBEDDED** | **DIMENSIONS** | **PINS** | **FLASH (MB)** | **PSRAM (MB)** | **ANETNNAE** | **DEVELOPMNT BOARD** |
| **ESP32-C6-MINI-1** | ESP32-C6-MINI-1 is a ESP32-C6 based module, which supports Wi-Fi 6 in 2.4 GHz band, Bluetooth 5, Zigbee 3.0 and Thread. It's small-sized and pin-to-pin compatible with the ESP32-C3-MINI series module. With low power consumption, it is an ideal choice for a variety of IoT devices. | ESP32-C6FH4 ESP32-C6FH8 | 13.2×16.6×2.4 | 53 | 4, 8 | N/A | PCB antenna | ESP32-C6-DevKitM-1 |
| **ESP32-C6-MINI-1U** | ESP32-C6-WROOM-1 is a ESP32-C6 based module, which supports Wi-Fi 6 in 2.4 GHz band, Bluetooth 5, Zigbee 3.0 and Thread. It's pin-to-pin compatible with the ESP32-WROOM series module. With low power consumption, it is an ideal choice for a variety of IoT devices. | ESP32-C6 | 18×25.5×3.2 | 28 | 4, 8, 16 | N/A | PCB antenna | ESP32-C6-DevKitC-1 |
| **ESP32-C6-WROOM-1** | ESP32-C6-WROOM-1 is a ESP32-C6 based module, which supports Wi-Fi 6 in 2.4 GHz band, Bluetooth 5, Zigbee 3.0 and Thread. It's pin-to-pin compatible with the ESP32-WROOM series module. With low power consumption, it is an ideal choice for a variety of IoT devices. | ESP32-C6 | 18×25.5×3.2 | 28 | 4, 8, 16 | N/A | PCB antenna | ESP32-C6-DevKitC-1 |
| **ESP32-C6-WROOM-1U** | ESP32-C6-WROOM-1U is a ESP32-C6 based module, which supports Wi-Fi 6 in 2.4 GHz band, Bluetooth 5, Zigbee 3.0 and Thread. It's pin-to-pin compatible with the ESP32-WROOM series module. With low power consumption, it is an ideal choice for a variety of IoT devices. | ESP32-C6 | 18x19.2x3.2 | 28 | 4，8，16 | N/A | IPEX antenna connector | ESP32-C6-DevKitC-1 |

* ***Main Features- ESP32-C61 Series***

*32-bit RISC-V MCU & 2.4 GHz Wi-Fi 6 & Bluetooth 5 (LE)*

* Ultra-low-power MCU with 32-bit RISC-V single-core microprocessor
* 2.4 GHz Wi-Fi 6 (802.11ax), Bluetooth® 5 (LE)
* 3.3 V flash or PSRAM in the chip’s package
* 25 GPIOs
* Security features: secure boot, flash and PSRAM encryption, cryptographic accelerators, Trusted Execution Environment (TEE), and ECDSA-based Digital Signature Peripheral

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **MODULE** | **DESCRIPTION** | **CHIP EMBEDDED** | **DIMENSIONS** | **PINS** | **FLASH (MB)** | **PSRAM (MB)** | **ANETNNAE** | **DEVELOPMNT BOARD** |
| **ESP32-C61-WROOM-1** | A powerful, general-purpose 2.4G Wi-Fi 6 and Bluetooth LE module | ESP32-C61HR2 | 18.0 × 25.5 × 3.1 | 29 | 8 | 2 | PCB antenna | ESP32-C61-DevKitC-1-N8R2 |
| **ESP32-C61-WROOM-1U** | A powerful, general-purpose 2.4G Wi-Fi 6 and Bluetooth LE module, an ideal choice for smart homes, industrial automation, health care, consumer electronics, etc. | ESP32-C61HR2 | 18.0 × 25.5 × 3.1 | 29 | 8 | 2 | IPEX Antenna Connector | ESP32-C61-DevKitC-1 |

* ***Main Features- ESP32-C5 Series***

*32-bit RISC-V MCU & 2.4 and 5 GHz Wi-Fi 6 & Bluetooth 5 (LE) & IEEE 802.15.4*

* Ultra-low-power MCU with 32-bit RISC-V single-core microprocessor
* 2.4 and 5 GHz dual-band Wi-Fi 6 (802.11ax), Bluetooth® 5 (LE), Zigbee, and Thread (802.15.4)
* Support connection to external flash and PSRAM
* 29 GPIOs, rich set of peripherals
* Security features: secure boot, flash and PSRAM encryption, cryptographic accelerators, Access Permission Management (APM) hardware block and Physical Memory Protection (PMP)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **MODULE** | **DESCRIPTION** | **CHIP EMBEDDED** | **DIMENSIONS** | **PINS** | **FLASH (MB)** | **PSRAM (MB)** | **ANETNNAE** | **DEVELOPMNT BOARD** |
| **ESP32-C5-WROOM-1** | A general-purpose 2.4 and 5 GHz dual-band Wi-Fi 6 (802.11ax), Bluetooth® 5 (LE), Zigbee, and Thread (802.15.4) module, ideal for smart homes, industrial automation, health care, consumer electronics, etc | ESP32-C5NR4 ESP32-C5NF4 | 18×27.5×3.3 | 29 | 4,8,16 | 4 N/A | PCB antenna | ESP32-C5-DevKitC-1 |
| **ESP32-C5-WROOM-1U** | A general-purpose 2.4 and 5 GHz dual-band Wi-Fi 6 (802.11ax), Bluetooth® 5 (LE), Zigbee, and Thread (802.15.4) module, ideal for smart homes, industrial automation, health care, consumer electronics, etc | ESP32-C5NR4 | 18×27.5×3.3 | 29 | 8 | 4 | IPEX antenna connector | ESP32-C5-DevKitC-1 |

* ***Main Features- ESP32-C3 Series***

*32-bit RISC-V MCU & 2.4 GHz Wi-Fi & Bluetooth 5 (LE)*

* ESP32-C3 embedded, 32-bit RISC-V single-core processor, up to 160 MHz
* State-of-the-art power and RF performance
* 400KB of SRAM and 384 KB of ROM on the chip, and SPI, Dual SPI, Quad SPI, and QPI interfaces that allow connection to flash
* Reliable security features ensured by RSA-3072-based secure boot, AES-128-XTS-based flash encryption, the innovative digital signature and the HMAC peripheral, hardware acceleration support for cryptographic algorithms
* Rich set of peripheral interfaces and GPIOs, ideal for various scenarios and complex applications
* Fully certified with integrated antenna and software stacks

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **MODULE** | **DESCRIPTION** | **CHIP EMBEDDED** | **DIMENSIONS** | **PINS** | **FLASH (MB)** | **PSRAM (MB)** | **ANETNNAE** | **DEVELOPMNT BOARD** |
| **ESP32-C3-MINI-1** | ESP32-C3-MINI-1 is a general-purpose Wi-Fi and Bluetooth LE module. This module features a rich set of peripherals and a small size, which makes it an ideal choice for smart home, industrial automation, health care, consumer electronics, etc. | ESP32-C3FH4 ESP32-C3FH4X | 13.2×16.6×2.4 | 53 | 4 MB embedded in chip | N/A | PCB antenna | ESP32-C3-DevKitM-1  ESP32-C3-AWS-ExpressLink-DevKit  ESP32-C3-DevKit-RUST-1 |
| **ESP32-C3-MINI-1U** | ESP32-C3-MINI-1U is a powerful, generic Wi-Fi + Bluetooth LE MCU modules that have a single core CPU, a rich set of peripherals, and with an U.FL connector for external antenna. According to different chips integrated, ESP32-C3-MINI-1U has two variants operating in different ambient temperature. It is an ideal choice for smart homes, industrial automation, health care, consumer electronics, etc. | ESP32-C3FH4X ESP32-C3FH4 ESP32-C3-MINI-1U-N4X | 13.2×12.5×2.4 | 53 | 4 MB embedded in chip | N/A | IPEX antenna connector | ESP32-C3-DevKitM-1 |
| **ESP32-C3-WROOM-02** | ESP32-C3-WROOM-02 is a general-purpose Wi-Fi and Bluetooth LE module. This module features a rich set of peripherals and pin-to-pin compatible with the ESP-WROOM-02/02D modules. It is an ideal choice for smart home, industrial automation, healthcare, consumer electronics, etc. | ESP32-C3 | 18×20×3.2 | 19 | 4 | N/A | PCB antenna | ESP32-C3-DevKitC-02  ESP32-C3-Lyra |
| **ESP32-C3-WROOM-02U** | ESP32-C3-WROOM-02U is a powerful, generic Wi-Fi + Bluetooth LE MCU modules that have a single core CPU, a rich set of peripherals, and with an U.FL connector for external antenna. | ESP32-C3 | 18×14.3×3.2 | 19 | 4 | N/A | IPEX antenna connector | ESP32-C3-DevKitC-02 |

1. ***ESP32- S Series -*** Focuses on lower power and smaller footprint, often used in battery-powered applications.

* ***Main Features-ESP32-S3 Series***

*32-bit MCU & 2.4 GHz Wi-Fi & Bluetooth 5 (LE)*

* 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, DAC and UART, SD/MMC host and TWAI™
* Reliable security features ensured by RSA-based secure boot, AES-XTS-based flash encryption, the innovative digital signature and the HMAC peripheral, “World Controller”
* Fully certified with integrated antenna and software stacks

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **MODULE** | **DESCRIPTION** | **CHIP EMBEDDED** | **DIMENSIONS** | **PINS** | **FLASH (MB)** | **PSRAM (MB)** | **ANETNNAE** | **DEVELOPMNT BOARD** |
| **ESP32-S3-WROOM-1** | ESP32-S3-WROOM-1 is a powerful, generic Wi-Fi + Bluetooth LE MCU module that has a Dual core CPU, a rich set of peripherals, and provides acceleration for neural network computing and signal processing workloads. It is an ideal choice for a wide variety of application scenarios related to AI + Internet of Things (AIoT), such as wake word detection and speech commands recognition, face detection and recognition, smart home, smart appliance, smart control panel, smart speaker etc. | ESP32-S3 ESP32-S3R2 ESP32-S3R8 | 18×25.5×3.1 | 41 | 4, 8, 16 | N/A 2 8 | PCB antenna | ESP32-DevKitC  ESP32-S3-DevKitC-1  ESP32-S3-BOX-3  ESP32-S3-BOX  ESP32-S3-EYE  ESP32-S3-Korvo-1  ESP32-S3-Korvo-2  ESP32-S3-LCD-EV-Board |
| **ESP32-S3-WROOM-1U** | ESP32-S3-WROOM-1U is a powerful, generic Wi-Fi + Bluetooth LE MCU module that has a Dual core CPU, a rich set of peripherals, and provides acceleration for neural network computing and signal processing workloads. It is an ideal choice for a wide variety of application scenarios related to AI + Internet of Things (AIoT). | ESP32-S3 ESP32-S3R2 ESP32-S3R8 | 18×19.2×3.2 | 41 | 4, 8, 16 | 0 2 8 | IPEX antenna connector | ESP32-S3-DevKitC-1 |
| **ESP32-S3-WROOM-2** | ESP32-S3-WROOM-2 is based on ESP32-S3R16V, with flash memory of Octal 32 MB and PSRAM memory of 16 MB. It provides acceleration for neural network computing and signal processing workloads. They are an ideal choice for a wide variety of application scenarios related to AI and Artificial Intelligence of Things (AIoT), such as wake word detection and speech commands recognition, face detection and recognition, smart home, smart appliances, smart control panel, smart speaker, etc. | ESP32-S3R16V ESP32-S3R8V (EOL) | 18×25.5×3.1 | 41 | 32 | 16 | PCB antenna | ESP32-S3-DevKitC-1 |
| **ESP32-S3-MINI-1** | ESP32-S3-MINI-1 is a powerful, generic Wi-Fi + Bluetooth LE MCU module that features a rich set of peripherals, yet an optimized size. It is an ideal choice for a wide variety of application scenarios related to Internet of Things (IoT), such as embedded systems, smart home, wearable electronics, etc. | ESP32-S3FN8 ESP32-S3FH4R2 | 15.4×20.5×2.4 | 65 | 8 MB 4MB embedded in chip | N/A 2 | PCB antenna | ESP32-S3-DevKitM-1  ESP32-S3-USB-OTG  ESP32-S3-USB-Bridge |
| **ESP32-S3-MINI-1U** | ESP32-S3-MINI-1U is a powerful, generic Wi-Fi + Bluetooth LE MCU module that features a rich set of peripherals, yet an optimized size. It is an ideal choice for a wide variety of application scenarios related to Internet of Things (IoT), such as embedded systems, smart home, wearable electronics, etc. | ESP32-S3FN8 ESP32-S3FH4R2 | 15.4×15.4×2.4 | 65 | 8 MB 4MB embedded in chip | N/A 2 | IPEX antenna connector | ESP32-S3-DevKitM-1 |

* ***Main Features-ESP32-S2 Series***

*32-bit MCU & 2.4 GHz Wi-Fi*

* ESP32-S2 embedded, Xtensa® single-core 32-bit LX7 microprocessor, up to 240 MHz
* 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
* Availability of common cloud connectivity agents and common product features shortens the time to market
* Fully certified with integrated antenna and software stacks

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **MODULE** | **DESCRIPTION** | **CHIP EMBEDDED** | **DIMENSIONS** | **PINS** | **FLASH (MB)** | **PSRAM (MB)** | **ANETNNAE** | **DEVELOPMNT BOARD** |
| **ESP32-S2-MINI-2** | ESP32-S2-MINI-2 is a powerful, generic Wi-Fi MCU modules, which integrates ESP32-S2 ECO1 with improved RF performance. | ESP32-S2FH4 ESP32-S2FN4R2 | 15.4×20×2.4 | 65 | 4 MB embedded in chip | N/A 2 | PCB antenna | ESP32-S2-DevKitM-1 |
| **ESP32-S2-MINI-2U** | ESP32-S2-MINI-2U is a powerful, generic Wi-Fi MCU modules, which integrates ESP32-S2 ECO1 with improved RF performance. | ESP32-S2FH4 ESP32-S2FN4R2 | 15.4×15.4×2.4 | 65 | 4 MB embedded in chip | N/A 2 | IPEX antenna connector | ESP32-S2-DevKitM-1 |
| **ESP32-S2-SOLO-2U** | ESP32-S2-SOLO-2U is a powerful, generic Wi-Fi MCU modules, which integrates ESP32-S2 ECO1 with improved RF performance. | ESP32-S2 ESP32-S2R2 | 18×19.2×3.2 | 41 | 4 | N/A 2 | IPEX antenna connector | ESP32-S2-DevKitC-1 |
| **ESP32-S2-MINI-1**  **ESP32-S2-MINI-2** | ESP32-S2-MINI-1 is a powerful, generic Wi-Fi MCU modules that have a rich set of peripherals. It's an ideal choice for a wide variety of application scenarios relating to Internet of Things (IoT), wearable electronics and smart home.  At the core of this module is ESP32-S2FH4 / ESP32-S2FN4R2, an Xtensa® 32-bit LX7 CPU that operates at up to 240 MHz. | ESP32-S2FH4 ESP32-S2FN4R2 | 15.4×20×2.4 | 65 | 4 MB embedded in chip | N/A 2 | PCB antenna | ESP32-S2-DevKitM-1 |
| **ESP32-S2-MINI-1U**  **ESP32-S2-MINI-2U** | ESP32-S2-MINI-1U is a powerful, generic Wi-Fi MCU modules that have a rich set of peripherals. It's an ideal choice for a wide variety of application scenarios relating to Internet of Things (IoT), wearable electronics and smart home.  At the core of this module is ESP32-S2FH4 / ESP32-S2FN4R2, an Xtensa® 32-bit LX7 CPU that operates at up to 240 MHz. | ESP32-S2FH4 ESP32-S2FN4R2 | 15.4×15.4×2.4 | 65 | 4 MB embedded in chip | N/A 2 | IPEX antenna connector | ESP32-S2-DevKitM-1 |
| **ESP32-S2-SOLO**  **ESP32-S2-SOLO-2** | ESP32-S2-SOLO is a powerful, generic Wi-Fi MCU modules that have a rich set of peripherals and pin-to-pin compatible with the ESP32-WROOM modules. Its an ideal choice for a wide variety of application scenarios relating to Internet of Things (IoT), wearable electronics and smart home. | ESP32-S2 ESP32-S2R2 | 18×25.5×3.1 | 40 | 4,8,16 | N/A 2 | PCB antenna | ESP32-S2-DevKitC-1 |
| **ESP32-S2-SOLO-U**  **ESP32-S2-MINI-2U**  **ESP32-S2-SOLO-2U** | ESP32-S2-SOLO-U is a powerful, generic Wi-Fi MCU modules that have a rich set of peripherals and pin-to-pin compatible with the ESP32-WROOM modules. Its an ideal choice for a wide variety of application scenarios relating to Internet of Things (IoT), wearable electronics and smart home. | ESP32-S2 ESP32-S2R2 | 18×19.2×3.2 | 40 | 4,8,16 | N/A 2 | IPEX antenna connector | ESP32-S2-DevKitC-1 |
| **ESP32-S2-WROVER**  **ESP32-S2-MINI-2U**  **ESP32-S2-SOLO-2U** | ESP32-S2-WROVER integrates ESP32-S2. It is a powerful, generic Wi-Fi MCU module that has a rich set of peripherals. | ESP32-S2 | 18x31x3.3 | 42 | 4,8,16 | 2 | PCB antenna | ESP32-S2-Kaluga-1  ESP32-S2-HMI-DevKit-1  ESP32-S2-Saola-1 |
| **ESP32-S2-WROVER-I**  **ESP32-S2-MINI-2U**  **ESP32-S2-SOLO-2U** | ESP32-S2-WROVER-I integrates ESP32-S2. It is a powerful, generic Wi-Fi MCU module that has a rich set of peripherals. | ESP32-S2 | 18x31x3.3 | 42 | 4,8,16 | 2 | IPEX antenna connector | ESP32-S2-Saola-1  ESP32-S2-Kaluga-1 |
| **ESP32-S2-WROOM**  **ESP32-S2-MINI-2U**  **ESP32-S2-SOLO-2U** | ESP32-S2-WROOM integrates ESP32-S2. It is a powerful, generic Wi-Fi MCU module that has a rich set of peripherals. | ESP32-S2 | 18x31x3.3 | 42 | 4,8,16 | N/A | PCB antenna | ESP32-S2-Saola-1 |
| **ESP32-S2-WROOM-I** | ESP32-S2-WROOM-I integrates ESP32-S2. It is a powerful, generic Wi-Fi MCU module that has a rich set of peripherals. | ESP32-S2 | 18x31x3.3 | 42 | 4,8,16 | N/A | IPEX antenna connector | ESP32-S2-Saola-1 |

*Advantages*

1. Has a dual core processor hence can handle multitasking better.
2. Has more GPIOs with support for ADC, DAC, capacitive touch, PWM, SPI, I2C, I2S UART etc.
3. Has Bluetooth and Wi-Fi support.
4. Has a larger memory capacity.
5. Has good hardware security features.
6. Has multiple low power modes.

*Disadvantages*

1. Has a high power consumption.
2. Can be complex in use for beginners.
3. Is slightly expensive.
4. Is wider in physical sense and hence can be a problem in compact designs.
5. It has no built in display driver unless connected via a Serial Peripheral Interface.
6. The Wi-Fi and Bluetooth share a common radio. When loaded, the microcontroller may end up experiencing a performance drop due to the internal conflict.
7. There is limited ADC accuracy and linearity especially when the Wi-Fi is active. This is because the ADC2 unit and Wi-Fi share internal resources. This may lead to a slack in performance and drop in accuracy when the two are used simultaneously.
8. It requires good power supply decoupling as it has sensitive analog and radio frequency performance.

**APPLICATIONS OF THE MICROCONTROLLERS TO VARIOUS SATELLITE SUBSYSTEMS.**

1. *Controller (On board computer - OBC)*

 The ESP32 is preferred due to its higher processing ability, multitasking and built-in peripheral support. It is however not radiation hardened and may need reset logic for reliability.

1. *Communication System*

The ESP32 would be preferred due to its Bluetooth and Wi-Fi capabilities. This will however only be useful for short range telemetry and data dump for ground testing. It has the capability of handling the bandwidth required by the hobby satellite but it would be better if it would be complemented by other modules especially for RF applications.

1. *Electrical Power System (EPS)*

ESP32 would be better used due to its multiple GPIOs, ADC  capabilities and low power modes. It can handle power monitoring with ADCs, control relays and switches. Deep sleep can be employed for power saving. It however needs efficient power management.

1. *Attitude Determination and Control System (ADCS)*

ESP32 can be used with external coprocessors for a more precise orientation control. It can be used with IMUs (Inertial Measurement Units), magnetometers, sun sensors and others. Its dual core allows for parallel sensor reading and processing.

1. *Payload*

The  ESP32 has a camera interface that can handle simple image capture and sensor interfacing. It can not, however, handle large images due to its limited image processing capacity and memory.

1. *Structure and Integration*

This mostly involves mechanical design and is not necessarily controlled by a microcontroller.

1. *Telemetry command and Data Handling*

The ESP32 is better as it can handle larger amounts of data better due to its dual core and faster speeds. It can run FreeRTOS allowing real time scheduling (e.g. sensor sampling, transmission, and logging). It has Wi-Fi and Bluetooth which are great for ground testing, OTA updates and short range communications. It supports UART, SPI, I2C, PWM, ADC, DAC that make interfacing with most telemetry sensors straightforward. It has good storage capacity and supports external SPI based micro SD card. It does have nonvolatile storage that can be used for storing telemetry logs without need of power supply. It also allows low power support with less current consumption for the sleep modes.

Its multiple GPIOs will allow room for connection to the multiple sensors e.g. temperature sensors, voltage and current sensors etc.

**CONCLUSION.**

All factors considered, the ESP32 is undoubtedly more useful and wide in its scope of application. This can even simply be based on the various improvements made by the manufacturers to address the shortcomings of the ESP8266 microcontroller. This has been demonstrated by the massive capabilities of the ESP32 microcontroller.

Taking into account all the advantages and disadvantages of the presented microcontrollers as applied in the various subsystems, the ESP32 would be suited to serve the ***Telemetry Command and Data Handling Subsystem.*** This subsystem is crucial for **monitoring the health, status, and performance** of the satellite and its subsystems. It collects data from sensors, formats it, and sends it to the ground station via a communication system. It could also store logs when needed.

The functionalities of the microcontroller as discussed above would be useful in this subsystem.