INFORMACION PINES ESP32

**Resumen de los pines**

Por lo tanto, es muy útil tener una tabla de referencia que nos proporcione información clara y precisa sobre qué pines son seguros para su utilización y cuáles tienen funciones específicas.

Os aconsejo tener esta tabla a mano y consultarla antes de empezar un proyecto con un ESP32, para evitarnos sustos por haber empleado un pin que no deberíamos usar.

| **GPIO** | **NAME** | **FUNCIONES** | **PUEDES USARLO** |
| --- | --- | --- | --- |
| 0 | IO0 | GPIO0, ADC2\_CH1, TOUCH\_CH1, Boot | ⚠️Pulled-UP. Debe ser HIGH al arranque. Debe ser LOW para flasheo. Genera PWM al arrancar |
| 1 | TX | GPIO1, U0TXD | ❌Pin TX. Salida de depuración al arrancar |
| 2 | IO2 | GPIO2, ADC2\_CH2, TOUCH\_CH2 | ⚠️Debe ser flotante o LOW para flasheo. Conectado al LED |
| 3 | RX | GPIO3, U0RXD | ❌Pin RX. En HIGH al arrancar |
| 4 | IO4 | GPIO4, ADC2\_CH0, TOUCH\_CH0 | ✔️ |
| 5 | IO5 | GPIO5 | ⚠️Debe ser HIGH al arrancar. Genera señal PWM al arrancar |
| 6 | CLK | GPIO6 | ❌FLASH SPI |
| 7 | D0 | GPIO7 | ❌FLASH SPI |
| 8 | D1 | GPIO8 | ❌FLASH SPI |
| 9 | D2 | GPIO9 | ❌FLASH SPI |
| 10 | D3 | GPIO10 | ❌FLASH SPI |
| 11 | CMD | GPIO11 | ❌FLASH SPI |
| 12 | IO12 | GPIO12, ADC2\_CH5, TOUCH\_CH5, MTDI | ⚠️Debe ser LOW al arrancar Debug JTAG |
| 13 | IO13 | GPIO13, ADC2\_CH4, TOUCH\_CH4, MTCK | 🟢Debug JTAG |
| 14 | IO14 | GPIO14, ADC2\_CH6, TOUCH\_CH6, MTMS | ⚠️Genera señal PWM al arrancar Debug JTAG |
| 15 | IO15 | GPIO15, ADC2\_CH3, TOUCH\_CH3, MTDO | ⚠️Debe ser HIGH al arrancar Debug JTAG |
| 16 | IO16 | GPIO16 | ✔️ |
| 17 | IO17 | GPIO17 | ✔️ |
| 18 | IO18 | GPIO18 | ✔️ |
| 19 | IO19 | GPIO19 | ✔️ |
| 21 | IO21 | GPIO21 | ✔️ |
| 22 | IO22 | GPIO22 | ✔️ |
| 23 | IO23 | GPIO23 | ✔️ |
| 25 | IO25 | GPIO25, ADC1\_CH8, DAC\_1 | ✔️ |
| 26 | IO26 | GPIO26, ADC2\_CH9, DAC\_2 | ✔️ |
| 27 | IO27 | GPIO27, ADC2\_CH7, TOUCH\_CH7 | ✔️ |
| 32 | IO32 | GPIO32, ADC1\_CH4, TOUCH\_CH9, XTAL\_32K\_P | ✔️ |
| 33 | IO33 | GPIO33, ADC1\_CH5, TOUCH\_CH8, XTAL\_32K\_N | ✔️ |
| 34 | IO34 | GPIO34, ADC1\_CH6, VDET\_1 | ⚠️Solo entrada |
| 35 | IO35 | GPIO35, ADC1\_CH7, VDET\_2 | ⚠️Solo entrada |
| 36 | VP | GPIO36, ADC1\_CH0, S\_VP | ⚠️Solo entrada |
| 37 | VP | GPIO37 | ⚠️Solo entrada |
| 38 | VP | GPIO38 | ⚠️Solo entrada |
| 39 | VN | GPIO39, ADC1\_CH3, S\_VN | ⚠️Solo entrada |
| EN | EN | CHIP\_PU, Reset | ❌ |

•Sin embargo, al igual que pasaba con el ESP8266, **no todos los pines del ESP32 son igualmente seguros** para usar en nuestros proyectos.

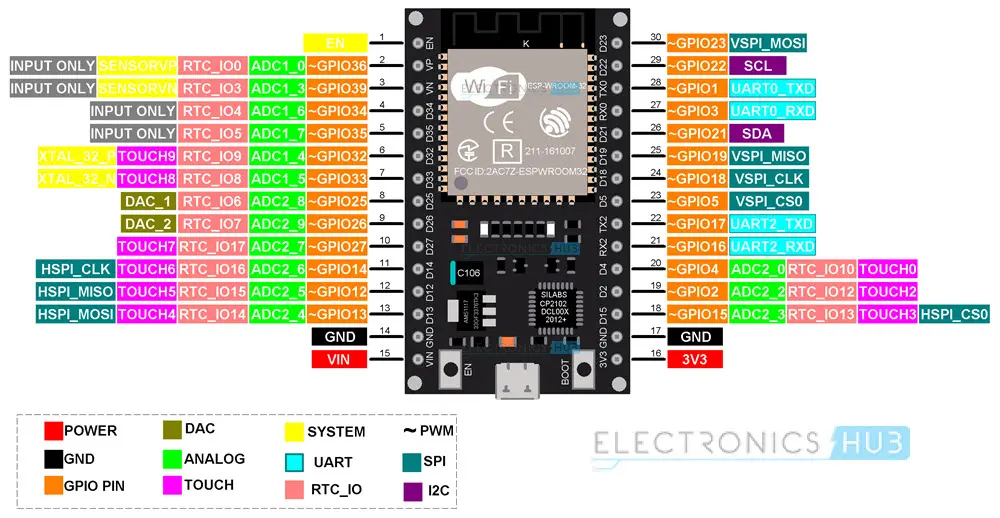
•Esto es debido debe a que algunos pines desempeñan roles específicos o participan en el proceso de arranque del dispositivo. Utilizar incorrectamente estos pines puede ocasionar conflictos que hagan que no funcione tu proyecto, o incluso que rompas algo.

• El pin 12 no utilizarlo, si lo haces tendrás problemas con las interferencias de los fluorescentes.

• **Es el que uso yo**

# •**ESP32 Pinout | ESP32 WROOM Pinouts.**





### •**Pines de Configuración (Strapping) del ESP32**

Existen cinco pines Strapping.

* GPIO 0 (debe ser LOW para entrar en modo de arranque)
* GPIO 2 (debe estar flotante o LOW durante el arranque)
* GPIO 4
* GPIO 5 (debe estar HIGH durante el arranque)
* GPIO 12 (debe ser LOW durante el arranque)
* GPIO 15 (debe estar en estado ALTO durante el arranque)

Estos pines intervienen en la configuración durante el arranque. Así que evita usarlos en tu proyecto.

### •**Pines GPIO de Entrada Únicamente**

Estos pines no pueden emplearse como salidas, pero se pueden utilizar como entradas digitales o analógicas, o para otros propósitos.

* GPIO 34
* GPIO 35
* GPIO 36
* GPIO 39

Además, a diferencia de los otros pines GPIO, carecen de resistencias internas de pull-up y pull-down.

### •**Pines de Interrupción del ESP32**

Todos los pines GPIO se pueden configurar como interrupciones.

### •**Pines GPIO RTC del ESP32**

Algunos GPIO están conectados al subsistema de bajo consumo RTC y se conocen como GPIO RTC

* RTC\_GPIO0 - GPIO 36
* RTC\_GPIO3 - GPIO 39
* RTC\_GPIO4 - GPIO 34
* RTC\_GPIO5 - GPIO 35
* RTC\_GPIO6 - GPIO 25
* RTC\_GPIO7 - GPIO 26
* RTC\_GPIO8 - GPIO 33
* RTC\_GPIO9 - GPIO 32
* RTC\_GPIO10 - GPIO 4
* RTC\_GPIO11 - GPIO 0
* RTC\_GPIO12 - GPIO 2
* RTC\_GPIO13 - GPIO 15
* RTC\_GPIO14 - GPIO 13
* RTC\_GPIO15 - GPIO 12
* RTC\_GPIO16 - GPIO 14
* RTC\_GPIO17 - GPIO 27

Estos pines se utilizan para despertar el ESP32 del modo de bajo consumo profundo cuando el coprocesador de ultra bajo consumo (ULP) está en funcionamiento.

### •**Pines de táctiles (Touch) del ESP32**

El ESP32 cuenta con 10 pines GPIO capacitivos de detección táctil.

* TOUCH0 - GPIO 4
* TOUCH1 - GPIO 0
* TOUCH2 - GPIO 2
* TOUCH3 - GPIO 15
* TOUCH4 - GPIO 13
* TOUCH5 - GPIO 12
* TOUCH6 - GPIO 14
* TOUCH7 - GPIO 27
* TOUCH8 - GPIO 33
* TOUCH9 - GPIO 32

Cuando una carga capacitiva (como un dedo humano) está cerca del pin GPIO, el ESP32-S3 detecta el cambio en la capacitancia.

### •**Pin Enable (EN)**

Enable (EN) es el pin que controla el regulador de 3V3. Está configurado con una resistencia pull-up, por lo que se conecta a tierra para desactivar el regulador de 3.3V. Por ejemplo, para reiniciar el ESP32.

## •**Pines analógicos**

**Pines PWM del ESP32**

La placa ESP32 tiene 16 canales PWM controlados por un controlador PWM. La salida PWM se puede utilizar para controlar motores y LEDs digitales.

### •**Pines ADC del ESP32**

El ESP32 integra dos ADC y admite mediciones en 18 canales

* ADC1\_CH0 - GPIO 36
* ADC1\_CH1 - GPIO 37
* ADC1\_CH2 - GPIO 38
* ADC1\_CH3 - GPIO 39
* ADC1\_CH4 - GPIO 32
* ADC1\_CH5 - GPIO 33
* ADC1\_CH6 - GPIO 34
* ADC1\_CH7 - GPIO 35
* ADC2\_CH0 - GPIO 4
* ADC2\_CH1 - GPIO 0
* ADC2\_CH2 - GPIO 2
* ADC2\_CH3 - GPIO 15
* ADC2\_CH4 - GPIO 13
* ADC2\_CH5 - GPIO 12
* ADC2\_CH6 - GPIO 14
* ADC2\_CH7 - GPIO 27
* ADC2\_CH8 - GPIO 25
* ADC2\_CH9 - GPIO 26

Los ADC son de 12 bits, por lo que disponemos de 4096 (2^12) niveles discretos, lo que se traduce en una precisión de 0.8mV.

### •**Pines DAC del ESP32**

El ESP32 incluye dos canales DAC para convertir señales digitales en voltajes analógicos.

* DAC1 - GPIO 25
* DAC2 - GPIO 26

Estos DAC tienen una resolución de 8 bits, lo que significa que los valores que van desde 0 hasta 255 se convierten en un voltaje analógico que va desde 0 hasta 3.3V.

## •**Pines de comunicación**

**Pines UART del ESP32**

La placa de desarrollo ESP32-S3 tiene tres interfaces UART: UART0, UART1 y UART2, que admiten comunicación asíncrona (RS232 y RS485) e IrDA de hasta 5 Mbps.

Los pines UART0 están conectados al convertidor USB a serie y se utilizan para la programación y depuración. No se recomienda utilizar los pines UART0.

Por otro lado, el UART están reservados para el chip de memoria FLASH integrado. Los pines UART1 están reservados para el chip de memoria flash integrado.

Por lo que lo mejor es usar el UART2 es una opción segura para conectar dispositivos UART.

### •**Pines I2C del ESP32**

El ESP32 dispone de un bus I2C único que permite conectar hasta 112 sensores y periféricos. Los pines SDA y SCL están asignados por defecto a los siguientes pines.

* SDA - GPIO 21
* SCL - GPIO 22

Sin embargo, es posible utilizar cualquier pin GPIO para implementar el protocolo I2C mediante el comando wire.begin(SDA, SCL).

### •**Pines SPI del ESP32**

El ESP32 cuenta con tres interfaces SPI (SPI, HSPI y VSPI). Solo las interfaces VSPI y HSPI son interfaces SPI utilizables, y el tercer bus SPI se utiliza para la memoria flash integrada.

| **SPI** | **MOSI** | **MISO** | **CLK** | **CS** |
| --- | --- | --- | --- | --- |
| VSPI | GPIO 23 | GPIO 19 | GPIO 18 | GPIO 5 |
| HSPI | GPIO 13 | GPIO 12 | GPIO 14 | GPIO 15 |

Los pines VSPI son comúnmente utilizados en bibliotecas estándar.

### •**SPI FLASH**

Estos pines están conectados a la memoria flash SPI integrada en el chip ESP32-S3. No utilices estos pines en tus proyectos.

* GPIO 6 (SCK/CLK)
* GPIO 7 (SDO/SD0)
* GPIO 8 (SDI/SD1)
* GPIO 9 (SHD/SD2)
* GPIO 10 (SWP/SD3)
* GPIO 11 (CSC/CMD)

•Caracteristicas ESP32

| **Feature** | **ESP32 Series** |
| --- | --- |
| Launch year | 2016 |
| Core | Xtensa® dual-/single core 32-bit LX6 |
| Wi-Fi protocols | 802.11 b/g/n, 2.4 GHz |
| Bluetooth® | Bluetooth v4.2 y BLE |
| Typical frequency | 240 MHz (160 MHz for ESP32-S0WD) |
| SRAM | 520 KB |
| ROM | 448 KB |
| External flash | Up to 16 MB device |
| External RAM | Up to 8 MB device |
| GPIO | 34 |
| ADC | 2x 12-bit |
| DAC | 2x 8-bit |
| Timers | 4x 64-bit, 3x watchdog |
| Temperature sensor | ✖️ |
| SPI | 4 |
| LCD interface | 1 |
| UART | 3 |
| I2C | 2 |
| I2S | 2 |
| LED PWM | 16 |
| Pulse counter | 8 |
| Touch sensor | 10 |
| Hall sensor | 1 |
| Camera interface | 1 |

•PINOUT Esp32 38 pines

