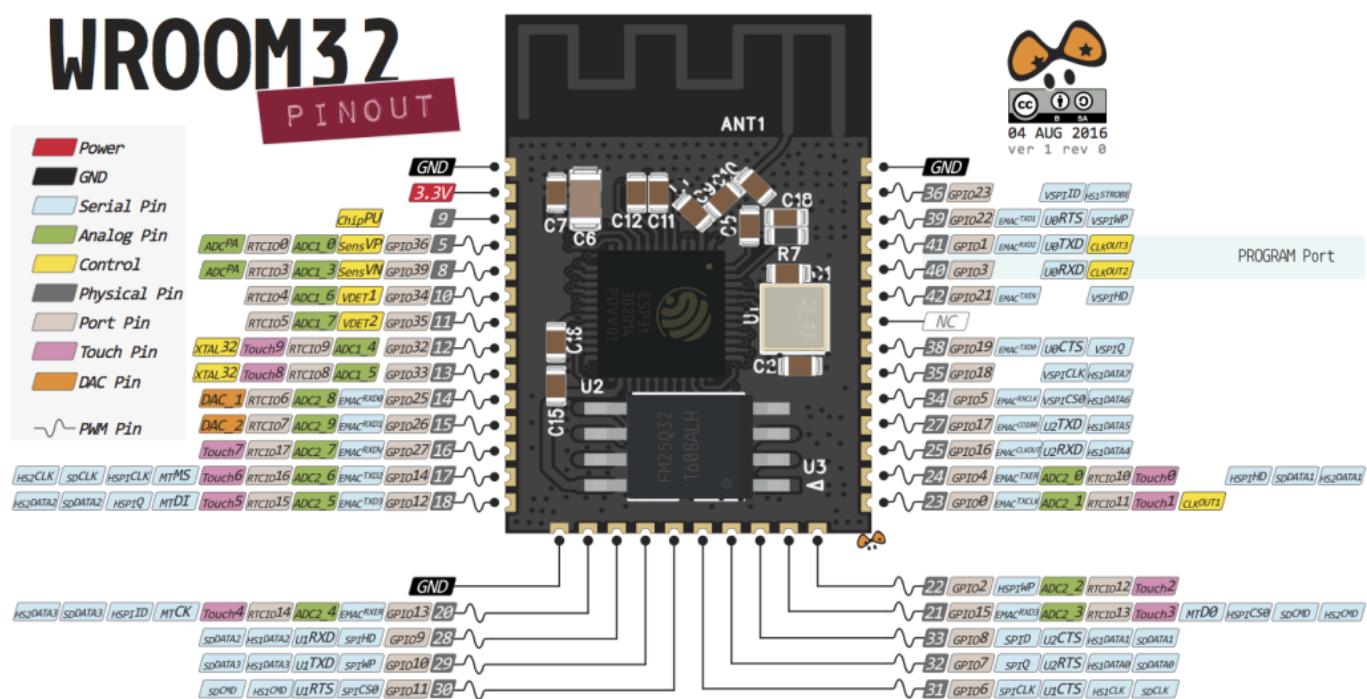


# ESP32 Pinout Reference: Which GPIO pins should you use?

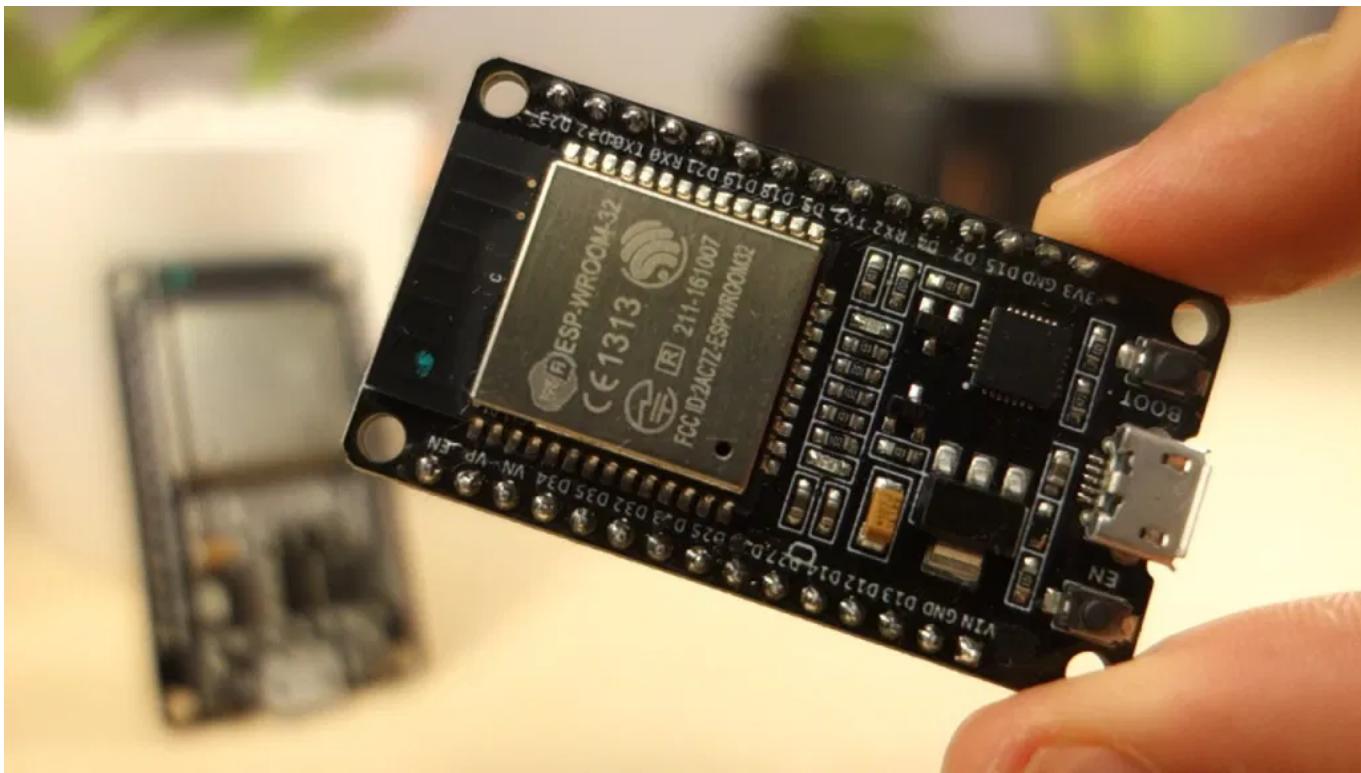
The ESP32 chip comes with 48 pins with multiple functions. Not all pins are exposed in all ESP32 development boards, and there are some pins that cannot be used.

There are many questions on how to use the ESP32 GPIOs. What pins should you use? What pins should you avoid using in your projects? This post aims to be a simple and easy to follow reference guide for the ESP32 GPIOs.

The figure below illustrates the ESP-WROOM-32 pinout. You can use it as a reference if you're using an **ESP32 bare chip** to build a custom board:



**Note:** not all GPIOs are accessible in all development boards, but each specific GPIO works in the same way regardless of the development board you're using. If you're just getting started with the ESP32, we recommend reading our guide: [Getting Started with the ESP32 Development Board](#).



## ESP32 Peripherals

The ESP32 peripherals include:

- [18 Analog-to-Digital Converter \(ADC\) channels](#)
- 3 SPI interfaces
- 3 UART interfaces
- [2 I2C interfaces](#)
- [16 PWM output channels](#)
- 2 Digital-to-Analog Converters (DAC)
- 2 I2S interfaces
- [10 Capacitive sensing GPIOs](#)

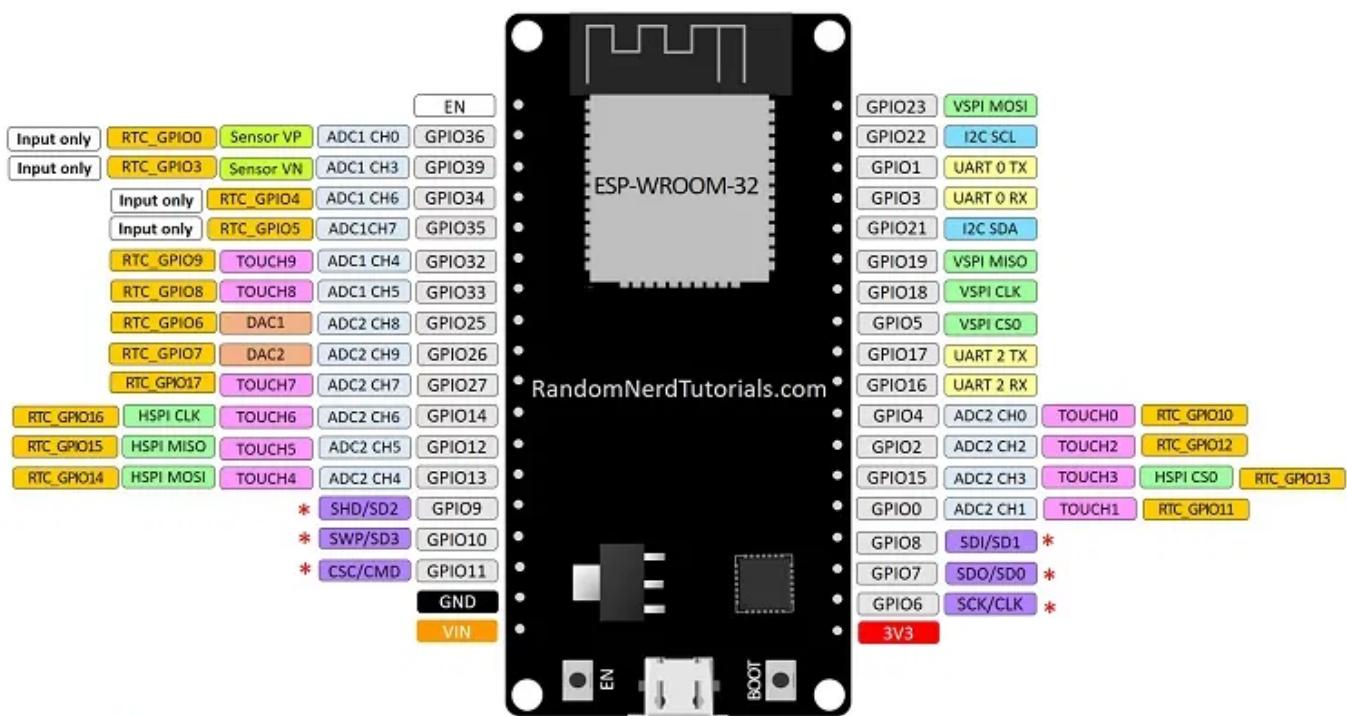
The ADC (analog to digital converter) and DAC (digital to analog converter) features are assigned to specific static pins. However, you can decide which pins are UART, I2C, SPI, PWM, etc – you just need to assign them in the code. This is possible due to the ESP32 chip's multiplexing feature.

Although you can define the pins properties on the software, there are pins assigned by default as shown in the following figure (this is an example for the [ESP32 DEVKIT](#)

V1 DOIT board with 36 pins – the pin location can change depending on the manufacturer).

# ESP32 DEVKIT V1 – DOIT

version with 36 GPIOs



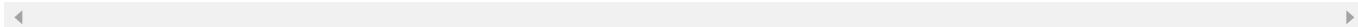
Additionally, there are pins with specific features that make them suitable or not for a specific project. The following table shows what pins are best to use as inputs, outputs and which ones you need to be cautious.

The pins highlighted in green are OK to use. The ones highlighted in yellow are OK to use, but you need to pay attention because they may have unexpected behavior mainly at boot. The pins highlighted in red are not recommended to use as inputs or outputs.

GPIO	Input	Output	Notes
0	pulled up	OK	outputs PWM signal at boot
1	TX pin	OK	debug output at boot

<b>2</b>	OK	OK	connected to on-board LED
<b>3</b>	OK	RX pin	HIGH at boot
<b>4</b>	OK	OK	
<b>5</b>	OK	OK	outputs PWM signal at boot
<b>6</b>	X	X	connected to the integrated SPI flash
<b>7</b>	X	X	connected to the integrated SPI flash
<b>8</b>	X	X	connected to the integrated SPI flash
<b>9</b>	X	X	connected to the integrated SPI flash
<b>10</b>	X	X	connected to the integrated SPI flash
<b>11</b>	X	X	connected to the integrated SPI flash
<b>12</b>	OK	OK	boot fail if pulled high
<b>13</b>	OK	OK	
<b>14</b>	OK	OK	outputs PWM signal at boot
<b>15</b>	OK	OK	outputs PWM signal at boot
<b>16</b>	OK	OK	
<b>17</b>	OK	OK	
<b>18</b>	OK	OK	
<b>19</b>	OK	OK	
<b>21</b>	OK	OK	
<b>22</b>	OK	OK	
<b>23</b>	OK	OK	
<b>25</b>	OK	OK	

<b>26</b>	OK	OK	
<b>27</b>	OK	OK	
<b>32</b>	OK	OK	
<b>33</b>	OK	OK	
<b>34</b>	OK		input only
<b>35</b>	OK		input only
<b>36</b>	OK		input only
<b>39</b>	OK		input only



Continue reading for a more detail and in-depth analysis of the ESP32 GPIOs and its functions.

## Input only pins

GPIOs 34 to 39 are GPIOs – input only pins. These pins don't have internal pull-ups or pull-down resistors. They can't be used as outputs, so use these pins only as inputs:

- GPIO 34
- GPIO 35
- GPIO 36
- GPIO 39

## SPI flash integrated on the ESP-WROOM-32

GPIO 6 to GPIO 11 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. So, don't use these pins in your projects:

- GPIO 6 (SCK/CLK)
- GPIO 7 (SDO/SD0)
- GPIO 8 (SDI/SD1)

- GPIO 9 (SHD/SD2)
- GPIO 10 (SWP/SD3)
- GPIO 11 (CSC/CMD)

## Capacitive touch GPIOs

The ESP32 has 10 internal capacitive touch sensors. These can sense variations in anything that holds an electrical charge, like the human skin. So they can detect variations induced when touching the GPIOs with a finger. These pins can be easily integrated into capacitive pads, and replace mechanical buttons. The capacitive touch pins can also be used to [wake up the ESP32 from deep sleep](#).

Those internal touch sensors are connected to these GPIOs:

- T0 (GPIO 4)
- T1 (GPIO 0)
- T2 (GPIO 2)
- T3 (GPIO 15)
- T4 (GPIO 13)
- T5 (GPIO 12)
- T6 (GPIO 14)
- T7 (GPIO 27)
- T8 (GPIO 33)
- T9 (GPIO 32)

[Learn how to use the touch pins with Arduino IDE: ESP32 Touch Pins with Arduino IDE](#)

## Analog to Digital Converter (ADC)

The ESP32 has 18 x 12 bits ADC input channels (while the ESP8266 only has 1x 10 bits ADC). These are the GPIOs that can be used as ADC and respective channels:

- 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)

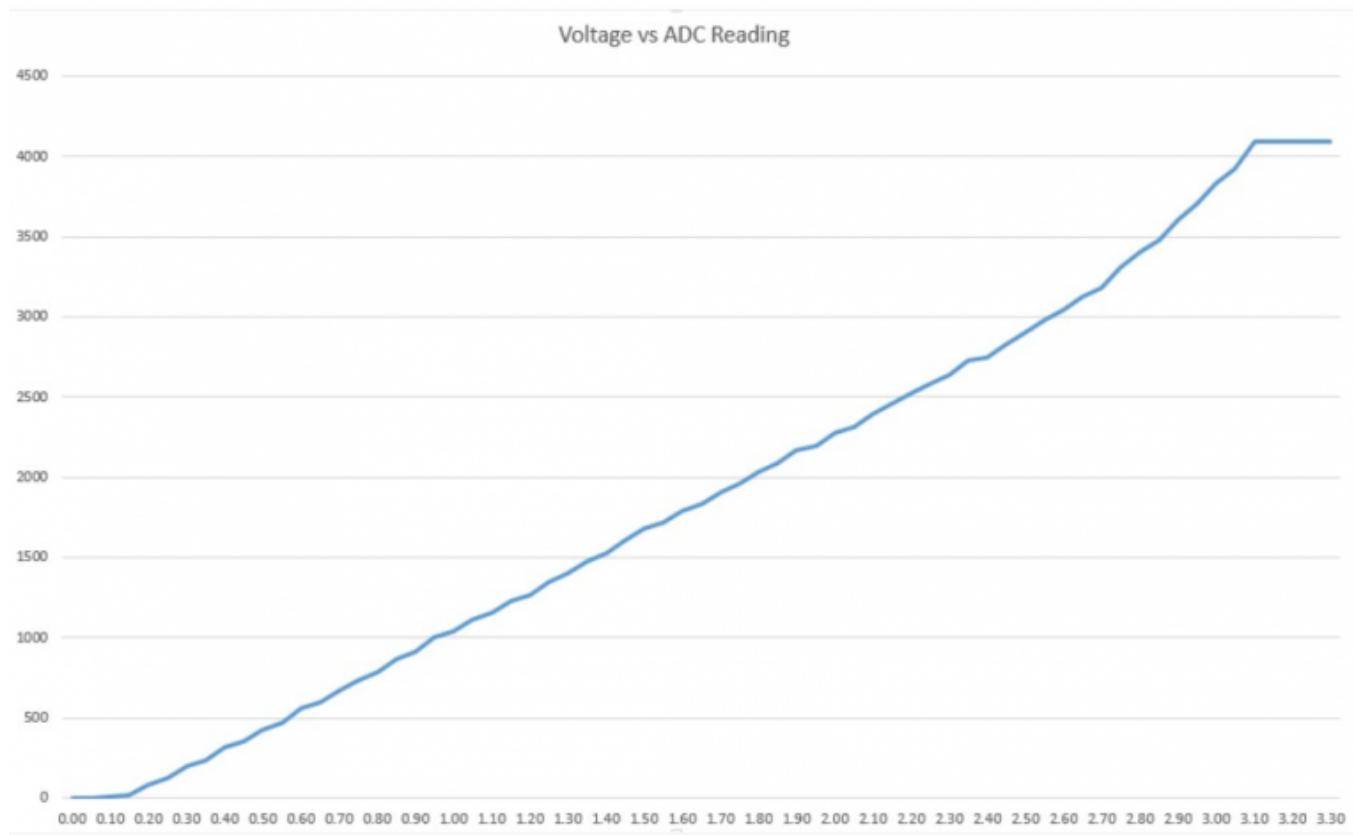
**Learn how to use the ESP32 ADC pins:**

- [ESP32 ADC Pins with Arduino IDE](#)
- [ESP32 ADC Pins with MicroPython](#)

**Note:** ADC2 pins cannot be used when Wi-Fi is used. So, if you're using Wi-Fi and you're having trouble getting the value from an ADC2 GPIO, you may consider using an ADC1 GPIO instead, that should solve your problem.

The ADC input channels have a 12 bit resolution. This means that you can get analog readings ranging from 0 to 4095, in which 0 corresponds to 0V and 4095 to 3.3V. You also have the ability to set the resolution of your channels on the code, as well as the ADC range.

The ESP32 ADC pins don't have a linear behavior. You'll probably won't be able to distinguish between 0 and 0.1V, or between 3.2 and 3.3V. You need to keep that in mind when using the ADC pins. You'll get a behavior similar to the one shown in the following figure.

[View source](#)

## Digital to Analog Converter (DAC)

There are 2 x 8 bits DAC channels on the ESP32 to convert digital signals into analog voltage signal outputs. These are the DAC channels:

- DAC1 (GPIO25)
- DAC2 (GPIO26)

## RTC GPIOs

There is RTC GPIO support on the ESP32. The GPIOs routed to the RTC low-power subsystem can be used when the ESP32 is in deep sleep. These RTC GPIOs can be used to wake up the ESP32 from deep sleep when the Ultra Low Power (ULP) co-processor is running. The following GPIOs can be used as an [external wake up source](#).

- RTC\_GPIO0 (GPIO36)
- RTC\_GPIO3 (GPIO39)

- RTC\_GPIO4 (GPIO34)
- RTC\_GPIO5 (GPIO35)
- RTC\_GPIO6 (GPIO25)
- RTC\_GPIO7 (GPIO26)
- RTC\_GPIO8 (GPIO33)
- RTC\_GPIO9 (GPIO32)
- RTC\_GPIO10 (GPIO4)
- RTC\_GPIO11 (GPIO0)
- RTC\_GPIO12 (GPIO2)
- RTC\_GPIO13 (GPIO15)
- RTC\_GPIO14 (GPIO13)
- RTC\_GPIO15 (GPIO12)
- RTC\_GPIO16 (GPIO14)
- RTC\_GPIO17 (GPIO27)

**Learn how to use the RTC GPIOs to wake up the ESP32 from deep sleep:**  
[ESP32 Deep Sleep with Arduino IDE and Wake Up Sources](#)

## PWM

The ESP32 LED PWM controller has 16 independent channels that can be configured to generate PWM signals with different properties. All pins that can act as outputs can be used as PWM pins (GPIOs 34 to 39 can't generate PWM).

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.

**Learn how to use ESP32 PWM with Arduino IDE:** [ESP32 PWM with Arduino IDE](#)

## I2C

The ESP32 has two I2C channels and any pin can be set as SDA or SCL. When using the ESP32 with the Arduino IDE, the default I2C pins are:

- GPIO 21 (SDA)
- GPIO 22 (SCL)

If you want to use other pins, when using the wire library, you just need to call:

```
Wire.begin(SDA, SCL);
```

**Learn more about I2C communication protocol with the ESP32 using Arduino IDE: [ESP32 I2C Communication \(Set Pins, Multiple Bus Interfaces and Peripherals\)](#)**

## SPI

By default, the pin mapping for SPI is:

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

## Interrupts

All GPIOs can be configured as interrupts.

**Learn how to use interrupts with the ESP32:**

- [ESP32 interrupts with Arduino IDE](#)
- [ESP32 interrupts with MicroPython](#)

## Strapping Pins

The ESP32 chip has the following strapping pins:

- GPIO 0
- GPIO 2

- GPIO 4
- GPIO 5 (must be HIGH during boot)
- GPIO 12 (must be LOW during boot)
- GPIO 15 (must be HIGH during boot)

These are used to put the ESP32 into bootloader or flashing mode. On most development boards with built-in USB/Serial, you don't need to worry about the state of these pins. The board puts the pins in the right state for flashing or boot mode.

More information on the [ESP32 Boot Mode Selection can be found here](#).

However, if you have peripherals connected to those pins, you may have trouble trying to upload new code, flashing the ESP32 with new firmware or resetting the board. If you have some peripherals connected to the strapping pins and you are getting trouble uploading code or flashing the ESP32, it may be because those peripherals are preventing the ESP32 to enter the right mode. Read the [Boot Mode Selection documentation](#) to guide you in the right direction. After resetting, flashing, or booting, those pins work as expected.

## Pins HIGH at Boot

Some GPIOs change its state to HIGH or output PWM signals at boot or reset. This means that if you have outputs connected to these GPIOs you may get unexpected results when the ESP32 resets or boots.

- GPIO 1
- GPIO 3
- GPIO 5
- GPIO 6 to GPIO 11 (connected to the ESP32 integrated SPI flash memory – not recommended to use).
- GPIO 14
- GPIO 15

## Enable (EN)

Enable (EN) is the 3.3V regulator's enable pin. It's pulled up, so connect to ground to disable the 3.3V regulator. This means that you can use this pin connected to a pushbutton to restart your ESP32, for example.

## GPIO current drawn

The absolute maximum current drawn per GPIO is 40mA according to the “Recommended Operating Conditions” section in the ESP32 datasheet.

## ESP32 Built-In Hall Effect Sensor

The ESP32 also features a [built-in hall effect sensor](#) that detects changes in the magnetic field in its surroundings.

## Wrapping Up

We hope you've found this reference guide for the ESP32 GPIOs useful. If you have more tips about the ESP32 GPIOs, please share by writing a comment down below.

If you're just getting started with the ESP32, we have some great content to get started:

- [Learn ESP32 with Arduino IDE](#)
- [Getting Started with the ESP32 Development Board](#)
- [20+ ESP32 Projects and Tutorials](#)
- [ESP32 Web Server Tutorial](#)
- [ESP32 vs ESP8266 – Pros and Cons](#)

Thanks for reading.

PCBWay HIGH-QUALITY PCB  
ONLY \$5 FOR 10 PIECES

- Rogers, HDI, aluminum and rigid-flex PCB are available now
- Production time 24 hours

PCB ASSEMBLY  
Free shipping + Free stencil  
ONLY \$30

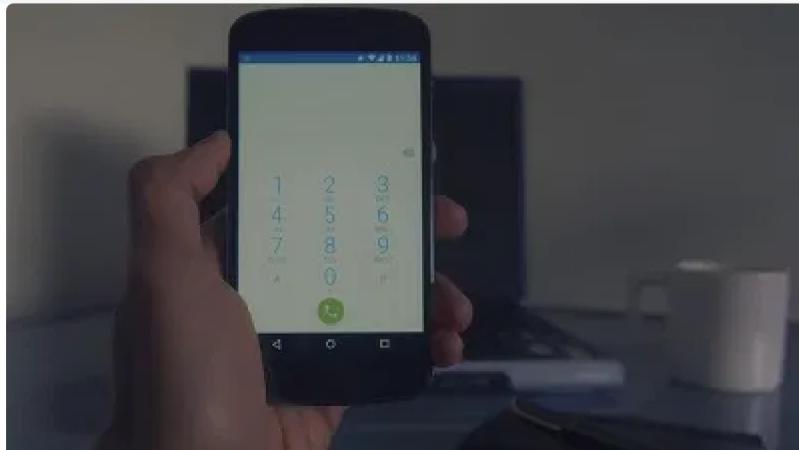
- Component sourcing
- Quality assurance

[eBook] Build Web Servers with ESP32 and ESP8266 (2nd Edition)

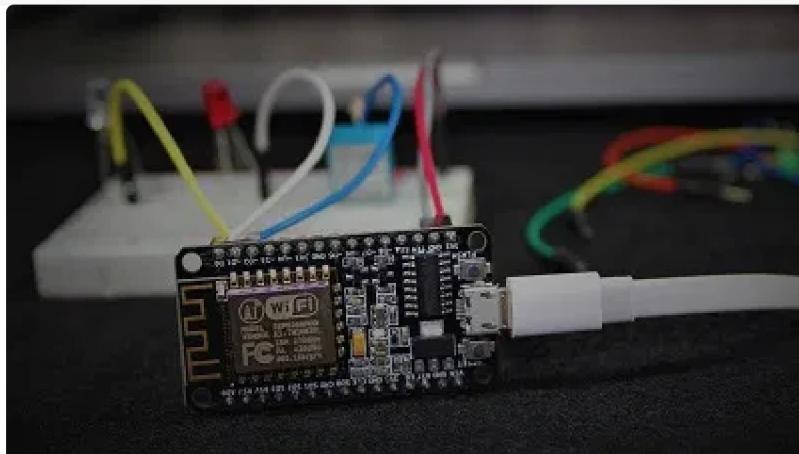


Build Web Server projects with the ESP32 and ESP8266 boards to control outputs and monitor sensors remotely. Learn HTML, CSS, JavaScript and client-server communication protocols [DOWNLOAD »](#)

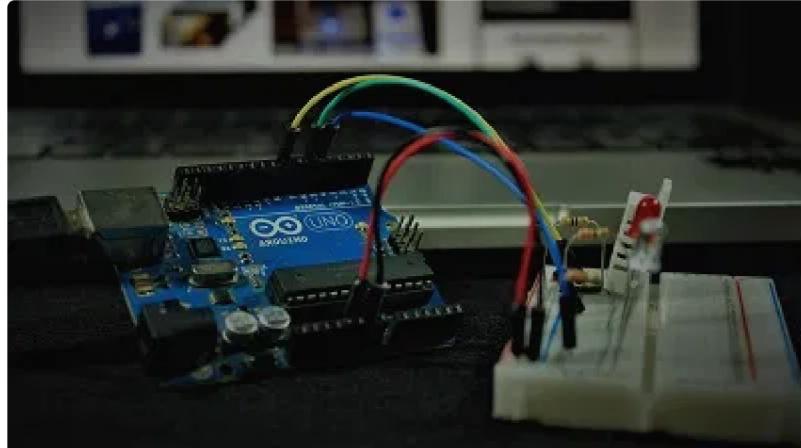
## Recommended Resources



[Build a Home Automation System from Scratch »](#) With Raspberry Pi, ESP8266, Arduino, and Node-RED.



[Home Automation using ESP8266 eBook and video course »](#) Build IoT and home automation projects.



[Arduino Step-by-Step Projects »](#) Build 25 Arduino projects with our course, even with no prior experience!

## What to Read Next...

---

[Get ESP32/ESP8266 MAC Address and Change It \(Arduino IDE\)](#)

---

[ESP32 Web Server with MPU-6050 Accelerometer and Gyroscope \(3D object representation\)](#)

---

[ESP32: ESP-NOW Web Server Sensor Dashboard \(ESP-NOW + Wi-Fi\)](#)

**Enjoyed this project? Stay updated by subscribing our newsletter!**

Your Email Address

SUBSCRIBE

## 129 thoughts on “ESP32 Pinout Reference: Which GPIO pins should you use?”

**Tobo**

August 30, 2018 at 11:50 am

Thanks, very usefull.

[Reply](#)**Sara Santos**

August 30, 2018 at 1:56 pm

Hi!

Thank you! 😊

[Reply](#)**Duncan Amos**

August 30, 2018 at 12:51 pm

Really, really useful – thank you...

[Reply](#)

**Sara Santos**

August 30, 2018 at 1:57 pm

Thanks 😊

[Reply](#)**carlos**

August 30, 2018 at 2:50 pm

Excellent , thanks i have been confused with this issue since there are so many pins and most have diffreneee possibilities. too many variations to remember them withoutu consulting

[Reply](#)**Doug Elliott**

August 30, 2018 at 4:43 pm

Very useful and well presented info. It might be useful to point out that the Adafruit Feather pinouts are not compatible with the similar looking DEVKIT V1 DOIT board. Reference:

[learn.adafruit.com/adafruit-huzzah32-esp32-feather/pinouts](https://learn.adafruit.com/adafruit-huzzah32-esp32-feather/pinouts)

[Reply](#)

**Sara Santos**

August 31, 2018 at 9:02 am

Hi Doug.

Yes, the Huzza 32 from Adafruit has a slightly different pinout.

But the guide you refer does a great job explaining each individual pin.

Thanks for sharing.

Regards,

Sara 😊

[Reply](#)**Keith G**

February 23, 2020 at 9:02 pm

Hi Doug,

I am absolutely brand new to this whole “trade” I guess I would call it, but I’m trying to get up to speed as quickly as I can to complete a 16 x 16 Led matrix graduation cap for my daughter, but everywhere I’ve looked there is very little or no information on this other than lots of videos that crank out music while showing someone’s finished project in operation. In summary, I have an ESP32 -WHOOM – 32D which is a 5v or 3.5v board with different pinouts than what I’ve seen here, so I was wondering if anyone has the same board ?

[Reply](#)**Sara Santos**

February 24, 2020 at 10:38 am

Hi Keith.

I think this is the pinout for that board:

<https://i0.wp.com/randomnerdtutorials.com/wp-content/uploads/2018/08/esp32-pinout-chip-ESP-WROOM-32.png>

Regards,

Sara

[Reply](#)



**Peter Calum**

August 31, 2018 at 6:52 am

Hi,

This is a fine guide, but my experience shows that not all ADC pins can be used when Wifi is started, for me only ADC0 worked ?

thanks Peter

[Reply](#)



**Sara Santos**

August 31, 2018 at 9:00 am

Hi Peter.

Thanks for sharing that info.

As I can remember we only had trouble in one specific project in using GPIO4 for analog reading with WiFi.

We should try each individual ADC pin with WiFi and see if we have any

trouble.

Thank you

Regards,

Sara 😊

[Reply](#)



**mtz8302**

December 2, 2018 at 7:13 am

Hi Sara,

I think you can only use A1.. while WiFi is used. It's a design problem of the ESP32 chip.

Greetings

Matthias

[Reply](#)



**Sara Santos**

December 3, 2018 at 4:51 pm

Hi.

Thank you for sharing that.

We definitely need to test that situation.

Thanks.

Regards 😊

[Reply](#)

**moataz**

March 16, 2021 at 10:54 pm

Hello,

could you share with us the results of your test about using ADC pins while using WIFI ?

**Waldemar**

August 31, 2018 at 12:04 pm

One more a real Rui-Tutorial !! Very, very useful for my projects.

Many thanks to Sara and Rui

[Reply](#)**Sara Santos**

August 31, 2018 at 4:18 pm

Hi.

Thank you so much for your support!

Regards,

Sara 😊

[Reply](#)

**Chris Parsons**

September 3, 2018 at 9:55 am

Bit confused! I have a NodeMCU-32S board, which pins are Analog and which are digital?? You say GPIO34-39 are input only – so I could use all the other pins (if setMode is set to OUTPUT) to turn on LED's etc?

What about the GPIO's you say to not use? GPIO6 – 11?

Chris

[Reply](#)**Renaud Fensie**

April 22, 2020 at 8:20 am

yes you can use any pin from 0-33 as outputs. you could also use gpio 6-11 but it would interfere with the spi communications between the esp's processor and its flash memory and it is thus a bad idea.

[Reply](#)**Jerry Ericsson**

September 12, 2018 at 8:27 pm

Thanks a billion. I purchased my first ESP32 the day I first read about it. I thought my knowledge of the 8266 should get me started. BIG MISTAKE. I

ordered 4 of them to begin with, and up until a few weeks ago, they have laid capture in a plastic parts box. At long last I found one project that might work with them, and learned how to get the Arduino IDE to write to them. Well my Frogger try ended up pretty bad, never could get it working right, however I did get it to display, so I guess that is something. That said, I have learned more in the last few minutes reading this page then I knew about this little sports car of devices. Again, thanks billions I shall go on reading the rest of your pages. I knew I had come on something good when I first found your pages a few years ago, and you keep the education going and going.

[Reply](#)



**Sara Santos**

September 13, 2018 at 9:05 am

Hi Jerry.

Thank you so much for your kind words!

I'm really glad you find our tutorials helpful.

If you want to learn more about the ESP32 you can take a look at our ESP32 projects repository (it's free):

– <https://randomnerdtutorials.com/projects-esp32/>

We also have a course about ESP32, take a look at the contents here:

– <https://randomnerdtutorials.com/learn-esp32-with-arduino-ide/>

Thank you and good luck with your projects! 😊

[Reply](#)



**Dali**

October 5, 2018 at 8:12 am

Thanks for the nice overview. I have a question about pin current ratings. You declare 12 and 6 mA in your article which can also be found at other web pages. However if I read the datasheet correctly ([https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32\\_datasheet\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32_datasheet_en.pdf)) it should be 40 and 28 mA. Where are your numbers from?

[Reply](#)



**Sara Santos**

October 10, 2018 at 11:20 am

Hi Dali.

You are right.

We've update the post.

Thanks for letting us know.

Regards,

Sara 😊

[Reply](#)



**Juan Pablo**

October 5, 2018 at 3:56 pm

Great work!.. Is it possible to powering esp32 by external supply? I mean, 5v to Vin Pin and Ground to ground. I've tried many ways but it is impossible to me. It just work with USB powering. Thanks in advance.

[Reply](#)

**Sara Santos**

October 8, 2018 at 11:26 am

Hi Pablo.

Yes, it is possible to power the ESP32 using external power supply.

What exactly are you using to power the ESP32?

Regards,

Sara 😊

[Reply](#)**Juan Pablo**

October 8, 2018 at 11:21 pm

Hi Sara. I'm using 5v power supply direct to V5 pin but doesn't work. It looks like it reboots and never runs normally, but if I connect to USB 1A it works fine. I've tried 3 boards and it doesn't work. It is Chinese, Wrover, most common, v4

[Reply](#)**Juan Pablo**

October 8, 2018 at 11:24 pm

It is like Aliexpress – DOIT ESP32 DEVKIT V1 Board (Wi-Fi and Bluetooth). Thanks for your kind response

[Reply](#)**Sara Santos**

October 9, 2018 at 1:49 pm

Hi Pablo.

That's the same ESP32 board that we usually work with.

That very weird.

It should power up by supplying 5V to the VIN pin. Make sure you're supplying enough current.

Otherwise, I have no idea why that is happening.

Regards,

Sara 😊

**Juan Pablo**

October 9, 2018 at 5:31 pm

Thanks!

**mtz8302**

December 2, 2018 at 7:05 am

might be a problem with the power supply:

solution Tantal capacitor 47uF between GND and VCC

see also:

[arduino-hannover.de/2018/07/25/die-tuecken-der-esp32-stromversorgung/](https://arduino-hannover.de/2018/07/25/die-tuecken-der-esp32-stromversorgung/)

[Reply](#)



**Paul Nguyen**

November 5, 2018 at 6:10 pm

Thanks, very usefull.

[Reply](#)



**bestware**

November 6, 2018 at 10:47 am

There is ONE I2C-interface only! Yes, 2 lines are used to provide one I2C-bus only.

Anyway, very useful overview!

[Reply](#)



**Sara Santos**

November 8, 2018 at 10:09 am

Hi.

Thank you for making that clear.

Regards,

Sara 😊

[Reply](#)



**Joao Lopes**

November 8, 2018 at 9:47 pm

Very good, congratulations !

[Reply](#)



**Sara Santos**

November 9, 2018 at 1:03 pm

Thank you 😊

[Reply](#)



**mohanraj52**

November 12, 2018 at 10:50 am

i want to use esp wroom32 module in my custom pcb,how to do it.can you please help me

[Reply](#)



**Sara Santos**

November 12, 2018 at 12:37 pm

Hi.

At the moment, we don't have any project/tutorial about that subject.

We have a project in which we build a shield for the ESP32 board. But I don't think it is what you're looking for:

<https://randomnerdtutorials.com/build-an-all-in-one-esp32-weather-station-shield/>

Regards,

Sara 😊

[Reply](#)



**alex**

November 23, 2018 at 11:43 pm

Hey, I need some kind of help...

I have a custom design with ESP32. My GPIO0 is held LOW, my EN pin is connected to 3.3v. When I boot, I get the "waiting for download" prompt. However, I always get the famous "Timed out waiting for packet header" error.

Any help is appreciated 😊

[Reply](#)



**Sara Santos**

November 24, 2018 at 5:27 pm

Hi Alex.

When uploading code to the ESP32, you should press the BOOT button, which is the same of held GPIO 0 to LOW. There are other pins that influence uploading code to the ESP32, such as GPIO2, GPIO12, and GPIO15.

Take a look at the following article:

[github.com/espressif/esptool/wiki/ESP32-Boot-Mode-Selection](https://github.com/espressif/esptool/wiki/ESP32-Boot-Mode-Selection)

It is very informative when it comes to boot mode selection.

I've never experimented with a custom design with only the ESP32 chip (I've just experimented with development boards with the chip), so I'm not aware of other stuff that you may need to pay attention to.

Anyway, I hope you found the article useful and you can solve your problem.

Regards,

Sara 😊

[Reply](#)



**deets**

June 23, 2019 at 9:44 am

Not sure if this is still relevant, but I encountered a similar issue in my custom PCB project with the ESP32-WROVER-B. Datasheet is

[espressif.com/sites/default/files/documentation/esp32-wrover-b\\_datasheet\\_en.pdf](https://espressif.com/sites/default/files/documentation/esp32-wrover-b_datasheet_en.pdf)

I hit two snags:

- using a strapping pin for my SPI display. Pay special attention to all pins mentioned in section 2.3 “Strapping Pins”. They all are trouble. So other than IO0 (for boot mode) I did not connect any of them
- missing pullups on #EN and IO0 and missing capacitor on #EN. The datasheet is a bit sneaky about these. The example schematic doesn’t mention the pullup for IO0 and the capacitance for #EN is 0.2uF (as there are two circuits mentioning that). I experimented and even use 0.3uF for my capacitor on #EN.

With this setup, I got the board to reliably flash using esptool.

My project can be found here: [github.com/deets/deets-fpv-vtx-scanner](https://github.com/deets/deets-fpv-vtx-scanner)

You can use KiCad 5 to see the schematic that I verified works for me.

[Reply](#)



AOT

December 22, 2018 at 10:47 am

After days of confusion between many articles and distinguishing the PINs name and functions, this article made my day. 😊  
Many thanks for your well written article.

[Reply](#)

**Sara Santos**

December 26, 2018 at 4:35 pm

Thank you! 😊

[Reply](#)**Dennis B. Hansen**

January 4, 2019 at 10:23 pm

@ bestware – I think you are mistaken.

There are indeed two separate I2C interfaces. These are referred to as I2C0 and I2C1.

See e.g. section 11.4 in the ESP32 Technical Reference Manual – where all the registers for control of the two interfaces are listed.

You can route the SDA and SCL pins of each I2C interface to any GPIO (which can be output and which is in the correct power domain). Please check the technical reference manual if you are interested in details.

But as Sara Santos correctly explains in the tutorial, the Arduino Wire Library allows one of the I2C's to be operated – and only on GPIO 21 and 22.

[Reply](#)**Rick009**

February 20, 2019 at 3:20 pm

Hi,

how can I connect ESP32 WROOM to 18650 (3.7 V) battery ?

Thanks

[Reply](#)



**Sara Santos**

February 20, 2019 at 5:16 pm

Hi Rick.

You can use a Low DropOut Regulator (LDO) to convert the 3.7V to 3.3V and power the ESP32 through the 3.3V pin.

You can use the [MCP1700-3320E LDO](#), for example.

I hope this helps.

Regards,

Sara

[Reply](#)



**Chris**

February 20, 2019 at 7:37 pm

You can also get an 18650 'shield' which has an LDO fitted and offers a 3v output? Cost me 2 UKP from eBay

[Reply](#)

**Sara Santos**

February 21, 2019 at 11:20 am

I'm not familiar with that "shield", but with what you've described, it seems a good fit.

[Reply](#)**Chris Parsons**

February 21, 2019 at 11:26 am

Here is a link to one?

<https://www.ebay.co.uk/itm/ESP32-18650-Battery-Shield-V3-ESP-32-Micro-USBWemos-LED-for-Arduino-Raspberry-Pi/292498985534?epid=24006249128&hash=item441a4c523e:rk:1:pf:0&checksum=292498985534af0ace93d18544359998ba8308878450&enc=AQADAAAC8FjVrDbVsZ8oH%2F8PNHtt9VX4%2Fw7FZcmMuqsX8uaFEduVaORNXMCeazS4LEKj6KZIHxIUwmX1v17N9DAobpUqdxVjZ98fBA3dYSriex3x9H0ZPD0NygfPp84ctvlzbJ%2BbzWtZG071XChWAyOm6s0TCSn27%2Bzt2yJn0frZtN3i9mn7V8q94X5tBKS9qorMMDmaAQzM1S18k7Mge13VoQhP88eYTF8y3jWwayxCR8L7f%2BcVGhJraUHz14JnFDDYJPoCLWAPPH40Fjlj3YAX7Z1nLrG%2Fzg6bvPtk8VJBMD2Ozz%2FrHbngt%2FNzoFFL5MYBPHfHk%2F3frz2CqDupRrQYiloFIHM5ruSLxEGE8TvyqxnbLoAOlkvbIe4FE3Bdbfc%2FgGHcQ4H3jCJAG1yLemonUsnNYCWuS%2FoqKUARurc4djDUabWTHG%2B9mvrbn%2F66S%2BnjIOv8ni%2Ba3Ik%2BwMy8pALql5%2BzKxYnqmyHSe%2BeG5nbeBLzu0jKoNFkiLuRxwsxNUEnrBLug8%2F1ZKZKI%2FPTRpv2QCvFLTGIEHCpECWO5V746H0W6m7%2BvQ5P8sio3qlpVDiOexu54CMg7eyLSQxXbdp3zwbX6wD2%2FpLqjrLC4GEIcyn65O1YhUmWuDeBZUJd3XuvW4jcYXuBwd7IMZkhHGjSHEAN8w8TMF%2FwDj%2BofDlu0R5I4UqTTpr1kvgVuRi5y5lv7>

%2B%2F1aTmlqosMeTQJUrr5jkQcdccwrqlRJwLQMmX2SI5Q%2BrjLh  
%2FjMM%2B6J7k4yilt%2ByL7oYtNsns76cjiX90MWO1B1WE98zFiUm3  
vtTMRZJ4HeGeqen%2FkROx0xzL%2FTQKiTUesXclxEhuKMJWwlal9  
R6rszHx%2BeaL4sjedyEevb1nro5QdYWA6yqJVIRAI%2FmeLHgylZp0c  
cPWbLpx%2FUzBw%2BqGb3Cq0sXWcFIQ%2BoQcangeThTD24FyA  
%2FMgD7UUJJerMpzz6gPq0SihxXSQ%2FhSpOG4hgeOtz12j3hUj%2  
FeQ5YU4rKAAz

**Sara Santos**

February 22, 2019 at 6:11 pm

Yes. That should work.

**Bob Lynas**

March 4, 2019 at 12:02 pm

You didnt mention about the UARTS apart from saying there are 3. I read somewhere that UART2 on GPIO 16/17 cant be used, or at least those pins cant be used, and I see no other mention of GPIO16/17 above. Do you know what the story is there ?

Thanks for a great write-up by the way, I come back to it often !

[Reply](#)**Sara Santos**

March 4, 2019 at 12:27 pm

Hi Bob.

As far as I know, you can use GPIO 17 and 16 as UARTs, as well as GPIO 1 and GPIO 3.

UART0: (GPIO 1 and GPIO3)

UART2: (GPIO 17 and GPIO 16)

UART1: (GPIO 9 and GPIO10) – these are connected to the ESP32 SPI flash memory, so you can't use them. But, you can use UART1 by defining other pins on the HardwareSerial. Andreas Spies has a great video explaining this: [youtu.be/GwShqW39jIE?t=182](https://youtu.be/GwShqW39jIE?t=182)

Thank you for your interest in our tutorials.

Regards,

Sara

[Reply](#)



**Jock Murphy**

May 1, 2021 at 10:02 pm

Interestingly I was just doing a small ESP32 Arduino project and I couldn't toggle GPIO 16 or 17 in my code. It took a while to realize what was going on, but I eventually moved to different pins. After that I did some digging I discovered that pins 16 and 17 are used by PSRAM on the WROVER modules.

Pin 16 is used as the chip select for PSRAM, and 17 is used for the clock signal. It would probably be good to note this above.

You can see it on the schematics on the WROVER datasheet:

[https://www.espressif.com/sites/default/files/documentation/esp32-wrover\\_datasheet\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/esp32-wrover_datasheet_en.pdf)

[Reply](#)**Sara Santos**

May 3, 2021 at 9:44 am

Hi.

Thanks for sharing that.

Can you share what your project does?

I've used GPIO 16 successfully with projects that involve taking a photo (I guess this doesn't use PSRAM). However, if trying to video stream in high-resolution settings, it might crash because it needs to use PRAM.

Regards,

Sara

[Reply](#)**Jock Murphy**

May 3, 2021 at 1:19 pm

I can give you the broad strokes. I was prototyping a industrial controller using a ESP32 using a ESP32 DevKitC — which uses a WROVER module and not a WROOM). I needed multiple serial ports, I2C, SPI, and assorted GPIO. Hence the conflict with 16&17

**Sara Santos**

May 3, 2021 at 4:54 pm

Hi again.

I thought you were referring to the ESP32-CAM, my mistake.

Regards,

Sara



**Jannen Siahaan**

March 21, 2019 at 1:32 am

Thank you very much

[Reply](#)



**Sara Santos**

March 21, 2019 at 10:53 am

You're welcome. 😊

[Reply](#)



**Kimanzi**

March 31, 2019 at 8:54 am

Randomnerdtutorials, you guys are great. Your articles are very very much helpful.

[Reply](#)**Sara Santos**

March 31, 2019 at 3:48 pm

Thank you.

[Reply](#)**Rusty**

April 4, 2019 at 8:26 am

GPIO 5, 12 (MTDI) and 15 (MTDO) are strapping pins, i.e. not OK to use during boot.

GPIO 12 must be low (0) and GPIO 5 and 15 high (1) during boot.

[Reply](#)**Tom**

April 17, 2019 at 7:37 pm

Thanks for the excellent overview.

Small remark: GPIO13 seems to be missing in the list of GPIOs

[Reply](#)**Sara Santos**

April 18, 2019 at 11:02 am

Hi Tom.

Thanks for noticing.

I've added GPIO 13 now.

Regards,

Sara

[Reply](#)**arun**

April 30, 2019 at 11:38 am

Hello all ,

RTC GPIOs... What is RTC is it REAL TIME CLOCK..or something else?

[Reply](#)**Rui Santos**

May 1, 2019 at 10:42 am

Yes, RTC stands for Real Time Clock in this case.

[Reply](#)

**arun**

May 1, 2019 at 11:13 am

hi

Can you give some explanation RTC in context with GPIO

I am not able to find any information.

Thanks

[Reply](#)**Rui Santos**

May 1, 2019 at 11:27 am

Basically, these GPIOs are routed to the RTC low-power subsystem can be used when the ESP32 is in deep sleep. You can use the RTC GPIOs to wake up the ESP32 from deep sleep ([read ESP32 External Wake Up](#)):  
<https://randomnerdtutorials.com/esp32-external-wake-up-deep-sleep/>

[Reply](#)**hamdani**

April 30, 2019 at 10:56 pm

thank you, this article is very useful

[Reply](#)

**Rui Santos**

May 1, 2019 at 10:40 am

You're welcome! Thanks for reading!

[Reply](#)**John**

May 7, 2019 at 11:14 am

Hi

Many thanks. Your article would have saved me hours if available earlier!!!

[Reply](#)**Nguyễn Minh Hiếu**

May 14, 2019 at 9:42 am

101/5000

I am having problems with analogRead (). It does not work.  
I need to read the value of the sensor. Thank you

[Reply](#)

**Rui Santos**

July 29, 2019 at 3:51 pm

Can you use a different GPIO? Any of the following GPIOs should be safe to use with Wi-Fi + AnalogRead():

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)

GPIO 15 (the one you're using belongs to ADC2 and any ADC2 pin causes some issues when used with Wi-Fi).

Try with GPIO 33 for example and let me know your results.

[Reply](#)**Logan Byrne**

May 17, 2019 at 7:48 pm

I have a ESPWROOM-32 module, but it has 38 pins. What pin layout do I use? Top 3.3V on left, GND on right and 5V bottom left and CLK on right. Putting the USB connect at the bottom. Help?

[Reply](#)

**Sara Santos**

May 19, 2019 at 9:33 am

Hi Logan.

To use the pinout, please check the labels on the board and then compare with the pinout that we have.

Usually, in different boards, the same labels refer to the same GPIO.

For example, the pin marked as RX2 on the silkscreen of the board is GPIO16. D2 is GPIO2 and so on.

I hope this helps.

Regards,

Sara

[Reply](#)**Valdryan Ivandito**

May 20, 2019 at 10:32 am

Hi, This is very good tutorial and it is so helpful.

Thankyou so much 😊

Regards

From Indonesian

[Reply](#)**Rui Santos**

July 29, 2019 at 3:48 pm

You're welcome! Thanks for reading

[Reply](#)



**Scott**

December 3, 2020 at 2:32 am

hi Rui and Sara,

Thanks for providing the great website. I wonder if you know how to flash a naked esp32 chip, not the module? I am working on a customer board which doesn't have USB port.

Cheers,

Scott

[Reply](#)



**Sara Santos**

December 3, 2020 at 6:50 pm

Hi.

Unfortunately, we've never tried it yet.

However, this can help you figure out what pins need to be high and low in order to enter boot mode:

<https://github.com/espressif/esptool/wiki/ESP32-Boot-Mode-Selection>

Regards,

Sara

[Reply](#)

**Scott**

December 3, 2020 at 10:33 pm

Thanks Sara. I am trying a method and will have a look at the reference you directed.

Cheers,  
Scott

**poornima**

May 21, 2019 at 4:24 pm

Super it is very useful. Thanks a lot.

[Reply](#)**Sara Santos**

May 22, 2019 at 11:16 am

Thank you 😊

[Reply](#)

**Ganesh**

May 26, 2019 at 1:26 pm

Hi !

It was a great description. Could please tell me which pins can't be used when using WIFI.

Thanks

[Reply](#)

**Sara Santos**

May 27, 2019 at 6:11 pm

Hi.

As mentioned in the article:

“ADC2 pins cannot be used when Wi-Fi is used. So, if you’re using Wi-Fi and you’re having trouble getting the value from an ADC2 GPIO, you may consider using an ADC1 GPIO instead, that should solve your problem.”

Regards,

Sara

[Reply](#)

**Jonas**

June 9, 2019 at 4:34 pm

I have exactly the board of this tutorial with 36 pins.

I connected GPIO16 and GPIO17 to a standard optocoupled relay board and when the asyncwebserver sends a web page to the client the relays drive crazy.

Avoid those pins.

[Reply](#)



Jay

July 5, 2019 at 12:11 am

hi, i have a DOIT ESP32 devkitV1(clone) –  
pinout seems to match the real doit devkitv1 😊

in some of the info i have read on the internet it has a onboard Temperature sensor,

yet i dont seem to be able to find examples that use it? or what specific sensor it is? – i have used the Hall sensor onboard fine.

is this temp sensor accessible, and worth the effort (will it just show me how much my board has warmed up, or will it give me usable environment data?)

ps — is there a way (software) to use the RED power led?? i can use the BLUE one, LED\_BUILTIN (its gpio2) but can i use the power led?

[Reply](#)



Umar

August 15, 2019 at 7:10 am

Hi, i was wondring, can i use other GPIO as CS pin for SPI

[Reply](#)



**Sara Santos**

August 24, 2019 at 10:57 am

Hi Umar.

Yes, you can use.

You just need to define that on your code.

Regards,

Sara

[Reply](#)



**Genny Esposito**

September 7, 2019 at 7:22 am

A really good guide, thanks

I have only one question, with the devkit (30 gpio), how can I take a manual RESET button outside? Thanks again 😊

[Reply](#)



**Sara Santos**

September 8, 2019 at 4:15 pm

Hi.

You can use the Enable pin.

Read the section about the enable pin in the post.

Regards,

Sara

[Reply](#)



**Neil**

September 8, 2019 at 11:18 pm

I wanted to take the time to thank you for these series of articles. I have a number of these generic AliExpress/Aokin ESP-WROOM-32 boards and while I was able to use them without much trouble, it always bothered me that I didn't have any "solid" documentation for them. Now I know exactly what I am dealing with.

Again, Thank You...

[Reply](#)



**Doug Elliott**

September 10, 2019 at 1:37 am

I went through a similar process figuring out the Adafruit Esp32 Huzzah Feather board, and created my own reference chart. If it's useful to anyone, help yourself:

[github.com/va3wam/TWIPi/blob/7e3c5c6ea300107d7c091d7392264c02d4c73f93/Eagle/doc/feather-pinout-map.pdf](https://github.com/va3wam/TWIPi/blob/7e3c5c6ea300107d7c091d7392264c02d4c73f93/Eagle/doc/feather-pinout-map.pdf)

[Reply](#)



**usrrrs**

September 13, 2019 at 5:04 am

Very useful, before creating any new project i always keep this page in front of me. Thanks

Update GPIO15 function for Enabling/Disabling Debugging Log Print over U0TXD During Booting very useful for final deliverable product

[Reply](#)



**James Kincaid**

January 7, 2020 at 3:52 pm

This particular article is for the 36 pin kit... Do you have the same breakdown of the 30 pin kit?

[Reply](#)



**Sara Santos**

January 8, 2020 at 10:19 am

Hi. The pinout is basically the same, but doesn't include GPIOs 6 to 11.  
Regards,  
Sara

[Reply](#)



**Eva**

February 3, 2020 at 9:06 am

Hello can u help me?  
i'm using esp32 for my project and i using pressure sensor .  
the output of the sensor, i using GPIO36 but it doesn't work..  
Thank You

[Reply](#)



**Sara Santos**

February 3, 2020 at 10:50 am

Hi Eva. What is the sensor that you're using?  
Regards,  
Sara

[Reply](#)

**Eva**

February 17, 2020 at 8:50 am

i'm using MPX5700AP

[Reply](#)**David**

February 29, 2020 at 2:34 pm

Thank You for this excellent resource! Random Nerds is definitely helping to educate and expand this community's capabilities. 😊

[Reply](#)**Sara Santos**

March 2, 2020 at 11:29 am

Hi David.

Thank you so much for your feedback.

Regards,

Sara

[Reply](#)

**Emmanuel HERRMANN**

March 10, 2020 at 7:24 pm

Very useful resource.

Many thanks from France.

[Reply](#)**Pekka Lehtikoski**

March 26, 2020 at 6:12 pm

This helped a lot, thank you.

[Reply](#)**K. Radhi**

March 29, 2020 at 9:10 am

Very informative article indeed, thank you.

I am planning to do my own home automation project using ESP32-WROOM-32U module.

I simply don't need all the IO pins, and am planning to breakout only the pins I need to minimize the PCB footprint (IC2 and IO 16-27).

I used a 'Test board Burn Fixture Programmer' to upload the sketch, and all

future updates should be OTA updates. Hence, I don't need to mess about with the boot mode once the module is soldered to the PCB.

The question is, do I have to connect any of the Strapping Pins to anything if I want a normal boot and occasional OTA updates?

[Reply](#)



**Sara Santos**

March 29, 2020 at 4:40 pm

Hi.

Thank you for your comment.

I think you get a clear answer if you read this guide:

<https://github.com/espressif/esptool/wiki/ESP32-Boot-Mode-Selection>

I hope this helps.

Regards,

Sara

[Reply](#)



**Khalil**

April 1, 2020 at 8:46 pm

Thanks for your help Sara.

I have read the link you provided, and found it very informative, but could not find a direct answer to my question.

Through trial and error, I found that providing 3.3V and GND did not work, but by using a pullup resistor on pin 3 (EN pin), the ESP started normally and I could flash it over the air without having to press anything or short

any pins. Later I found this information already provided in the datasheet on page 15.

Best regards,  
Khalil

[Reply](#)



**Sara Santos**

April 3, 2020 at 10:34 am

Hi.  
Thanks for sharing it. I wasn't aware of that.  
Regards,  
Sara

[Reply](#)



**Itaief**

May 13, 2021 at 4:10 pm

I need libraries Esp32 devkit v1 36pin in fritzing

[Reply](#)



**eduardo**

April 9, 2020 at 8:04 pm

hello thank you for the article, you what pins do I have to use to connect the ultrasonic sensor?

[Reply](#)



**Sara Santos**

April 10, 2020 at 11:08 am

Hi.

With the ESP32, you can use any GPIO that can be input for the TRIG pin and any GPIO that can be output for the ECO pin.

For example, you can use GPIO 26 and 27.

Regards,

Sara

[Reply](#)



**Ahmed**

May 29, 2020 at 6:42 pm

Thank you very much for this informative article ...but what about the libraries that works with arduino, are all compatible with the ESP32.

Best regards,

[Reply](#)

**Sara Santos**

May 30, 2020 at 9:00 am

Hi.

It depends on the library.

Many libraries are compatible.

Regards,

Sara

[Reply](#)**Dima Bagino**

July 14, 2020 at 12:08 pm

Hi

In DOIT dev kit GPIO21 and GPIO22 are used by I2C bus. Why those pins are not mentioned as I2C interface in the bare module, in the case if I decide to develop my own custom board.

[Reply](#)**Sara Santos**

July 14, 2020 at 10:15 pm

Hi.

In fact, with the ESP32 you can choose almost any pin to act as I2C. However, when using Arduino IDE libraries, it chooses GPIO 21 and

GPIO22 as default I2C pins.

Regards,  
Sara

[Reply](#)



**Ion**

September 13, 2020 at 3:16 pm

Hello,

I have a project developed on TTGO T7 Ver 1.0

It is a simple keypad 3X4 reader, plus and RFID decoder (get RFID data from TX of RDM6300 and extract the ID of the card) Match the keyboard input with the one store on EEPROM, match the RFID with the stored on EEPROM and if the match send an encrypted signal (like Go) through Bluetooth to a slave identical TTGO chip.

I wonder if the project can work if i try to use a regular ESP32 NodeMCU.

TTGO chips are only in China and very hard to get through Ali. ESP32 is easier to buy , even from Amazon. This is my problem.

[Reply](#)



**Nuno**

October 3, 2020 at 11:53 am

Hi,

Does the VIN pin also work as a 5V output when ESP board is powered from USB? Mine doesn't but I've seen you do it on your tutorials...

[Reply](#)**Sara Santos**

October 4, 2020 at 4:47 pm

Hi.

Yes, the 5V pin can be an output when the board is powered from USB.

Regards,

Sara

[Reply](#)**Nuno**

October 5, 2020 at 11:41 am

Hi Sara

Thanks for the quick reply. Problem turned out to be a mislabeled silkscreen on my board, marking as ground the CMD pin.

[Reply](#)**Wally**

October 26, 2020 at 3:23 pm

Greatly useful.

You guys are the best!

[Reply](#)**Bob**

October 30, 2020 at 2:36 pm

Lots of great info on this page! Thank you.

[Reply](#)**James Demello**

November 18, 2020 at 11:05 am

Thank you, very useful!

I migrated my latest project (self balancing gyroscopic two wheeler) from Arduino Uno to ESP32 because the processor is just too slow. Loop() runs in less than a millisecond. Wow.

[Reply](#)**Delanghe**

December 16, 2020 at 2:54 pm

ESP32 PICO kit V4 require manual RESET after powering off.

After powering ON the device, the software doesn't boot up automatically. I have to press the reset button. This means that the PICO device is

completely unusable as Alarm Device, due to a failure in automatic restart after power break down! Googling for solutions bring me up to install a 4.7 k resistance between +3.3V and IO0 pin. This works only if the powerOFF take place for severals seonds!

Is there an other solution to activate the automatically StartUp after a power failure ?

[Reply](#)



**Petr Sedlák**

December 21, 2020 at 10:19 am

Hello Sara,

thank you very much for this realy useful reference guide for the ESP32 GPIOs .

I have one DS18B20 conncted to the ESP32 GPIO 14 and sometime, the ESP32 has a problem to conect to the WiFi.

Could the problem be in the badly selected GPIO, as mentioned above?

Best regards

Petr

[Reply](#)



**Knoch1**

January 3, 2021 at 12:48 pm

Hi,

I'm a bit confused about the "Input only" pins as well. I did a board and used one of them (IO34) as an output, didn't work of course.

The original ESP32 WROOM Pinout picture (also on top of this article), doesn't mark them as "input only".

they have a PWM "snake line", doesn't make sense for an input only.

GPIO stands for General Purpose Input OUTPUT... how can this be Input only???

Thanks

Knoch1

[Reply](#)



Alex Padgett

January 11, 2021 at 12:08 am

Very helpful.

1) Any chance for an explanation about the UART pins or include this link to this webpage: [<https://circuits4you.com/2018/12/31/esp32-hardware-serial2-example/>]

2) It would also be super helpful to have an example of UART being used, maybe using it to show two ESP32s communicating via UART0 and UART2.

3) Some explanation on SoftwareSerial or a link to this page: [<https://www.arduino.cc/en/Reference/softwareSerial>] and a URL to as an example [<https://randomnerdtutorials.com/sim900-gsm-gprs-shield-arduino/#more-42170>]. I think SoftwareSerial is no longer being included in

IDE because I just installed latest clean version of Arduino IDE 1.8.13 Windows and I get an error 'include SoftwareSerial.h not found' (or I am doing something wrong as something isn't making sense to me!) when I use code like yours above.

Sorry if any of these things have already been done, I tried your website search for “UART2” and get no results.

Many thanks,  
Alex

[Reply](#)



**Matt**

January 15, 2021 at 12:59 pm

Any idea why float batteryLevel = map(analogRead(12), 0.0f, 4095.0f, 0, 100); seems to give back 100 no matter how bad the battery is?

[Reply](#)



**Wagner**

February 13, 2021 at 10:22 pm

Hi Sara and Rui,

I'm new to ESP32, and found your posts extremely helpful. Thanks.

I'm preparing a project to be used with Home Assistant, but I have three questions regarding the pinout:

1- You mention that if my project uses WiFi (which is the case) I should not use ADC2. So, is WiFi killing 10 of my GPIO? I'm a little confused, since many ADC2 are green on your table. If I exclude all ADC2 and all non-recommended pins, I would end up with just two “touch” pins – 32 and 33. Is that so?

2- If I load the ESP32 with 2 touch pads, 4 sensors (thermistor, LDR, humidity, PIR) on GPIO 33, 32, 35, 34, 39, 36 and 10 pulse keys on GPIO 23,

22, 1, 21, 19, 18, 5, 17 and 16 do you think it would work?

3- Do you have similar pinout recommendations of the ESP32-S2?

Thanks in advance.

[Reply](#)



**Sara Santos**

February 14, 2021 at 9:46 pm

Hi.

1- What we mean is that you can't use ADC in the ADC2 pins when using wi-fi. You can use those pins as standard input or output or touch pins, and they will work fine. They won't work properly as ADC pins.

2- I think those pins will work, but it is always better to test everything before making any permanent connections.

3- At the moment, we don't have any articles for the ESP32-S2.

I hope this helps.

Regards,

Sara

[Reply](#)



**Larry**

February 27, 2021 at 6:40 pm

So I tried the tutorial and sent you a message about things that didn't work but never got an answer. Basically one issue was in the code const char had to be changed to char, also no matching function for call to HttpClient http

[Reply](#)

**Sara Santos**

February 27, 2021 at 6:48 pm

Hi Larry.

Where did you post your question?

Since you are one of our customers, you can post your issues in the RNTLAB forum.

It is easier for me to follow your issues there:<https://rntlab.com/forum/>

Regards,

Sara

[Reply](#)**Sammy**

April 14, 2021 at 7:05 pm

What is the meaning of ADC(PA) for gpio36 and 39.

ADC is clear (Analog-Digital-Converter)

But what stays PA for?

[Reply](#)**don clay**

May 26, 2021 at 4:14 pm

Are there any of the GPIO pins that can be read as a nibble or a byte for the purpose of reading jumpers or a switch? Thanks.

[Reply](#)



**David Lehrian**

June 5, 2021 at 2:18 am

Hey Sara,

Great write up and very helpful reference page, I have it bookmarked. One thing I want to mention is that you list the absolute maximum per GPIO current draw as 40mA and I'm not so sure this is exactly correct. I see where you are getting this value from, the "output drive strength" on page 43 of [https://www.espressif.com/sites/default/files/documentation/esp32\\_datasheet\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf) (and page 53 lists the configurable values), but after doing a lot of searching for this value I came across the following thread which claims the maximum current draw is 12mA per GPIO and goes on to explain what "output drive strength" means. I found it enlightening.

<https://esp32.com/viewtopic.php?p=9557#p9557>

I don't know where he gets the number 12mA per GPIO from as I myself don't see any reference in any data sheet as to the maximum current draw per GPIO. The only thing I see is a cumulative current draw of 1200mA on page 42 of the above referenced datasheet. So at this point I'm left taking his word for it and erring on the side of caution. And if this is a maximum value, I would guess that running at 60% of this value would be about where you would want to be for stability and longevity.

[Reply](#)

## Leave a Comment

Name \*

Email \*

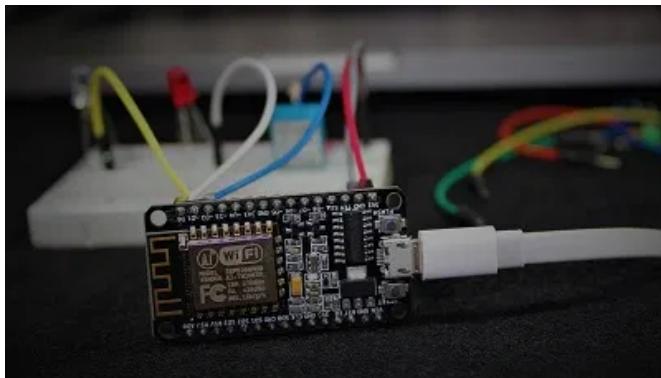
Website

- Notify me of follow-up comments by email.
- Notify me of new posts by email.

Post Comment

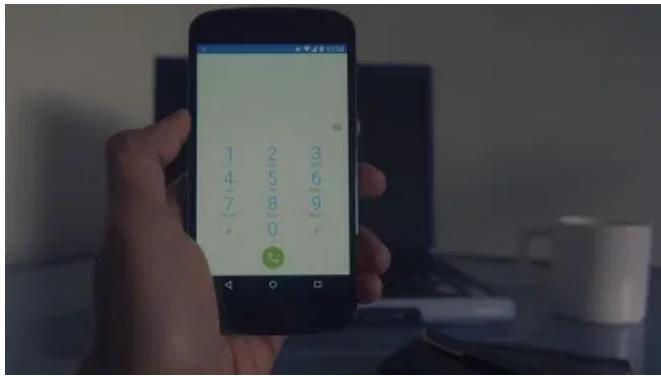


[Visit Maker Advisor – Tools and Gear for makers, hobbyists and DIYers »](#)



## [Home Automation using ESP8266 eBook](#)

[and video course »](#) Build IoT and home automation projects.



## [Build a Home Automation System from Scratch »](#)

With Raspberry Pi, ESP8266, Arduino, and Node-RED.

[About](#)   [Support](#)   [Terms and Conditions](#)   [Privacy Policy](#)   [Refunds](#)   [Complaints' Book](#)

[MakerAdvisor.com](#)   [Join the Lab](#)

Copyright © 2013-2021 · RandomNerdTutorials.com · All Rights Reserved