

August 21, 2020

## The ESP8266 - working with an ESP-12 module

This is **not** the best way to get started with the ESP8266. You would be better served to purchase a NodeMCU board (and ditch Lua immediately). The cons of the ESP-12 are:

- Pins on 2mm spacing (not 0.1 inch == 2.54 mm)
- Requires 3.3 volt power
- No USB to serial chip provided
- No buttons.

If you buy a NodeMCU board of some sort, you get all of the above, so you can just plug it in using a USB cable and get to work.

On the other hand, there can be some projects where a bare ESP-12 module is exactly what you want, in particular a battery powered project. The advantages are:

- Runs off 3.3 volt power, no power consuming LEDs.
- Tiny size -- 0.65 by 0.95 inches.

## What ESP-12 do you have?

I now take care to buy an ESP-12E or ESP-12F, which has the extra pins along the top edge. The "F" is said to have a better antenna, so that would be the top choice. I have never needed those extra pins, but it is better to have them and not need them. That aside, most of these pins are internally connected to a flash memory chip and you can't use them anyway, so who cares. See below. I have one or two "plain vanilla" ESP-12 units that only have the side pins.

## The Diagram Above

The nice ESP-12 diagram above is one of Alberto Piganti's excellent graphics for device pinouts. More are discussed in this article:

- [ABC basic connections](#)
- [Instructables: ESP 12 getting started](#)

## How to do it

- [ESP-12E manual \(pdf, 18 pages, includes schematic\)](#)

First you will need to provide 3.3 volt power. There are any number of regulators to take 5 volts down to regulated 3.3 volts. I bought a bunch of tiny little circuit boards on AliExpress that do this.

Second you will need some kind of USB to serial chip. There are a myriad of possibilities. I am using an Adafruit FTDI friend. Be sure whatever you use is in 3.3 volt mode.

If you want to work on a breadboard, you will need some kind of transition board to take the signals from the 2mm spacing to the 0.1 inch spacing. These are readily available from the sellers in China, and you may want to just purchase your ESP-12 with the transition board included.

For basic work (like trying a blink demo) you need to connect at least the following pins:

- Tx (TxD0/IO1)- to the FTDI or other USB/serial gadget
- Rx (RxD0/IO3)- to the FTDI or other USB/serial gadget
- Ground - to power and the FTDI
- Vcc - to 3.3 volt power
- Reset - I use a jumper wire to tap against ground
- GPIO0 - needs to be pulled low to run the boot loader
- GPIO2 - needs to be pulled high
- GPIO15 - needs to be pulled low
- CH\_PD (EN) - needs to be pulled up for the device to run.

## Boot loader

I use a jumper wire to connect GPIO-0 to ground, hit reset and I am running the bootloader. Once the bootloader starts running, you can release GPIO-0. Or you can leave GPIO0 grounded during the whole flashing process and then remove it and hit reset to run your application. If you want to be civilized, you might provide a 2 pin jumper that could be installed to ground this pin for flashing.

Of course the bootloader runs every time you hit reset, but it looks at GPIO0 on startup (along with GPIO2 and GPIO15). If it is low, it goes into boot mode. Otherwise it jumps to your application code. Note that this means that you are wise to avoid all three of these pins in your application.

So what I do with my ESP-12 is to connect a jumper wire from GPIO-0 to ground. The other pins are OK left alone (in my case anyway). Then I tap my reset wire against ground briefly and voila -- I am running the boot loader. I confirm this using:

```
esptool -p /dev/ttyUSB1 read_mac
Connecting...
MAC: 18:fe:34:07:85:99
```

```
esptool -p /dev/ttyUSB1 flash_id
Connecting...
Manufacturer: e0
Device: 4016
```

For some reason reading the flash\_id takes me out of the bootloader and starts running my application.

## Reset and low power sleep

On some of my projects, for increased battery life, I use a low power sleep mode and the timer to have the device wake itself up every minute or so. For this to work, you have to make a connection from IO16 (wake) to the reset pin (RST). You also have to remove this connection to flash new code. Another header and jumper might be in order.

## LED

There is an on-board LED connected to GPIO2 that you can use for a blink demo. This apparently has no conflict with the state of GPIO2 to launch the bootloader.

## Variants

The plain vanilla ESP-12 has only 16 pins, 8 on each side. The ESP-12E adds 6 more on the bottom edge (for 22 total). The ESP-12F has the same pinout as the "E", but an improved antenna.

## Additional notes

- CH\_PD (aka EN) is a signal that can be used to "power down" the unit in an application that wants to do this to save power. I just ignore it and the part runs just fine without anything connected to this pin.
- GPIO16 is special. You get 16 ordinary GPIO, and then this "extra" one. More elsewhere on this.
- TxD and RxD are also GPIO1 and GPIO3, but you will be hard put to use these other than to connect to a USB to serial device for programming the part.

## What happened to those 16 GPIO pins they told me about?

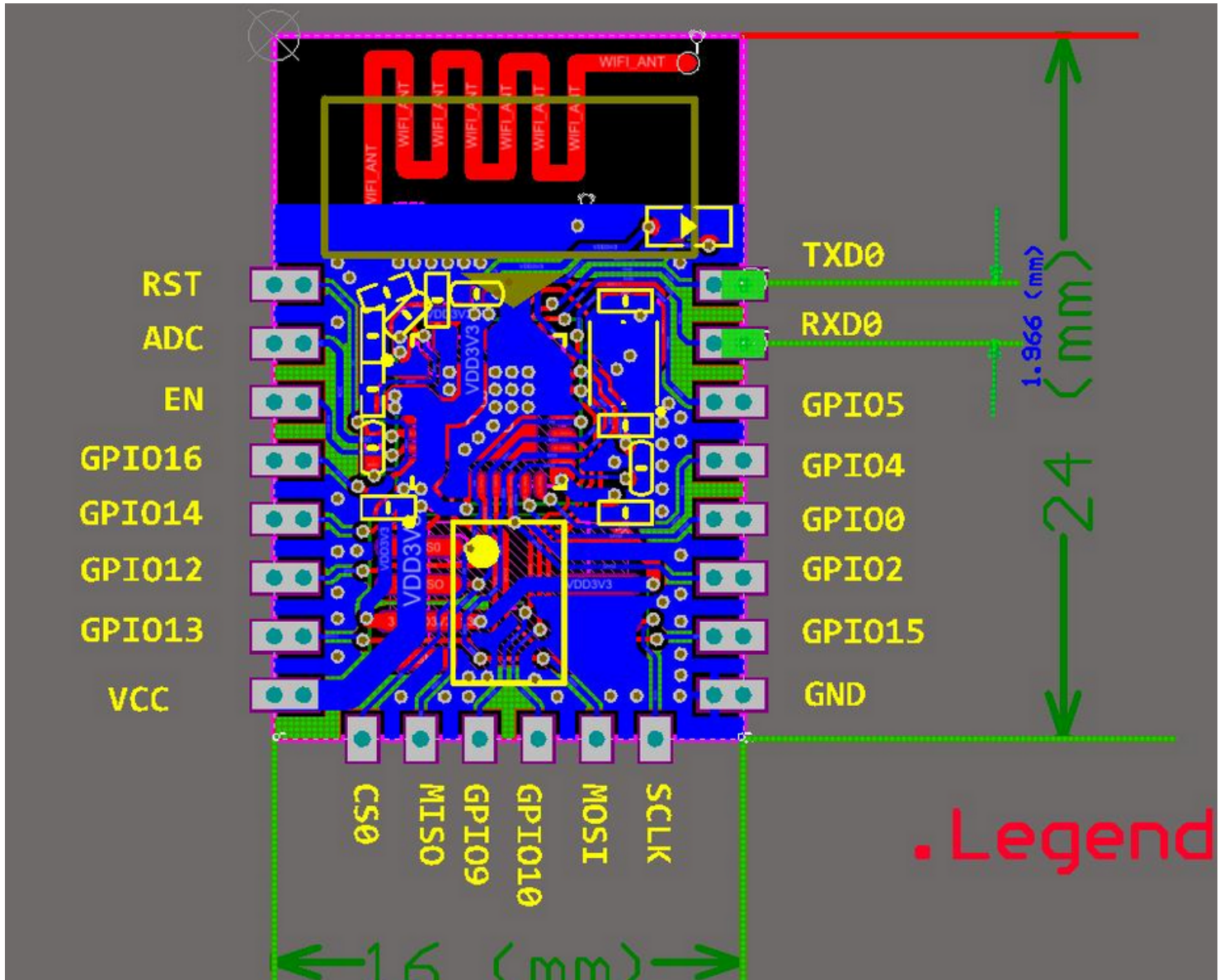
A good number of them get used being connected to the flash memory chip. Six of them in fact in a configuration that has a 4 bit data path to flash. More on this elsewhere. Another two pins for Tx and Rx and you are left with only 8. And there they are along the sides of the ESP-12 !!! But don't forget that 0, 2, and 15 set boot loader modes on reset!

Don't get too excited about those "extra" pins on the bottom edge of the ESP-12E. They are connected to the flash chip under the can, and it is hard to imagine how you will use them for any external purpose.

So why route them to the card edge? Beats me.

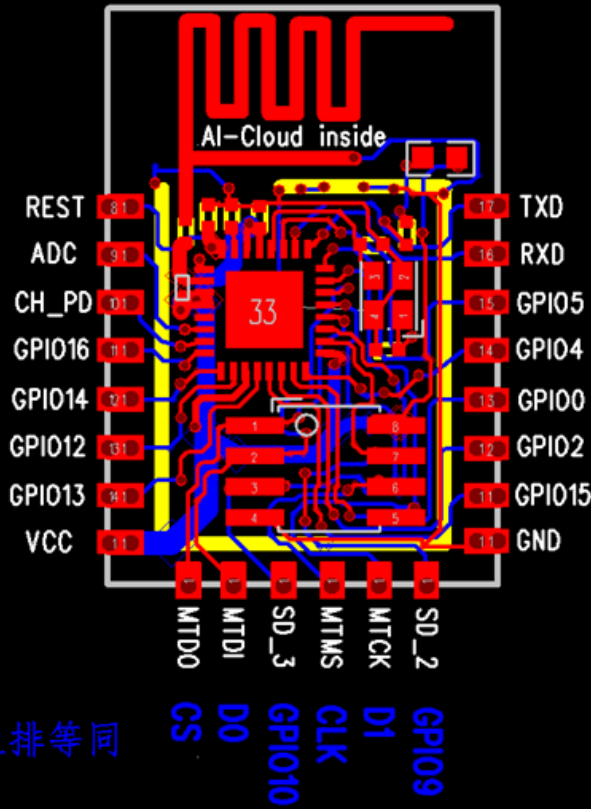
## Diagrams

First here are two diagrams provided just to show the pinout for one of these. Note that one of the old ESP-12 will not have the pins on the bottom. These diagrams show the unit viewed from the top.



Everyone seems to have their own ideas about how to label the pins on the bottom.

# ES-12E



注:此排脚位和上排等同

And here is the schematic for the ESP-12F. Note that the ESP-12E and ESP-12F are said to be identical except for the antenna (which is better on the ESP-12F).

