

## 太极创客

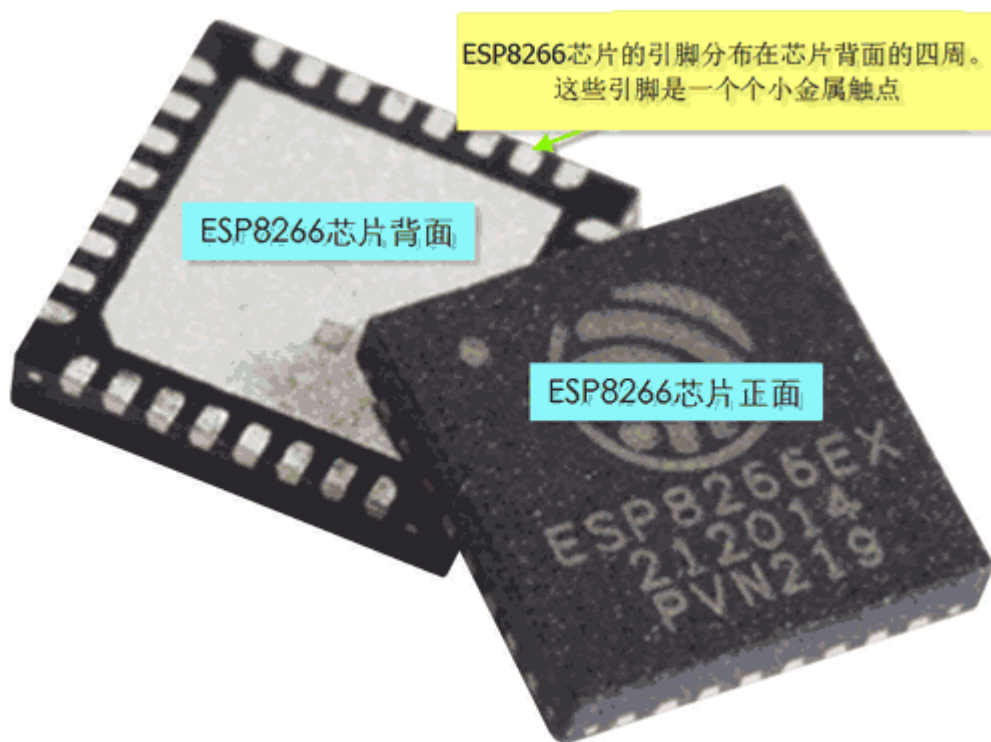
Arduino，人工智能，物联网的应用、开发和学习资料

# NodeMCU开发板详解 (第1章-第2节)

位置导航: [首页](#) / [物联网教程](#) / [第1章 – 开发板详解](#) / 本页

### 数字输入输出引脚（GPIO）

如下图所示，ESP8266芯片四周分布很多引脚。这些引脚大部分可用作输入输出使用。这些用作输入输出的引脚统称为GPIO。



esp8266芯片引脚位置

ESP8266芯片的GPIO引脚可用作数字输入来读取引脚电平，也可用作数字输出向外围电路发出控制信号。在这一点上，GPIO引脚与Arduino开发板的引脚功能十分类似。（请注意，我在这里用的是类似而不是相同。因为GPIO引脚与Arduino开发板引脚还是存在着很多不同的。）

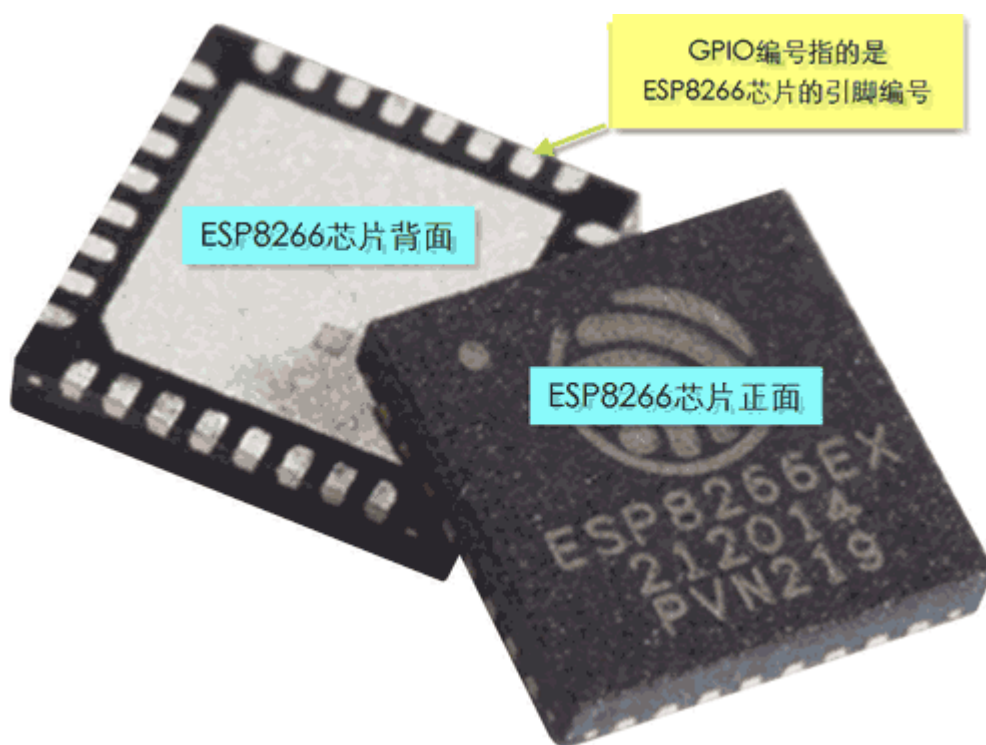
当引脚以数字输出模式工作时，低电平是0V (灌电流)，高电平是3.3V (拉电流)。

请注意:ESP8266芯片与Arduino Uno/Mega/Nano等开发板的引脚电平电压有所区别。  
Arduino开发板的高电平是+5V，低电平是0V。

## GPIO编号与NodeMCU开发板引脚名的区别

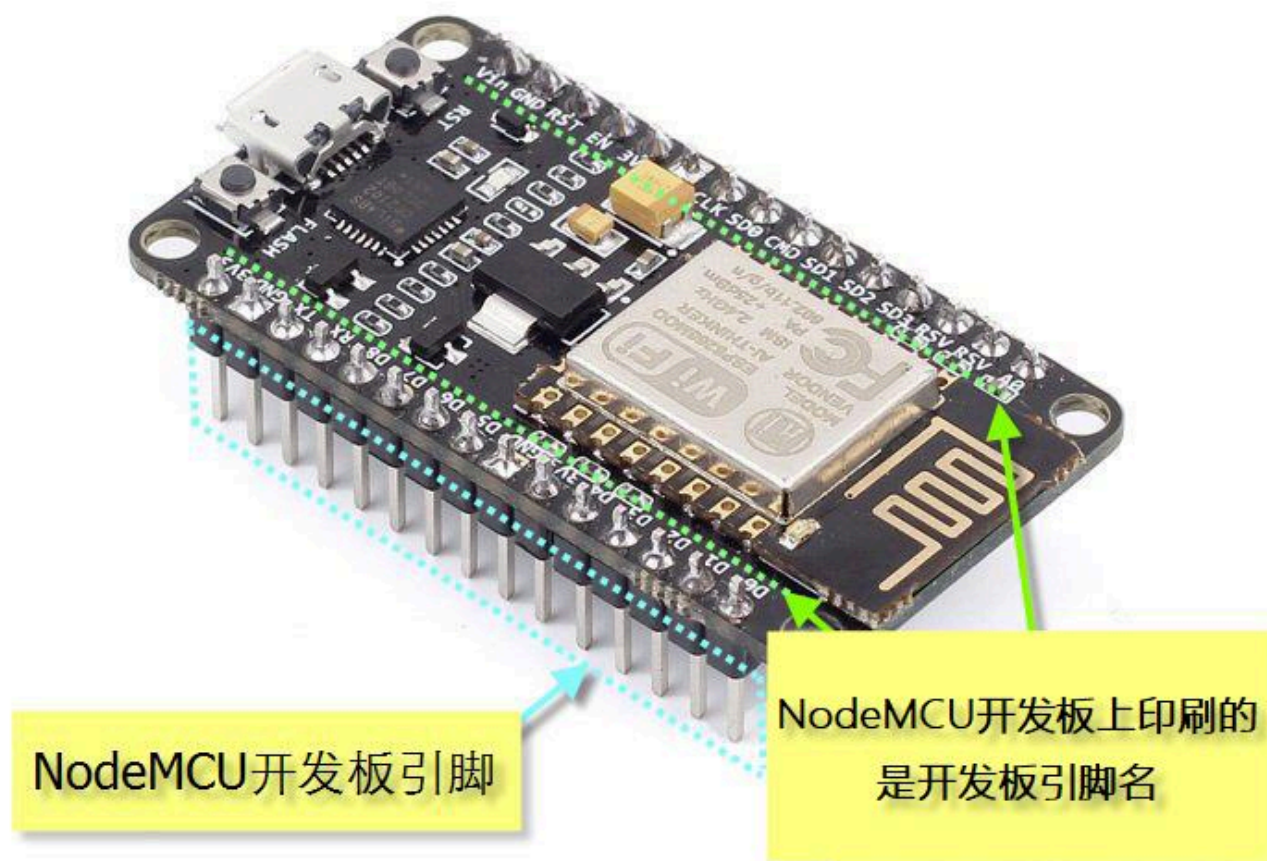
请注意：在很多介绍ESP8266以及NodeMCU的资料里会出现两种引脚命名方法。一种是GPIO编号，一种是NodeMCU引脚名。请注意这两者是不同的，请千万不要混淆。

GPIO编号指的是ESP8266芯片的引脚编号如下图所示：



GPIO编号与ESP8266芯片引脚编号区别

而NodeMCU引脚名指的是电路板上印刷的引脚名称。如下图所示



nodemcu引脚名称

本教程中“GPIO编号”这一名称指代的是ESP8266芯片的引脚编号。我们会以“GPIO + 数字”这一格式来指代，如GPIO2, GPIO16...

NodeMCU开发板的引脚名指的是开发板上印刷的文字。通常这些引脚名都是一个字符接一个数字的组合，如：D0、D1、A0... 我们会用“开发板引脚”或直接使用“引脚”来表示它们。

简而言之，只要您看到GPIO这几个字母，就说明是芯片引脚，而没有GPIO这几个字母，那肯定是指开发板引脚。

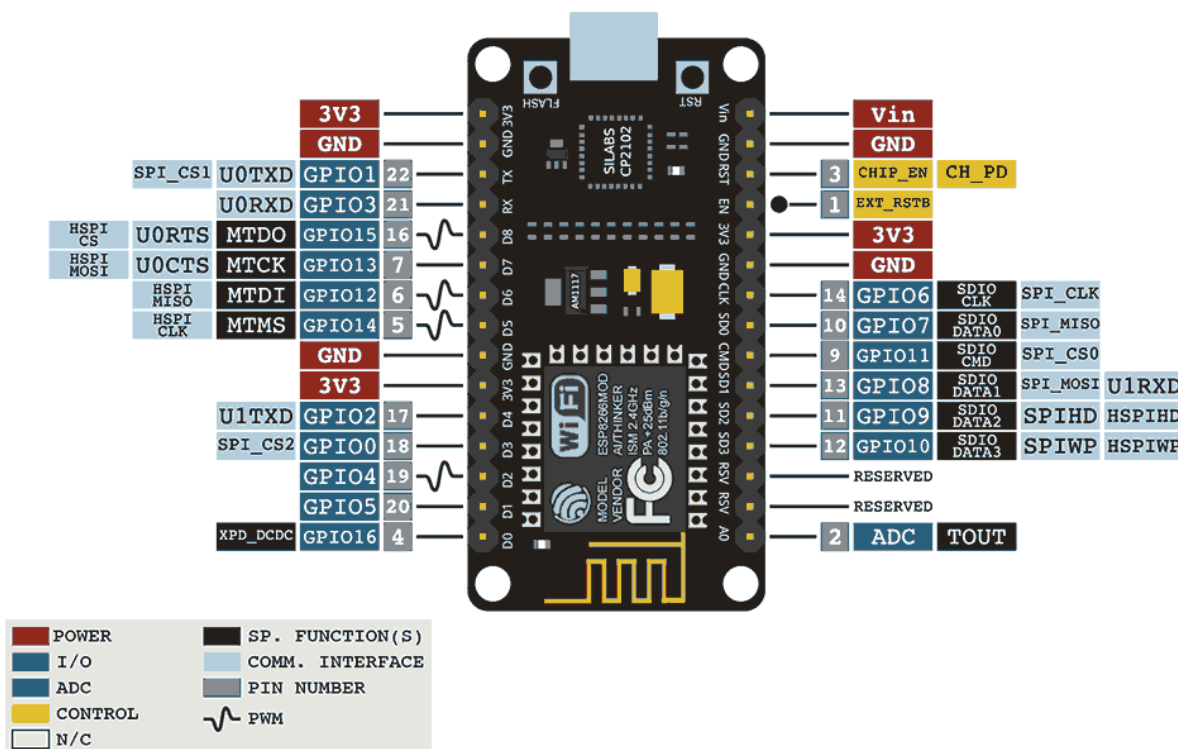
## ESP8266 GPIO编号与NodeMCU开发板引脚名的对应关系

在本节教程开始的地方我曾经跟大家讲过，ESP8266芯片的GPIO与开发板的引脚是连在一起的。但是GPIO与开发板引脚又是如何相互连接的呢？这个问题的答案就在下面的说明图中。

## ESP-12E DEVELOPMENT BOARD PINOUT

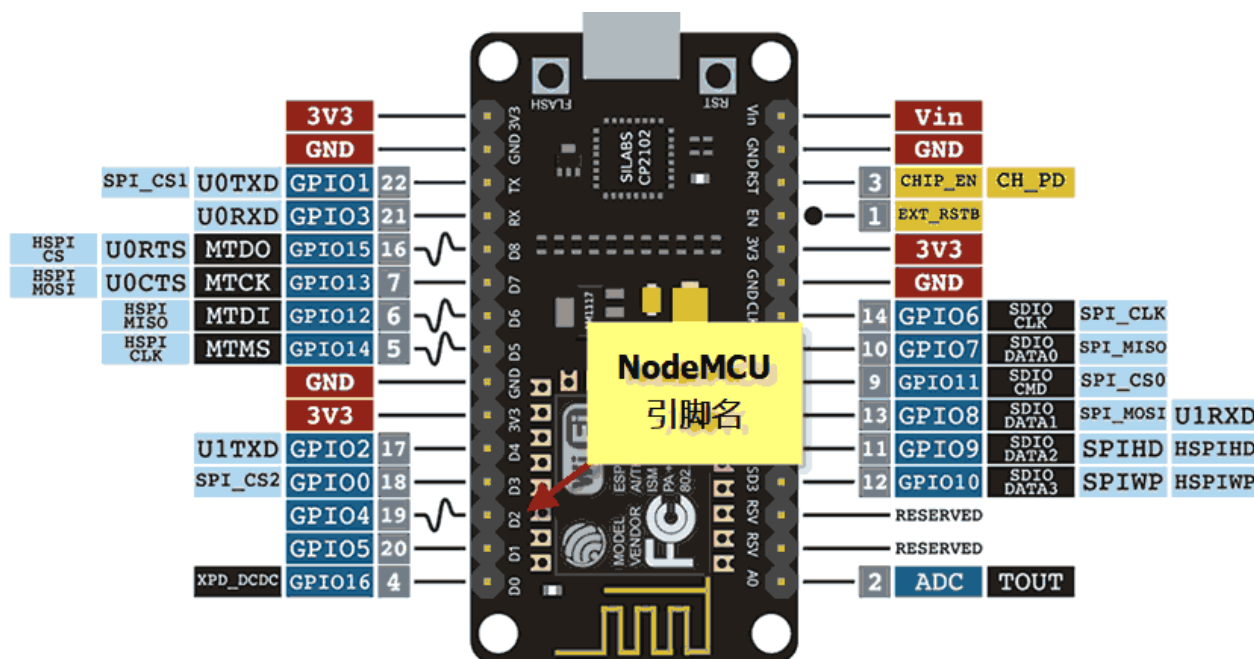
### NOTES:

- ▲ Typ. pin current 6mA (Max. 12mA)
- ▲ For sleep mode, connect GPIO16 and EXT\_RSTB. On wakeup, GPIO16 will output LOW for system reset.
- ▲ On boot/reset/wakeup, keep GPIO15 LOW and GPIO2 HIGH.



ESP8266-NodeMCU引脚功能

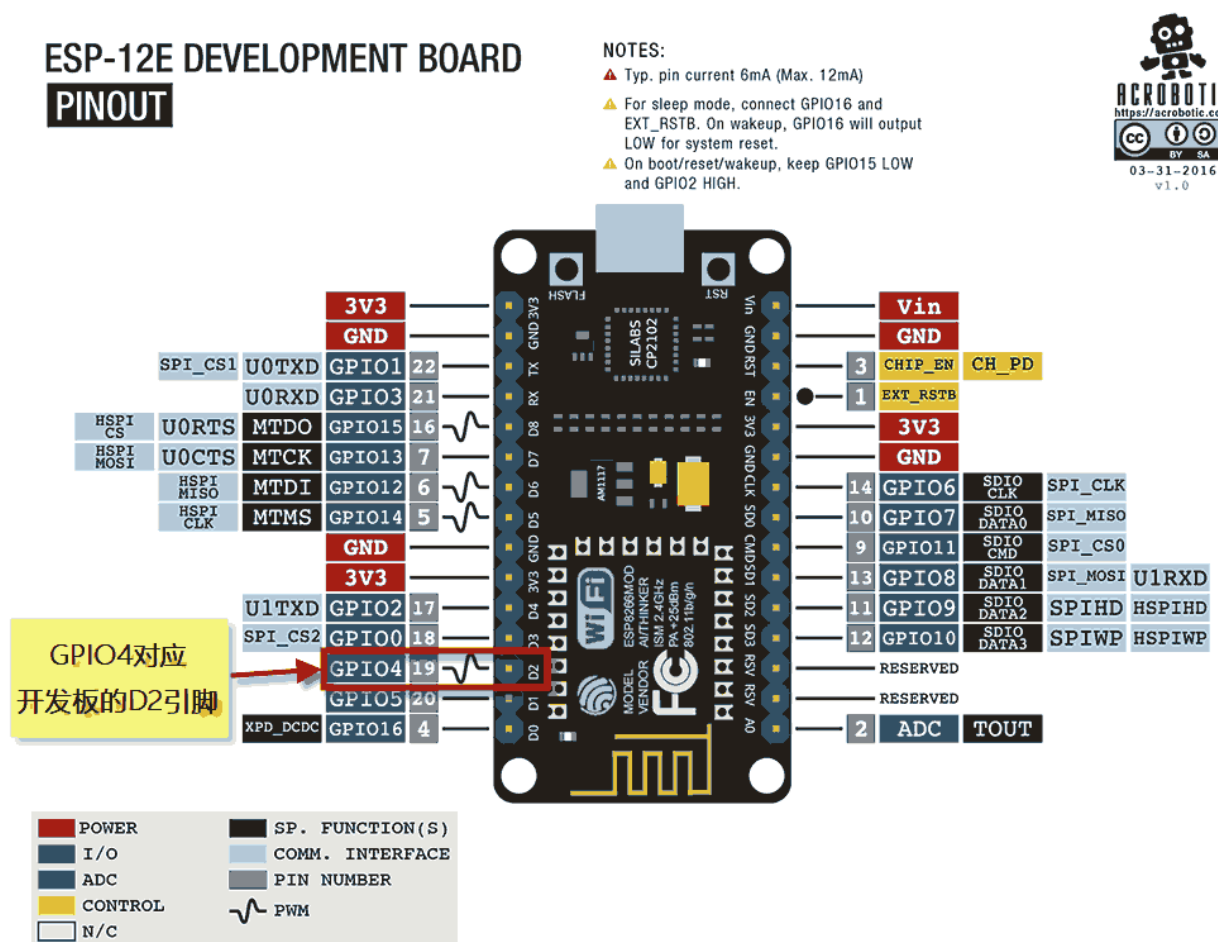
在以上图片中深蓝底白色字的标识就是GPIO引脚编号。如“GPIO4”。而开发板上所印刷的D2,D3等等就是NodeMCU开发板引脚名称。如下图所示。



NodeMCU的GPIO名



如下图所示，NodeMCU的D2引脚是与GPIO4相互连接的。以此类推，通过以下说明图您就可以找到D2与GPIO的对应关系了。



那么，我们为什么要花这么多时间讲解开发板引脚名和GPIO号的对应关系呢？

将来我们在编写NodeMCU开发板的控制程序时，经常会进行引脚操作。如以下程序语句。

```
digitalWrite(4, HIGH);
```

以上语句通过digitalWrite函数将引脚4设置为高电平。那么您知道这个数字4到底是指GPIO4还是指开发板D4引脚呢？

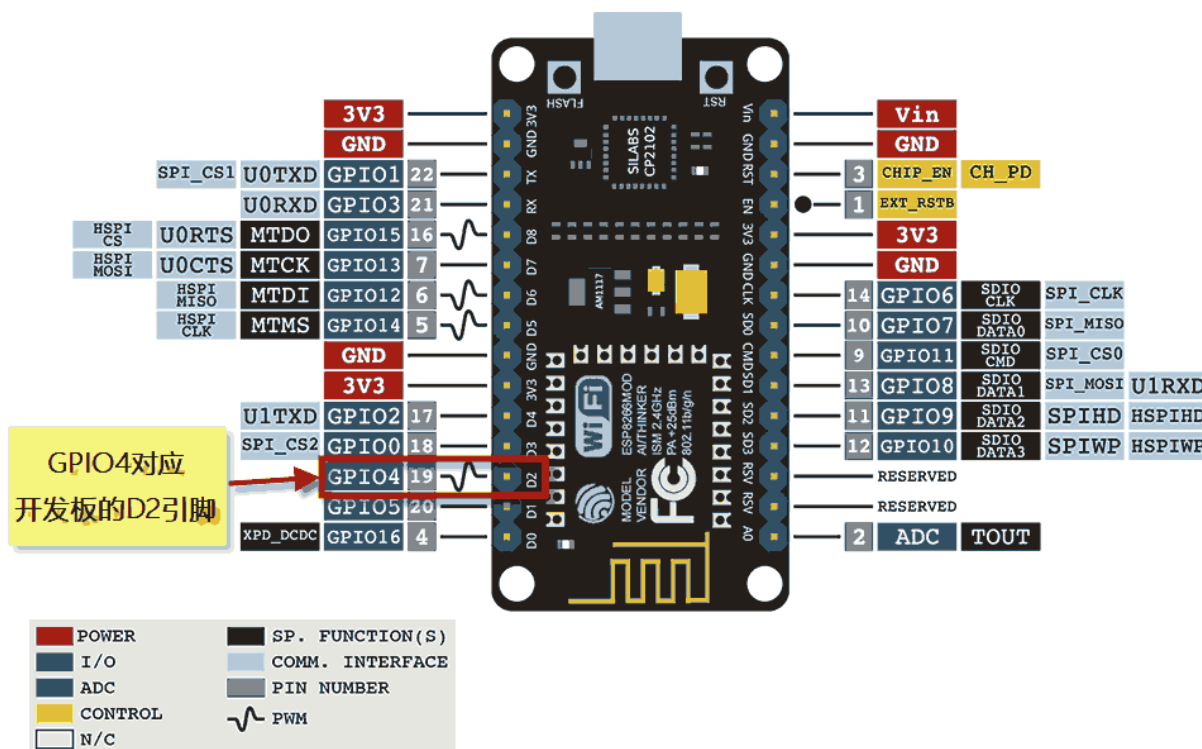
答案是GPIO4而不是D4。

通过下面这个GPIO编号与开发板引脚对应图我们不难看出。以上语句实际是将NodeMCU开发板的D2引脚设置为高电平。因为D2对应的时GPIO4。如下图所示：

## ESP-12E DEVELOPMENT BOARD PINOUT

### NOTES:

- ▲ Typ. pin current 6mA (Max. 12mA)
- ▲ For sleep mode, connect GPIO16 and EXT\_RSTB. On wakeup, GPIO16 will output LOW for system reset.
- ▲ On boot/reset/wakeup, keep GPIO15 LOW and GPIO2 HIGH.



NodeMCU的GPIO4对应开发板的D2引脚

如果要对D2引脚设置为高电平，您可以使用以下语句。

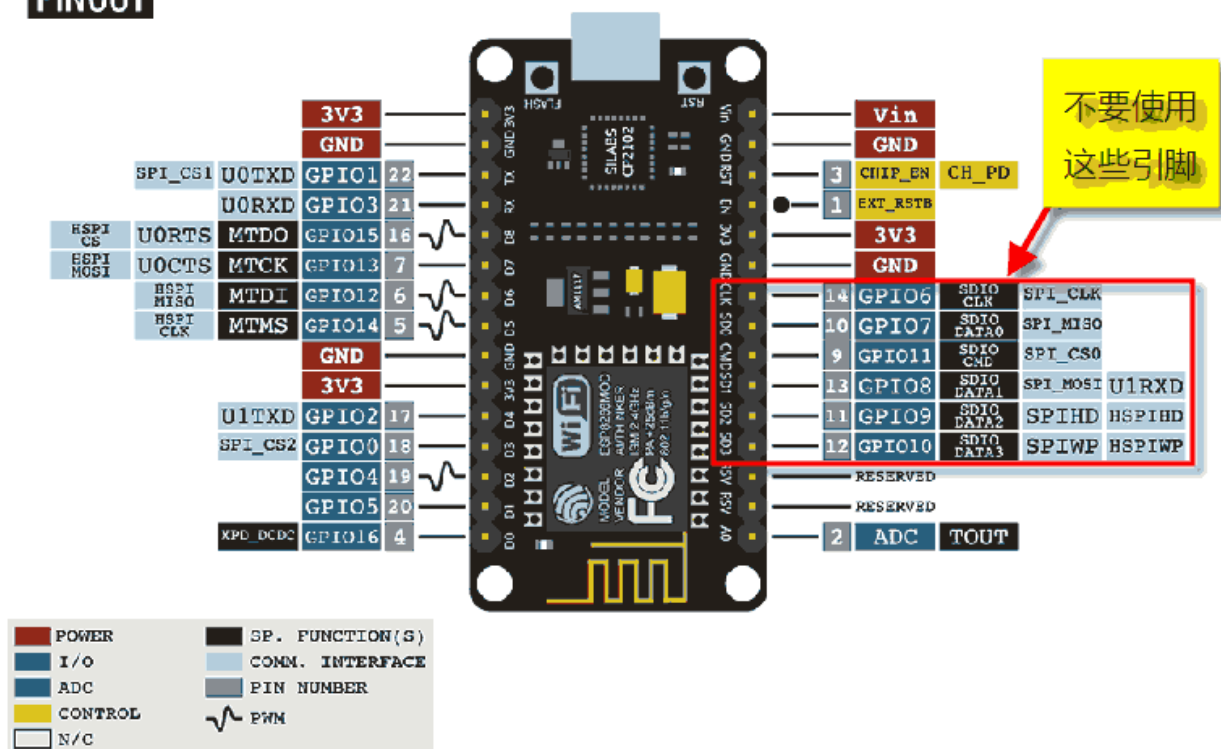
```
digitalWrite(D2, HIGH);
```

### 可用引脚

ESP8266芯片有17个GPIO引脚（GPIO0 ~ GPIO16）。这些引脚中的GPIO6 ~ GPIO 11被用于连接开发板的闪存（Flash Memory）。如果在实验电路中使用GPIO6 ~ GPIO11，NodeMCU开发板将无法正常工作。因此建议您**不要使用GPIO6 ~ GPIO 11**。

## ESP-12E DEVELOPMENT BOARD

## PINOUT



ESP8266的GPIO6-GPIO11用于连接闪存，因此不建议使用这些引脚。

## 电压电流限制

NodeMCU开发板引脚的输入输出电压限制是3.3 V。如果向引脚施加3.6V以上的电压就有可能对芯片电路造成损坏。同时请注意，这些引脚的最大输出电流是12mA。

由于NodeMCU开发板的引脚允许电压和电流都是低于Arduino开发板的引脚，所以如您想要将NodeMCU与Arduino引脚相互连接，请特别注意这两个开发板的引脚电压和电流的区别。如果操作不当可能会损坏NodeMCU开发板。

## 特殊引脚情况说明

GPIO2引脚 在NodeMCU开发板启动时是不能连接低电平的。

GPIO15引脚在开发板运行中一直保持低电平状态。因此请不要使用GPIO15引脚来读取开关状态或进行I<sup>2</sup>C通讯。

GPIO0引脚在开发板运行中需要一直保持高电平状态。否则ESP8266将进入程序上传工作模式也就无法正常工作了。您无需对GPIO0引脚进行额外操作，因为NodeMCU的内置电路可

以确保GPIO0引脚在工作时连接高电平而在上传程序时连接低电平。

## 上拉电阻/下拉电阻

GPIO 0-15引脚都配有内置上拉电阻。这一点与Arduino十分类似。GPIO16 引脚配有内置下拉电阻。

## 模拟输入

ESP8266 只有一个模拟输入引脚（该引脚通过模拟-数字转换将引脚上的模拟电压数值转化为数字量）。此引脚可以读取的模拟电压值为 0 – 1.0V。请注意：ESP8266 芯片模拟输入引脚连接在1.0V以上电压可能损坏ESP8266芯片。

以上所描述的是针对ESP8266芯片的引脚。而对于NodeMCU开发板引脚，情况就不同了。

NodeMCU开发板配有降压电路。您可以用NodeMCU开发板的模拟输入引脚读取0-3.3V的模拟电压信号。

## 通讯

### 串行端口

ESP8266有2个硬件串行端口（UART）。

串行端口0（UART0）使用GPIO1和GPIO3引脚。其中GPIO1引脚是TX0，GPIO3是RX0。

串行端口1（UART1）使用GPIO2和GPIO8引脚。其中GPIO2引脚是TX1，GPIO8是RX1。**请注意，由于GPIO8被用于连接闪存芯片，串行端口1只能使用GPIO2来向外发送串行数据。**

### I<sup>2</sup>C

ESP8266只有软件模拟的I<sup>2</sup>C端口，没有硬件I<sup>2</sup>C端口。也就是说我们可以使用任意的两个GPIO引脚通过软件模拟来实现I<sup>2</sup>C通讯。ESP8266的数据表（datasheet）中，GPIO2标注为SDA，GPIO14标注为SCL。



SPI

ESP8266的SPI端口情况如下：

- GPIO14 — CLK
- GPIO12 — MISO
- GPIO13 — MOSI
- GPIO 15 — CS(SS)

ESP8266引脚功能一览

GPIO	功能	状态	限制
0	引导模式选择	3.3V	无Hi-Z
1	TX0	—	串口通讯过程中不能使用
2	引导模式选择 TX1	3.3V	启动时不能接地 启动时发送调试信息
3	RX0	—	串口通讯过程中不能使用
4	SDA (I <sup>2</sup> C)	—	—
5	SCL (I <sup>2</sup> C)	—	—
6 – 11	连接闪存	x	不可用
12	MISO (SPI)	—	—
13	MOSI (SPI)	—	—
14	SCK (SPI)	—	—
15	SS (SPI)	0V	上拉电阻不可用
16	睡眠唤醒	—	无上拉电阻，仅有下拉电阻 连接 RST 引脚实现睡眠唤醒

[<- 上一节](#)

[下一章目录 ->](#)