

kerwin cui

Rome is not build in one day

博客园

首页

新随笔

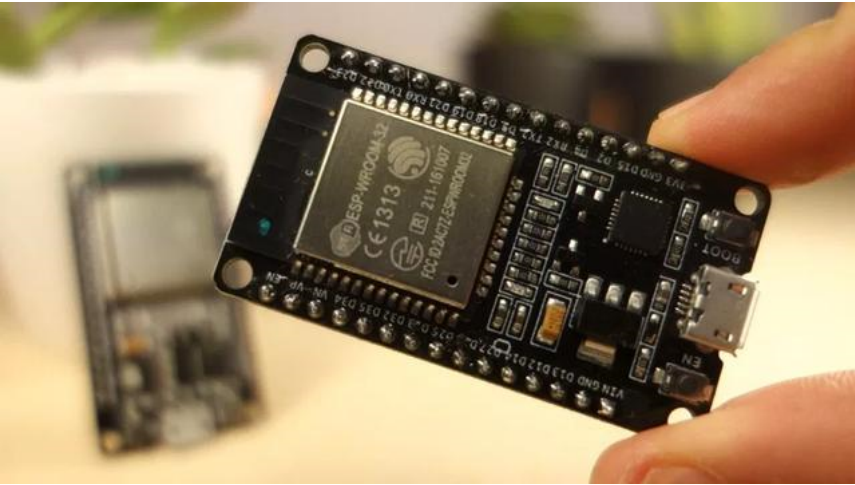
联系

管理

随笔 - 196 文章 - 0 评论 - 0

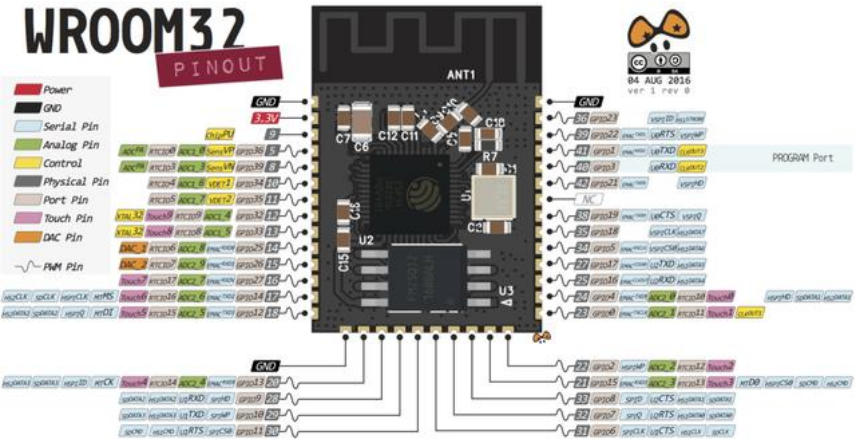
ESP32引脚参考(转)

ESP32芯片配有48个具有多种功能的引脚。并非所有的引脚都暴露在所有ESP32开发板中，有些引脚不能使用。



关于如何使用ESP32 GPIO有很多问题。你应该用什么pin? 在项目中应该避免使用哪些pin? 本文旨在成为一个为ESP32的GPIO提供一个简单易懂的参考指南。

下图显示了ESP-WROOM-32引脚。如果使用ESP32裸芯片构建自定义板，可以将其用作参考：



注意：并非所有的GPIO都可以在所有的开发板中访问，但是每个特定的GPIO都以相同的方式工作，而不管您使用的是什么开发板。

ESP32外围设备

ESP32外围设备包括：

公告

昵称: kerwin cui  
园龄: 8年8个月  
粉丝: 10  
关注: 0  
+加关注

搜索

我的标签

- esp32(32)
- Android(9)
- stm32(9)
- C语言(9)
- spring cloud(5)
- 单片机(4)
- java(4)
- mvc(4)
- linux(4)
- sql(4)
- 更多

随笔分类

- .Net开发(24)
- Android开发(9)
- Java开发(13)
- Linux系统(5)
- 编程语言(13)
- 机器人(2)
- 其他(25)
- 前端开发(7)
- 嵌入式开发(72)
- 数据库(15)
- 物联网(1)
- 运维测试(12)

友情链接

物美智能

阅读排行榜

- 1. postgresql 使用游标
- 2. 依赖注入是什么? (C
- 3. ESP32引脚参考(转)
- 4. STM32中STD、HA
- 5. postgresql分组后获

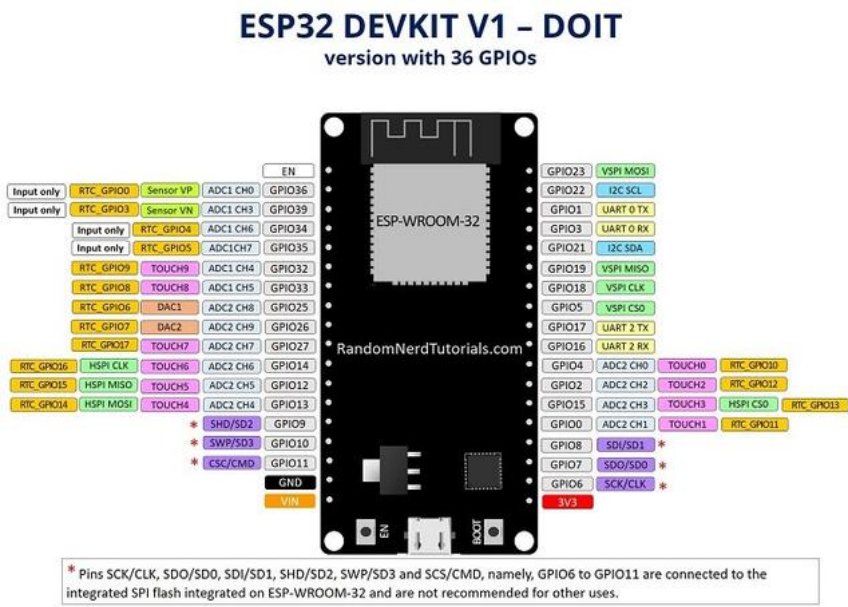
评论排行榜

- 1. wumei-smart智能身

- 18个模数转换器（ADC）通道
- 3个SPI接口
- 3个UART接口
- 2个I2C接口
- 16个脉冲宽度调制输出通道
- 2个数模转换器（DAC）
- 2个I2S接口
- 10个电容传感GPIO

ADC（模数转换器）和DAC（数模转换器）功能分配给特定的静态引脚。但是，您可以决定哪些管脚是UART、I2C、SPI、PWM等，您只需要在代码中分配它们。这是可能的，因为ESP32芯片的多路复用功能。

尽管您可以在软件上定义管脚属性，但默认情况下会分配管脚，如下图所示（这是带有36个管脚的ESP32 DEVKIT V1 DOIT板的示例-管脚位置可能会根据制造商而更改）。



此外，还有具有特定功能的管脚，使它们适合或不适合特定项目。下表显示了哪些管脚最适合作输入和输出，哪些管脚需要小心。

绿色突出显示的管脚可以使用。黄色突出显示的可以使用，但您需要注意，因为它们可能在启动时有意外行为。不建议将红色突出显示的管脚用作输入或输出。

- 2. WSL中使用systemc
  - 3. ESP32的Flash加密!
  - 4. 长按短按控制LED灯
- ### 推荐排行榜
- 1. 分布式、集群和微服
  - 2. ASP.NET MVC5中的
  - 3. Mybatis内嵌对象或
  - 4. ADC电阻分压采集精
  - 5. Spring Cloud组件和

- ### 最新评论
- 1. Re:WSL中使用systemc  
service service\_name
  - 2. Re:wumei-smart智!  
牛
  - 3. Re:长按短按控制LE  
支持!
  - 4. Re:ESP32的Flash加  
你好，目前我使用ESF  
开发模式中怎么烧录主  
eFuse中，还是编译之  
烧录到eFuse中了。谢

GPIO	Input	Output	Notes
0	pulled up	OK	Strapping 管脚
1	TX Pin	OK	debug output at boot
2	OK	OK	Strapping 管脚
3	OK	RX Pin	HIGH at boot
4	OK	OK	
5	OK	OK	Strapping 管脚
6	✗	✗	模组集成 SPI flash
7	✗	✗	模组集成 SPI flash
8	✗	✗	模组集成 SPI flash
9	✗	✗	模组集成 SPI flash
10	✗	✗	模组集成 SPI flash
11	✗	✗	模组集成 SPI flash
12	OK	OK	Strapping 管脚
13	OK	OK	
14	OK	OK	
15	OK	OK	Strapping 管脚
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		输入引脚
35	OK		输入引脚
36	OK		输入引脚
39	OK		输入引脚

### 仅输入的管脚

GPIO34到39是GPIOs—仅输入的管脚。这些引脚没有内部上拉或下拉电阻。它们不能用作输出，因此只能将这些管脚用作输入：

GPIO 34

GPIO 35

GPIO 36

GPIO 39

### 集成在ESP-WROOM-32上的SPI闪存

GPIO 6到GPIO 11在一些ESP32开发板中是公开的。但是，这些引脚连接到ESP-WROOM-32芯片上的集成SPI闪存，不建议用于其他用途。所以，不要在项目中使用这些管脚：

GPIO 6 (SCK/CLK)

GPIO 7 (SDO/SD0)

GPIO 8 (SDI/SD1)

GPIO 9 (SHD/SD2)

GPIO 10 (SWP/SD3)

GPIO 11 (CSC/CMD)

### 电容式触摸GPIO

ESP32有10个内部电容式触摸传感器。它们能感应到任何带电物质的变化，比如人类皮肤。因此，他们可以检测到当用手指触摸gpio时引起的变化。这些引脚可以很容易地集成到电容垫，并取代机械按钮。电容式触针也可以用来唤醒ESP32的深度睡眠。

这些内部触摸传感器连接到这些GPIO：

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)

#### 模数转换器 (ADC)

ESP32有18 x 12位ADC输入通道（而ESP8266只有1 x 10位ADC）。这些是可用作ADC和相应通道的GPIO：

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)

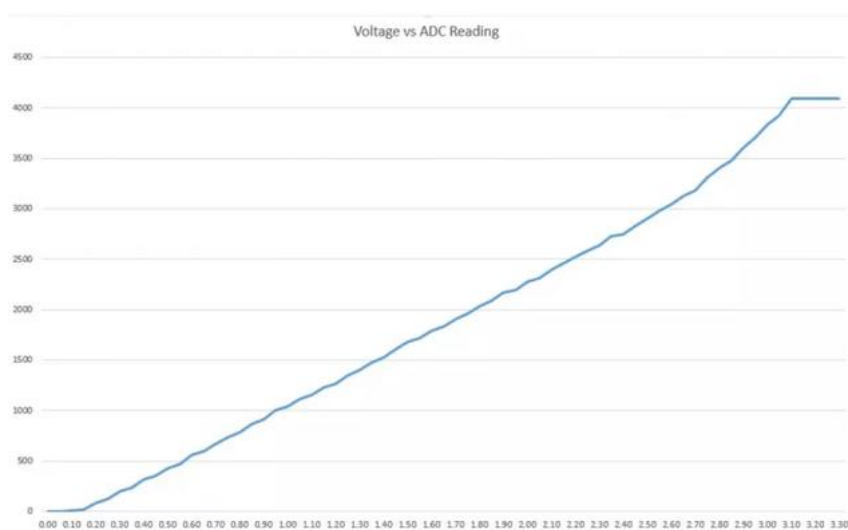
ADC2\_CH8 (GPIO 25)

ADC2\_CH9 (GPIO 26)

注意：使用Wi-Fi时不能使用ADC2管脚。因此，如果您使用Wi-Fi，并且无法从ADC2 GPIO获取值，则可以考虑改用ADC1 GPIO，这应该可以解决您的问题。

ADC输入通道具有12位分辨率。这意味着您可以获得0到4095之间的模拟读数，其中0对应于0V，4095对应于3.3V。您还可以在代码上设置通道的分辨率以及ADC范围。

ESP32 ADC引脚没有线性行为。您可能无法区分0和0.1V，或3.2和3.3V。在使用ADC引脚时，您需要记住这一点。您将得到类似于下图所示的行为。



#### 数模转换器 (DAC)

ESP32上有2 x 8位DAC通道，用于将数字信号转换为模拟电压信号输出。这些是DAC通道：

- DAC1 (GPIO25)
- DAC2 (GPIO26)

**RTC GPIOs**

ESP32支持RTC GPIO。当ESP32处于深度睡眠状态时，可以使用路由到RTC低功耗子系统的GPIO。这些RTC gpio可用于在超低功耗（ULP）协处理器运行时将ESP32从深度睡眠中唤醒。以下GPIO可以用作外部唤醒源。

- 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)

**脉冲宽度调制**

ESP32 LED PWM控制器有16个独立通道，可以配置为生成具有不同特性的PWM信号。所有可以作为输出的管脚都可以用作PWM管脚（GPIOs 34到39不能产生PWM）。

要设置脉冲宽度调制信号，需要在代码中定义这些参数：

- 信号频率；
  - 占空比；
  - 脉宽调制通道；
  - 要输出信号的GPIO。
- I2C

ESP32有两个I2C通道，任何管脚都可以设置为SDA或SCL。将ESP32与Arduino IDE一起使用时，默认I2C引脚为：

- GPIO 21（SDA）
- GPIO 22（SCL）

如果要使用其他管脚，在使用导线库时，只需调用：

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

**SPI**

默认情况下，SPI的pin映射为：

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

## 中断

所有GPIO都可以配置为中断。

## Strapping pins

ESP32芯片具有以下Strapping pins:

GPIO 0

GPIO 2

GPIO 4

GPIO 5 (启动期间必须为高)

GPIO 12 (启动期间必须低)

GPIO 15 (启动期间必须为高)

这些用于将ESP32置于引导加载程序或烧录模式。在大多数内置USB/Serial的开发板上, 您不需要担心这些管脚的状态。电路板使引脚处于正确的烧录或启动模式。有关ESP32引导模式选择的更多信息, 请参见此处。

但是, 如果你有外设连接到这些引脚, 你可能会遇到困难, 试图上传新的代码, 用新固件烧录ESP32或重置板。如果您有一些外设连接到Strapping pins, 并且您在上载代码或烧录ESP32时遇到问题, 可能是因为这些外设阻止ESP32进入正确的模式。请阅读引导模式选择文档以指导您朝正确的方向前进。复位、烧录或启动后, 这些引脚按预期工作。

## Pins HIGH at Boot

一些GPIO在启动或重置时将其状态更改为高或输出PWM信号。这意味着, 如果输出连接到这些gpio, 则在ESP32重置或引导时可能会得到意外的结果。

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

## 启用 (EN)

启用 (EN) 是3.3V调节器的启用引脚。它被拉起来了, 所以接地使3.3V调节器失效。例如, 这意味着您可以使用连接到按钮的该管脚来重新启动ESP32。

## GPIO电流消耗

根据ESP32数据表中的“推荐操作条件”部分, 每个GPIO的绝对最大电流为40毫安。

ESP32内置霍尔效应传感器

ESP32还具有内置霍尔效应传感器, 可检测周围磁场的变化。

转自: <https://www.cnblogs.com/Insane/p/13756430.html>

分类: 嵌入式开发

标签: esp32



kerwin cui

关注 - 0

粉丝 - 10

[+加关注](#)

« 上一篇: [c语言格式字符](#)

» 下一篇: [按键控制LED灯-ESP32中断处理](#)

posted @ 2020-11-06 00:33 kerwin cui 阅读(5472) 评论(0) 编辑 收藏 举报

[刷新评论](#) [刷新页面](#) [返回顶部](#)

登录后才能查看或发表评论, 立即 [登录](#) 或者 [逛逛](#) 博客园首页

【推荐】百度智能云2022开年大促0.4折起, 企业新用户享高配优惠

【推荐】开发者必读: 华为 2022 年移动应用趋势洞察白皮书

**编辑推荐:**

- 记我第一次做线下技术分享的那些事
- 换个数据结构，一不小心节约了 591 台机器！
- [ASP.NET Core] MVC 模型绑定：非规范正文内容的处理
- 全面认识数据指标体系
- 项目开发中对成长的一些思考

**最新资讯:**

- 华为x86业务被卖半年后，超聚变活得还好吗？
  - 都这样了，奈雪的茶还要开300多家新店？
  - 咖啡格局已变，还能容下瑞幸？
  - 图灵奖颁给超算跑分开创者！Matlab&SciPy都跟他有关
  - 不容易的网易云音乐
- » 更多新闻...