

ESP8266

Sleep 模块

功能描述



版本 1.0

乐鑫科技 IOT 团队

<http://bbs.espressif.com>

版权 © 2015

免责声明和版权公告

本文中的信息，包括供参考的 URL 地址，如有变更，恕不另行通知。

文档“按现状”提供，不负任何担保责任，包括对适销性、适用于特定用途或非侵权性的任何担保，和任何提案、规格或样品在他处提到的任何担保。本文档不负任何责任，包括使用本文档内信息产生的侵犯任何专利权行为的责任。本文档在此未以禁止反言或其他方式授予任何知识产权使用许可，不管是明示许可还是暗示许可。

Wi-Fi 联盟成员标志归 Wi-Fi 联盟所有。

文中提到的所有商标名称、商标和注册商标均属其各自所有者的财产，特此声明。

版权归 © 2015 乐鑫信息技术有限公司所有。保留所有权利。

目录

1. 概述.....	1
2. Modem-sleep.....	2
2.1. 特性.....	2
2.2. 接口说明.....	2
2.3. 应用.....	2
3. Light-sleep.....	3
3.1. 特性.....	3
3.2. 接口说明.....	3
3.3. 外部唤醒.....	3
3.4. 应用.....	3
4. Deep-sleep.....	4
4.1. 特性.....	4
4.2. 接口说明.....	4
4.2.1. 进入 Deep-sleep.....	4
4.2.2. 配置 Deep-sleep.....	4
4.3. 外部唤醒.....	5
4.4. 应用.....	5



1.

概述

ESP8266 系列芯片提供以下三种可配置的睡眠模式，您可以结合具体需求选择和配置。

- Modem-sleep
- Light-sleep
- Deep-sleep

三种模式的区别如下：

项目		Modem-sleep	Light-sleep	Deep-sleep
Wi-Fi		关闭	关闭	关闭
系统时钟		开启	关闭	关闭
RTC		开启	开启	开启
CPU		开启	暂停	关闭
衬底电流		15 mA	0.6 mA	~ 20 μ A
平均电流	DTIM = 1	16.2 mA	1.8 mA	-
	DTIM = 3	15.4 mA	0.9 mA	
	DTIM = 10	15.2 mA	0.55 mA	

说明：

- 对于 Modem-sleep 和 Light-sleep 模式，SDK 提供接口来使能睡眠模式，并由系统底层决定何时进入睡眠。具体请参考“2. Modem-sleep”和“3. Light-sleep”。
- 在 Deep-sleep 模式下，何时进入睡眠由用户控制，调用接口函数就可立即进入 Deep-sleep 模式。具体请参考“4. Deep-sleep”。
- RTC（Real-Time Clock）
- DTIM（Delivery Traffic Indication Message）



2.

Modem-sleep

2.1. 特性

目前 ESP8266 的 Modem-sleep 仅工作在 Station 模式下，连接路由器后生效。ESP8266 通过 Wi-Fi 的 DTIM Beacon 机制与路由器保持连接。

说明：

一般路由器的 DTIM Beacon 间隔为 100 ms ~ 1000 ms。

在 Modem-sleep 模式下，ESP8266 会在两次 DTIM Beacon 间隔时间内，关闭 Wi-Fi 模块电路，达到省电效果，在下一次 Beacon 到来前自动唤醒。睡眠时间由路由器的 DTIM Beacon 时间决定。睡眠同时可以保持与路由器的 Wi-Fi 连接，并通过路由器接受来自手机或者服务器的交互信息。

2.2. 接口说明

通过以下接口进入 Modem-sleep 模式。

```
wifi_set_sleep_type(MODEM_SLEEP_T)
```

说明：

在 Modem-sleep 模式下，系统可以自动被唤醒，无需配置接口。

2.3. 应用

Modem-sleep 一般用于必须打开芯片 CPU 的应用场景，例如 PWM 彩灯，需要 CPU 实时控制。



3. Light-sleep

3.1. 特性

Light-sleep 的工作模式与 Modem-sleep 相似，不同的是，除了关闭 Wi-Fi 模块电路以外，在 Light-sleep 模式下，还会关闭时钟并暂停内部 CPU，比 Modem-sleep 功耗更低。

3.2. 接口说明

通过以下接口进入 Light-sleep 模式。

```
wifi_set_sleep_type(LIGHT_SLEEP_T)
```

说明：

在 Wi-Fi 连接后，并且 CPU 处于空闲状态时，会自动进入 Light-sleep 状态。

3.3. 外部唤醒

在 Light-sleep 模式下，CPU 在暂停状态下不会响应来自外围硬件接口的信号与中断，因此需要配置通过外部 GPIO 信号将 ESP8266 唤醒，唤醒过程小于 3 ms。

通过 GPIO 唤醒只能配置为电平触发模式，接口如下。

```
void gpio_pin_wakeup_enable(uint32 i, GPIO_INT_TYPE intr_state);
```

uint32 i	唤醒功能的 IO 序号。
	唤醒的触发模式。
GPIO_INT_TYPE intr_state	<ul style="list-style-type: none">GPIO_PIN_INTR_LOLEVELGPIO_PIN_INTR_HILEVEL

3.4. 应用

Light-sleep 模式可用于需要保持与路由器的连接，可以实时响应路由器发来的数据的场合。并且在未接收到命令时，CPU 可以处于空闲状态。比如 Wi-Fi 开关的应用，大部分时间 CPU 都是空闲的，直到收到控制命令，CPU 才需要进行 GPIO 的操作。

说明：

若系统应用中有小于 DTIM Beacon 间隔时间的循环定时，系统将不能进入 Light-sleep 模式。



4. Deep-sleep

4.1. 特性

相对于其他两种模式，Deep-sleep 由用户控制，调用接口函数就可立即进入 Deep-sleep 模式。在该模式下，芯片会断开所有 Wi-Fi 连接与数据连接，进入睡眠模式，只有 RTC 模块仍然工作，负责芯片的定时唤醒。

使用 Deep-sleep 必须将 GPIO16 与芯片 EXT_RSTB 管脚连接。

4.2. 接口说明

4.2.1. 进入 Deep-sleep

接口名称

通过以下接口使能 Deep-sleep。

```
void system_deep_sleep(uint32 time_in_us)
```

参数说明

`uint32 time_in_us = 0` 不会定时唤醒，即不会主动醒来。

`uint32 time_in_us ≠ 0` 会在设定的时间后，自动唤醒（单位 μs ）。

4.2.2. 配置 Deep-sleep

可以通过以下接口配置 Deep-sleep 唤醒时的软件工作流程，从而影响长期运行的平均功耗。

```
bool system_deep_sleep_set_option(uint8 option)
```

`deep_sleep_set_option(0)` 由 init 参数的第 108 字节 控制 Deep-sleep 醒来后的是否作 RF 校准。

`deep_sleep_set_option(1)` 表示下一次 Deep-sleep 醒来后要作 RF 校准，功耗较大。

`deep_sleep_set_option(2)` 表示下一次 Deep-sleep 醒来后不作 RF 校准，功耗较小。

`deep_sleep_set_option(4)` 表示下一次 Deep-sleep 醒来后，不打开 RF，和 Modem-sleep 一样，电流最小。

**说明：**

init 参数即 esp_init_data_default.bin 内的参数值。比如，将第108 字节的数据改为 8，并且调用 deep_sleep_set_option(0)，则表示芯片每 8 次Deep-sleep 唤醒才会进行 RF 校准。

应用详情参考：<http://bbs.espressif.com/viewtopic.php?f=5&t=272>。

4.3. 外部唤醒

在 Deep-sleep 状态下，可以通过外部 IO 在芯片 EXT_RSTB 管脚上产生一个低电平脉冲，芯片即可被唤醒并启动。

△ 注意：

如果自动唤醒与外部唤醒须要同时作用，须要在外部电路设计时，使用合适的线逻辑操作电路。

4.4. 应用

Deep-sleep 可以用于低功耗的传感器应用，或者大部分时间都不需要进行数据传输的情况。设备可以每隔一段时间从 Deep-sleep 状态醒来测量数据并上传，之后继续进入 Deep-sleep。也可以将多个数据存储在 RTC memory（RTC memory 在 Deep-sleep 模式下仍然可以保存数据），然后一次发送出去。