ESP8266

Sleep Mode

Function Description

Version 1.0

Espressif Systems IOT Team

http://bbs.espressif.com

Copyright © 2015

Disclaimer and Copyright Notice

Information in this document, including URL references, is subject to change without notice.

THIS DOCUMENT IS PROVIDED AS IS WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY

WARRANTY OF MERCHANTABILITY, NON-INFRINGEMENT, FITNESS FOR ANY PARTICULAR

PURPOSE, OR ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION

OR SAMPLE. All liability, including liability for infringement of any proprietary rights, relating to use of information in this document is disclaimed. No licenses express or implied, by estoppel or

otherwise, to any intellectual property rights are granted herein.

The Wi – Fi Alliance Member Logo is a trademark of the Wi – Fi Alliance.

All trade names, trademarks and registered trademarks mentioned in this document are property of their respective owners, and are hereby acknowledged.

Copyright © 2015 Espressif Systems Inc. All rights reserved.

Table of Contents

1. Overview

ESP8266 series chip provides the following 3 sleep modes that can be configured, you can choose and and configure with the specific requirements.

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

The differences of the 3 sleep modes are shown as Table 1-1.

Table1-1. The differences of the 3 sleep modes

Item		Modem-sleep	Light-sleep	Deep-sleep
Wi-Fi		Turn off	Turn off	Turn off
System Clock		Turn on	Turn off	Turn off
RTC		Turn on	Turn on	Turn on
CPU		Turn on	Pending	Turn off
Substrate current		15 mA	0.6 mA	~ 20 μA
Average current	DTIM = 1	16.2 mA	1.8 mA	–
	DTIM = 3	15.4 mA	0.9 mA	
	DTIM = 10	15.2 mA	0.55 mA	

Notes:

- For modem-sleep and light-sleep mode, SDK provides interfaces to enable sleep mode, and the system decides when to go into sleep. For the details refer to ***2. Modem-sleep*** and ***3. Light-sleep***.
- In deep-sleep mode, you can call the function to control when to go into sleep. For the details refer to ***4. Deep-sleep***.
- RTC (Real-Time Clock)
- DTIM (Delivery Traffic Indication Message)

2. Modem-sleep

2.1. Feature

The modem-sleep of ESP8266 only works on the station mode, and will be enabled when connecting to the router. ESP8266 keep connecting with the router through the DTIM beacon mechanism.

> ***Note***:
> *The DTIM beacon interval of the router is from 100 ms to 1000 ms.*

In modem-sleep mode, the ESP8266 will close the Wi-Fi module circuit within the two DTIM Beacon interval in order to the power savings, and it will be automatically waked up before the next Beacon arrival. The sleep time is decided by Beacon DTIM of router. During sleep, ESP8266 can keep the Wi-Fi connecting with router, and receiving the interactive information from mobile phone or server.

2.2. Interface

The system will go into the modem-sleep mode via the following interface.

```
wifi_set_sleep_type(MODEM_SLEEP_T)
```

> ***Note***:
> In the modem-sleep, the system can be waken up automatically, you don't need to configure the interface.

2.3. Application

Modem-sleep is generally used to the application scenarios that should keep the CPU chip power on, such as Pulse Width Modulation (PWM) light needs CPU control with real-time.

3. Light-sleep

3.1. Feature

The working mode of light-sleep is similar to the modem-sleep mostly, the difference is light-sleep will power off the clock and suspend internal CPU, that will cause a lower power consumption than modem-sleep.

3.2. Interface

The system will go into the light-sleep mode via the following interface.

```
wifi_set_sleep_type(LIGHT_SLEEP_T)
```

> *Note*:
>
> *After the Wi-Fi connection, and CPU is idle, it automatically enters the light-sleep mode.*

3.3. External wake-up

Under the light-sleep mode, the CPU will not respond to the signal and interrupt from the peripheral hardware interface under the pause state. Therefore, the ESP8266 need to be waked up via external GPIO, and the wake process is less than 3 ms.

The GPIO wake can only be configured as the level trigger mode, the interface is as follows.

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

uint32 i	The IO serial number of the wake-up function.
GPIO_INT_TYPE intr_state	The trigger mode of wake-up. • GPIO_PIN_INTR_LOLEVEL • GPIO_PIN_INTR_HILEVEL

3.4. Application

Light-sleep mode can be used to the application scenarios that can real-time respond to the sending data from the router with which CPU is keeping connection. And the CPU can be idle when the command is not received. For example, the application of the Wi-Fi switch, the CPU need to carry out GPIO operation until receives the control commands while most of the time it's idle.

Note:

If the cycle time of the system application is less than beacon interval time of DTIM, the light-sleep mode won't be enabled.

4. Deep-sleep

4.1. Feature

Compared to the other two modes, the deep-sleep is controlled by the user, and the deep-sleep can be enabled immediately when user calls interface function. In this mode, the chip will disconnect all Wi-Fi connection and data connection and enter the sleep mode. Only the RTC module is still working, responsible for the timing of the chip wake-up.

When using the deep-sleep, you must connect the GPIO16 with the EXT_RSTB pin of the chip.

4.2. Interface

4.2.1. Go into deep-sleep

Go into the deep-sleep mode via the following interface.

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

uint32 time_in_us = 0	Don't wake up at regular intervals.
uint32 time_in_us ≠ 0	Automatically wake up at regular intervals.

4.2.2. Configure deep-sleep

The software workflow of the waking-up from deep-sleep can be configured by the following interface to affect the average power consumption of the long run.

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

deep_sleep_set_option(0)	The 108th byte of esp_init_data_default.bin decides whether make a RF calibration after the waking-up from deep-sleep.
deep_sleep_set_option(1)	Make RF calibration after the waking-up from deep-sleep with a higher power consumption.
deep_sleep_set_option(2)	Don't make RF calibration after the waking-up from deep-sleep with a lower power consumption.
deep_sleep_set_option(4)	Don't turn on RF after the waking-up from deep-sleep with a lowst power consumption as modem-sleep.

Note:

The init parameter is the parameter in esp_init_data_default.bin. For example, change the 108th byte of data to 8, and call deep_sleep_set_option (0), which means that chip will be undertaken the RF calibration every 8 times Deep-sleep wake up. For the details refer to:

http://bbs.espressif.com/viewtopic.php?f=5&t=272.

4.3. External wake-up

In the Deep-sleep state, a low level pulse can be generated from the chip IO pin via an external EXT_RSTB, and the chip can be awakened and launched.

⚠ **Notice:**

If the automatic wake-up and the external wake-up need to be the same time, we need to use the appropriate line logic operation circuit when the external circuit is designed.

4.4. Application

Deep-sleep can be used for low-power sensor applications, or the cases that do not need data transmission most of the time. The device can wake up from the deep-sleep state at intervals to measure and upload the data, and then go to the deep-sleep again. You can also store multiple data in the memory RTC (memory RTC can still save the data in the deep-sleep mode), and then send it at a time.