

ESP8266

FLASH TOOL

使用手册



版本 1.0

乐鑫科技 IOT 团队

<http://bbs.espressif.com>

Copyright © 2015

本手册对应的 ESP8266 FLASH TOOL 的软件版本为 V 2.4。

日期	版本	发布说明
2015.11	V1.0	第一次发布

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 乐鑫信息科技（上海）有限公司所有。保留所有权利。

目录

1. 概述.....	1
2. 使用方法.....	2
2.1. 烧录	2
2.2. 射频初始化设置	5
3. 烧录文件及地址说明.....	7
3.1. 不支持云端升级	7
3.1.1. 512 KB Flash	7
3.1.2. 1024 KB Flash	7
3.1.3. 2048 KB Flash	8
3.1.4. 4096 KB Flash	8
3.2. 支持云端升级 (FOTA)	8
3.2.1. 512 KB Flash	9
3.2.2. 1024 KB Flash	9
3.2.3. 2048 KB Flash	10
3.2.4. 4096 KB Flash	10



概述

1.

ESP8266 FLASH TOOL 是由 Espressif 官方开发的烧录工具，用户可根据实际的编译方式和 Flash 的容量，将 SDK 编译生成的多个 bin 文件一键烧录到 ESP8266 母板的 SPI Flash 中。

ESP8266 FLASH TOOL 官方下载地址为：

<http://bbs.espressif.com/viewtopic.php?f=5&t=433>

说明：

具体 SDK 软件编译方法可参考“ESP8266 SDK 使用手册”。



2.

使用方法

2.1. 烧录

ESP8266 FLASH TOOL 可在Windows XP 或 Windows 7 环境下运行，通常情况下烧录时需保证 3.3 V 电源供应，外部电源输出电流在 500 mA 及以上。以 ESP LAUNCHER 开发板为例（见图1），通过本工具将 bin 文件烧录至 SPI Flash 的方法如下：

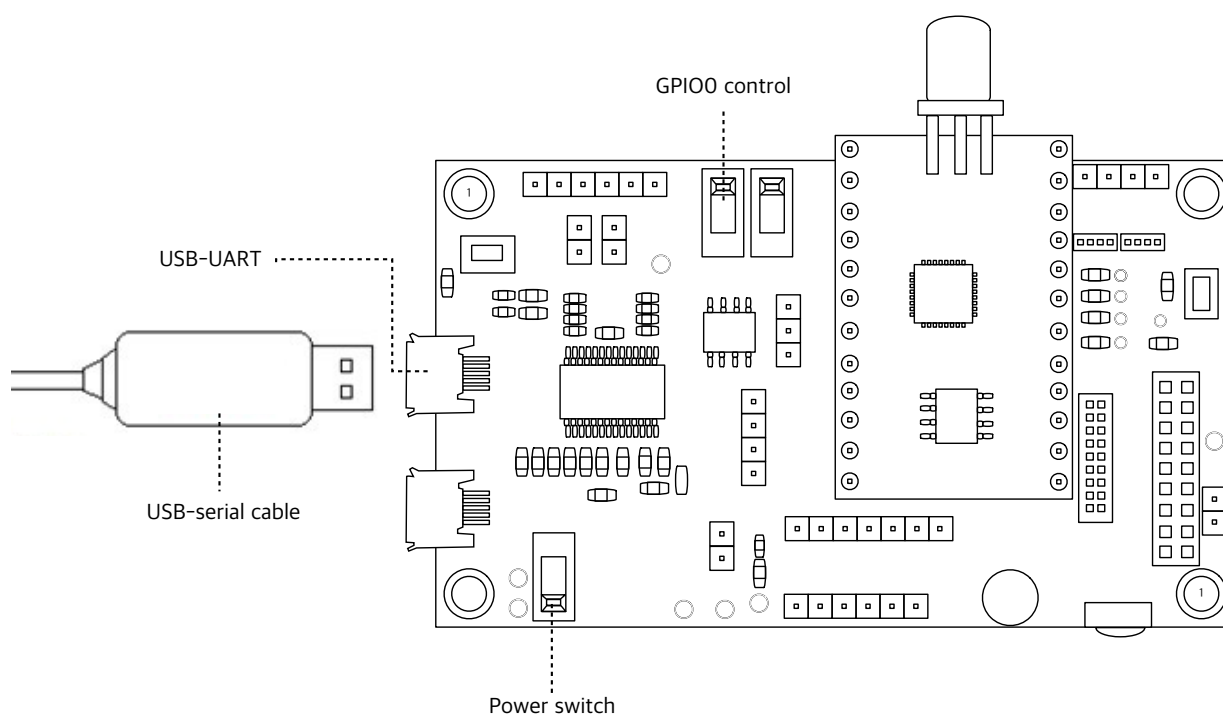


图 1: ESP-LAUNCHER 实例图

1. 用一根串口线将 PC 机的 USB 口与 ESP LAUNCHER 的 USB-UART 接口相连，或将 ESP8266 母板上的 GND, U0RXD 和 U0TXD 通过串口线连接至 PC。

说明：

ESP-LAUNCHER 上有两个 USB-UART 接口，烧录须使用上方的 USB-UART 接口（见图 1）。

更多 ESP-LAUNCHER 的信息，请见“ESP8266 硬件描述”



- 在 ESP LAUNCHER 开发板上，将 GPIO0 开关拨到下方进入 UART Download 模式，或将 ESP8266 母板上跳线设置为 MTDO : GPIO0 : GPIO2 = 0 : 0 : 1。

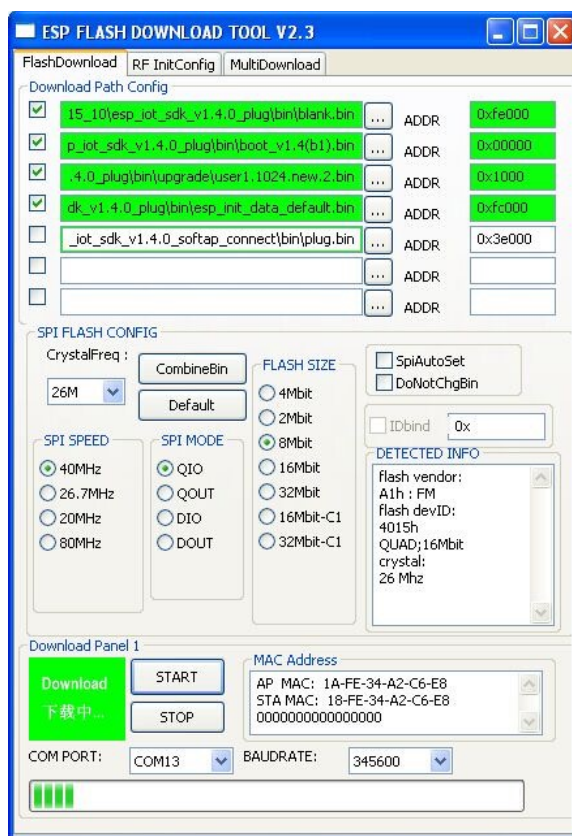



图 2: ESP FLASH DOWNLOAD TOOL - FlashDownload 页签

- 将电源开关拨到上方上电。
- 双击 `ESP_DOWNLOAD_TOOL.exe` 打开烧录工具进行烧录配置（见图2）。
- 在 **Download Path Config** 区域内点击  选择需要烧录的bin文件并勾选，在 **ADDR** 内设置相应的烧录地址。

说明：

烧录文件与地址根据 *SPI Flash* 的型号及实际的使用需求而不同，具体信息可参考第三章。

- 配置 SPI Flash 属性如下。



SPI FLASH CONFIG	
CrystalFreq	根据实际选用的晶振型号选择晶振频率。
CombineBin	将勾选的 bin 文件合成一个 target.bin，烧录地址为 0x0000。
Default	将 SPI Flash 的配置恢复到默认值。
SPI SPEED	选择 SPI Flash 的读写速度，最大值为 80 MHz。
SPI MODE	根据实际使用的 Flash 选择对应的模式。如果 Flash 采用 Dual SPI，选择 DIO 或 DOUT ；如果 Flash 采用 Quad SPI，选择 QIO 或 QOUT 。
FLASH SIZE	根据实际编译的配置对应选择的 Flash 大小。
SpiAutoSet	用户如果勾选 SpiAutoSet ，下载工具将会按照默认的 Flash map 下载，16 Mbit 和 32 Mbit 的 Flash map 会被设置为 512 Kbyte + 512 Kbyte。
DoNotChaBin	用户可勾选 DoNotChgBin ，Flash 的运行频率，方式和布局会以用户编译时的配置选项为准；如果不勾选该选项，Flash 的运行频率，方式和布局会以编译器最终的配置为准。
Download Panel	
START	点击 START 开始烧录。当烧录结束后，左边绿色状态显示 完成 。
STOP	点击 STOP 停止烧录。
MAC Address	烧录成功后，系统会显示 ESP8266 的 MAC 地址。烧录 target.bin 后，系统会显示 STA 和 AP 的 MAC 地址。
COM PORT	选择 ESP8266 的 COM 端口。
BAUDRATE	选择烧录的波特率，默认为 115200。

说明：

不建议勾选 **SpiAutoSet**，推荐用户根据实际情况对 Flash 进行手动配置。

7. 下载完成后，在 ESP LAUNCHER 开发板上将 GPIO0 开关拨动到上方，可进入 Flash boot 模式，即运行模式（ESP8266母板上跳线设置为 MTDO : GPIO0 : GPIO2 = 0 : 1 : 1）。

△ 注意：

请在断电状态下进行跳线操作。





2.2. 射频初始化设置

在烧录之前，用户可以进入 **RF InitConfig** 页签修改射频初始化设置（见图3），生成的 **esp_init_data_setting.bin** 可代替 **esp_init_data_default.bin** 烧录到 Flash 中。

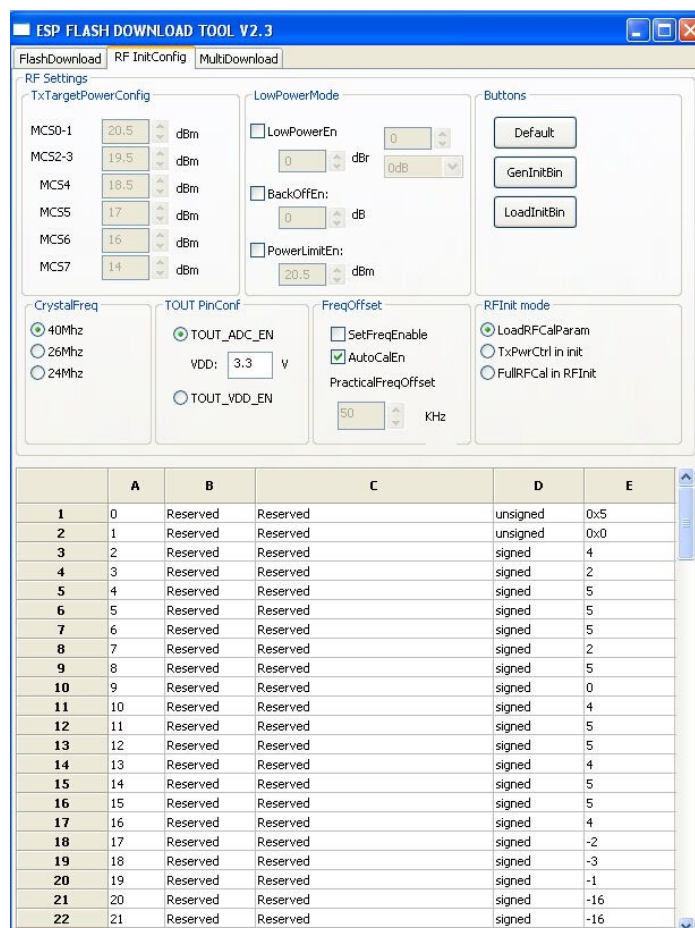


图 3: ESP FLASH DOWNLOAD TOOL - RF InitConfig 页签



PowerLimtEn	限定输出功率的最大值。
BackOffEn	为每个速率的功率设置统一的避退阶数。
LowPowerEn	开启低功耗模式，为每个速率设定统一的功耗值。
CrystalFreq	根据实际选用的晶振型号选择晶振频率，错选会影响正常使用。
TOUT PinConf	根据实际情况进行 TOUT 管脚的配置，错选可能会影响正常使用。 <ul style="list-style-type: none">• TOUT_ADC_EN: 在 TOUT 管脚接外部电路的情况下，使用芯片内部的 ADC 端口测量外部电压值 (0 V - 1 V)。不可同时使用 uint16 system_get_vdd33(void) 测量 VDD33 的电压值。• TOUT_VDD_EN: 在 TOUT 管脚悬空的情况下，通过 uint16 system_get_vdd33(void) 测量 VDD33 的电压值。
FreqOff	<ul style="list-style-type: none">• SetFreqEnable: 手动设置频偏值。• AutoCalEn: 自动调整频偏值。
RFInt mode	用户可选择射频的初始化行为： <ul style="list-style-type: none">• LoadRFCalParam: 初始化时射频数据直接从Flash读取，不作任何校准，耗时约 2 ms，初始电流最小。• TxPwrCtrl in init: 初始化时仅做 Tx Power 校准，其他从 Flash 读取，耗时约 20 ms，初始电流较小。• FullRFCal in RFin: 初始化时进行全部的校准，耗时约 200 ms，初始电流较大。

△ 注意：

选择 **TOUT_ADC_EN** 时，需要填写真实的 VDD3P3 管脚 3 和 4 上的电源电压。

说明：

用户也可以在最下方的表格中对 **esp_init_data_setting.bin** 的 0-127 byte 的参数进行修改。

设置完成后，点击 **GenInitBin** 按键，生成 **esp_init_data_setting.bin**。

用户也可以点击 **Default** 按键恢复默认值，或点击 **LoadInitBin** 按键，选择导入一个 bin 文件进行参数配置。



3. 烧录文件及地址说明

每个 bin 文件编译成功后都会提示该 bin 在 Flash 的烧录地址，烧录地址与 bin 文件根据实际使用需求和 Flash 的容量而不同。本章节提供了支持云端升级与不支持云端升级两种情况下，不同容量的 Flash (512/1024/2048/4096 KB) 需要烧录的 bin 文件及对应的烧录地址。

3.1. 不支持云端升级

3.1.1. 512 KB Flash

bin	烧录地址	说明
master_device_key.bin	0x3E000	用户在 Espressif Cloud 申请，依此享受 Espressif 云端服务
esp_init_data_default.bin	0x7C000	初始化射频参数，由 Espressif 在 SDK 中提供
blank.bin	0x7E000	初始化系统参数，由 Espressif 在 SDK 中提供
eagle.flash.bin	0x00000	主程序，编译代码生成
eagle.irom0text.bin	0x40000	主程序，编译代码生成

3.1.2. 1024 KB Flash

bin	烧录地址	说明
master_device_key.bin	0x3E000	用户在 Espressif Cloud 申请，依此享受 Espressif 云端服务
esp_init_data_default.bin	0xFC000	初始化射频参数，由 Espressif 在 SDK 中提供
blank.bin	0xFE000	初始化系统参数，由 Espressif 在 SDK 中提供
eagle.flash.bin	0x00000	主程序，编译代码生成
eagle.irom0text.bin	0x40000	主程序，编译代码生成



3.1.3. 2048 KB Flash

bin	烧录地址	说明
master_device_key.bin	0x3E000	用户在 Espressif Cloud 申请，依此享受 Espressif 云端服务
esp_init_data_default.bin	0x1FC000	初始化射频参数，由 Espressif 在 SDK 中提供
blank.bin	0x1FE000	初始化系统参数，由 Espressif 在 SDK 中提供
eagle.flash.bin	0x00000	主程序，编译代码生成
eagle.irom0text.bin	0x40000	主程序，编译代码生成

3.1.4. 4096 KB Flash

bin	烧录地址	说明
master_device_key.bin	0x3E000	用户在 Espressif Cloud 申请，依此享受 Espressif 云端服务
esp_init_data_default.bin	0x3FC000	初始化射频参数，由 Espressif 在 SDK 中提供
blank.bin	0x3FE000	初始化系统参数，由 Espressif 在 SDK 中提供
eagle.flash.bin	0x00000	主程序，编译代码生成
eagle.irom0text.bin	0x40000	主程序，编译代码生成

3.2. 支持云端升级 (FOTA)

说明：

支持云端升级 (FOTA) 的软件无需烧录 *user2.bin*，用户可以通过网络升级下载 *user2.bin* 到 Flash 并重启运行。后文仅为说明 *user2.bin* 的实际存放地址。



3.2.1. 512 KB Flash

bin	烧录地址	说明
master_device_key.bin	0x3E000	用户在 Espressif Cloud 申请，依此享受 Espressif 云端服务
esp_init_data_default.bin	0x7C000	初始化射频参数，由 Espressif 在 SDK 中提供
blank.bin	0x7E000	初始化系统参数，由 Espressif 在 SDK 中提供
boot.bin	0x00000	启动程序，由 Espressif 在 SDK 中提供，建议使用最新版本
user1.bin	0x01000	主程序，编译代码生成
user2.bin	0x41000	主程序，编译代码生成，无需烧录

3.2.2. 1024 KB Flash

bin	烧录地址	说明
master_device_key.bin	0x3E000 (建议更改)	用户在 Espressif Cloud 申请，依此享受 Espressif 云端服务，存放于用户参数区，IOT_Demo 中设置为 0x3E000，用户可自行更改。 使用 1 MB Flash 时，建议参考 BBS 修改，烧录到 0x7E000 http://bbs.espressif.com/viewtopic.php?f=10&t=305
esp_init_data_default.bin	0xFC000	初始化射频参数，由 Espressif 在 SDK 中提供
blank.bin	0xFE000	初始化系统参数，由 Espressif 在 SDK 中提供
boot.bin	0x00000	启动程序，由 Espressif 在 SDK 中提供，建议使用最新版本
user1.bin	0x01000	主程序，编译代码生成
user2.bin	0x81000	主程序，编译代码生成，无需烧录



3.2.3. 2048 KB Flash

bin	烧录地址	说明
master_device_key.bin	0x3E000 (建议更改)	用户在 Espressif Cloud 申请, 依此享受 Espressif 云端服务, 存放于用户参数区, IOT_Demo 中设置为 0x3E000, 用户可自行更改。 如果编译时 STEP 5 选择 3, 建议参考 BBS 修改, 烧录到 0x7E000 http://bbs.espressif.com/viewtopic.php?f=10&t=305 同理如果编译时 STEP 5 选择 5, 建议修改代码地址为 0xFE000, 并烧录到 0xFE000
esp_init_data_default.bin	0x1FC000	初始化射频参数, 由 Espressif 在 SDK 中提供
blank.bin	0x1FE000	初始化系统参数, 由 Espressif 在 SDK 中提供
boot.bin	0x00000	启动程序, 由 Espressif 在 SDK 中提供, 建议使用最新版本
user1.bin	0x01000	主程序, 编译代码生成
user2.bin	0x81000	主程序, 编译代码生成, 无需烧录

3.2.4. 4096 KB Flash

bin	烧录地址	说明
master_device_key.bin	0x3E000 (建议更改)	用户在 Espressif Cloud 申请, 依此享受 Espressif 云端服务, 存放于用户参数区, IOT_Demo 中设置为 0x3E000, 用户可自行更改。 如果编译时 STEP 5 选择 4, 建议参考 BBS 修改, 烧录到 0x7E000 http://bbs.espressif.com/viewtopic.php?f=10&t=305 同理如果编译时 STEP 5 选择 6, 建议修改代码地址为 0xFE000, 并烧录到 0xFE000
esp_init_data_default.bin	0x3FC000	初始化射频参数, 由 Espressif 在 SDK 中提供
blank.bin	0x3FE000	初始化系统参数, 由 Espressif 在 SDK 中提供
boot.bin	0x00000	启动程序, 由 Espressif 在 SDK 中提供, 建议使用最新版本
user1.bin	0x01000	主程序, 编译代码生成
user2.bin	0x81000	主程序, 编译代码生成, 无需烧录



说明：

- 系统参数区固定为 Flash 的最后四个扇区，每扇区 4 KB，即 Flash 最后 16 KB；
- 用户参数区地址由用户自定义，IOT_Demo 中设置为 0x3C000 开始的四个扇区，用户可以设置为任意未占用的地址。
- **master_device_key.bin** 是 ESP8266 设备享受 Espressif 云端服务的身份证明，如不使用 Espressif Cloud 可以不烧录，否则仅烧录一次即可；烧录地址在 IOT_Demo 中设置为用户参数区的第三个扇区；
- **blank.bin** 为初始化系统参数，烧录地址为 Flash 的倒数第二个扇区；
- **esp_init_data_default.bin** 为初始化射频相关参数，烧录地址为 Flash 的倒数第四个扇区；
- Espressif 提供的应用示例 IOT_Demo 默认为 512 KB Flash。