

Flash Download Tool

User Guide

Related Product:

- ESP32 Serials
- ESP8266 Serials



Version 1.1
Espressif Systems
Copyright © 2019

About This Guide

This document describes how to download and configure firmware on Espressif modules, using Espressif's Flash Download Tool. It also lists some frequently asked questions and answers. This document is applicable to Flash Download Tool V3.6.0 and later versions.

Release Notes

Date	Version	Release notes
2018.08	V1.0	Initial release
2019.03	V1.1	<ul style="list-style-type: none">• Updated Sections 3.2.2.5, 3.5, 5.1, and Appendix A;• Added Section 4.1.3;• Removed Section 5.6.

Documentation Change Notification

Espressif provides email notifications to keep customers updated on changes to technical documentation. Please subscribe <https://www.espressif.com/en/subscribe>.

Certification

Download certificates for Espressif products <https://www.espressif.com/en/certificates>.

Table of Contents

1. Preparation	1
2. Hardware	2
2.1. USB to Serial Port Test Board.....	2
2.2. Module to which firmware is downloaded.....	3
2.2.1. Working modes.....	3
2.2.2. Device Connection.....	3
3. Software	4
3.1. Software Interface.....	4
3.2. SPIDownload Tab.....	6
3.2.1. Common Configuration.....	6
3.2.2. Other Configuration.....	7
3.2.2.1. CombineBin Button.....	7
3.2.2.2. DoNotChgBin Option.....	7
3.2.2.3. Default Button.....	7
3.2.2.4. SpiAutoSet Option.....	8
3.2.2.5. LOCK SETTINGS Option.....	8
3.3. HSPIDownload Tab.....	8
3.4. MultiDownload Tab.....	9
3.5. RFConfig Tab.....	9
3.6. GPIOConfig Tab.....	9
4. Download	11
4.1. ESP32 Series Products.....	11
4.1.1. ESP32 Series Products that Use External Flash.....	11
4.1.2. ESP32 Series Products that Use Integrated Flash.....	16
4.1.3. Enable Encryption for Firmware Downloading.....	16
4.2. ESP8266 Series Products.....	21
4.2.1. ESP-WROOM-02.....	21
4.2.2. ESP-WROOM-S2.....	26
4.3. ESP8266 Series Products.....	27
5. FAQs	28
5.1. COM Related Errors.....	28

5.2. Synchronization Related Errors.....	28
5.3. Efuse Related Errors	29
5.4. Download Related Errors	29
5.5. Operation related error	29
A. Appendix - Contents of the Flash Download Tool Folder.....	30



1.

Preparation

The software and hardware resources required for downloading firmware to flash are listed below.

- Hardware:
 - 1 x USB test board
 - 1 x module to which firmware is downloaded
 - 1 x PC (Windows XP, Windows 7, Windows 10)
- Software:
 - [Flash Download Tool](#) (For the detailed structure of this tool, please refer to Appendix A)



2.

Hardware

2.1. USB to Serial Port Test Board

In this guide, ESP_Test Board (as shown below) is used as a USB-to-Serial Port test board, the core of which is a chip for USB-UART conversion. You can also purchase other USB-to-UART chips or test boards to connect the modules to a PC, and download firmware to modules.

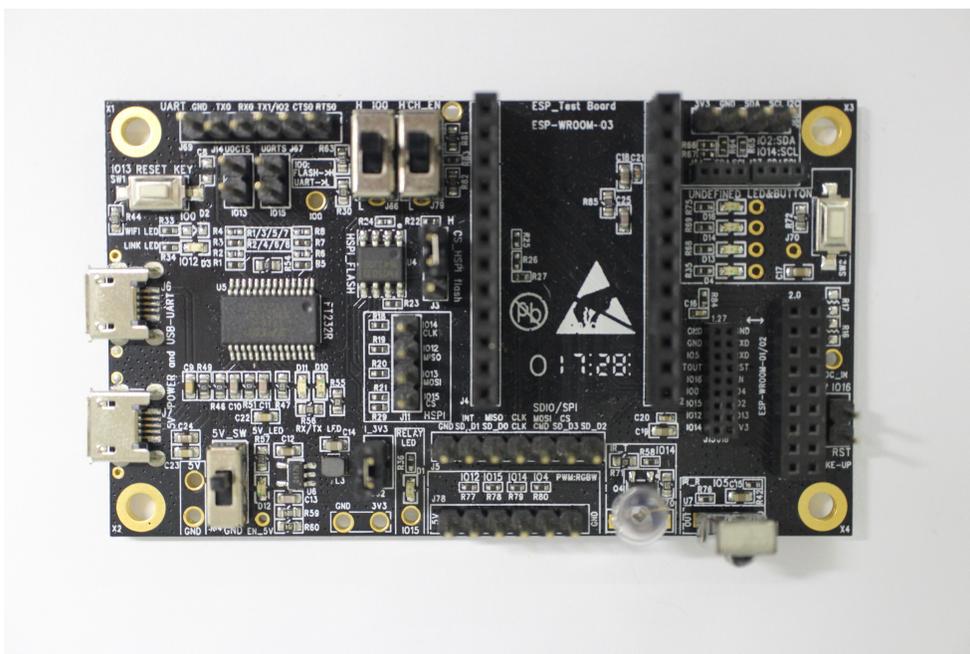


Figure 2-1. Espressif USB-to-Serial Port Board



2.2. Module to which firmware is downloaded

2.2.1. Working modes

The Espressif module has two working modes listed below:

- Downloading Mode: If IO0 is at low-voltage level, the chip works in downloading mode.
- Operating Mode: If IO0 is at high-voltage level, the chip works in operating mode.

2.2.2. Device Connection

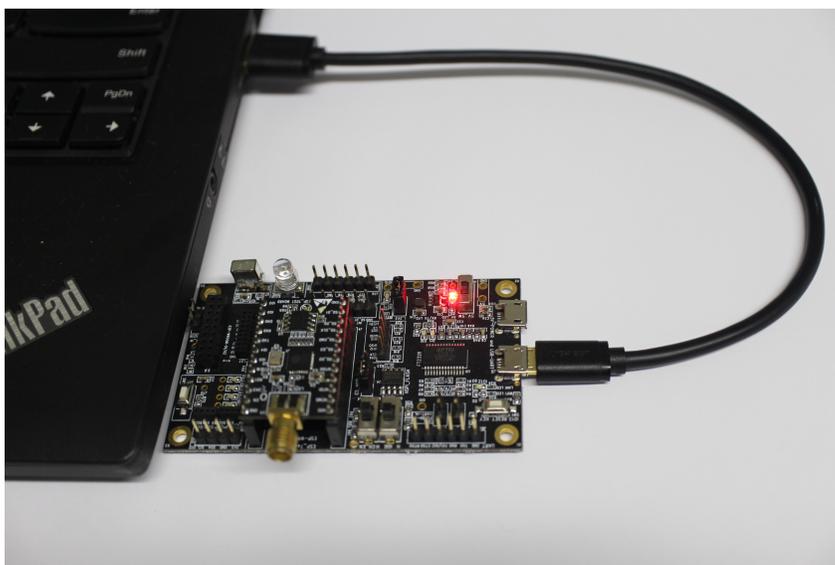


Figure 2-2. Device Connection for Flash Downloading - ESP8266 Series

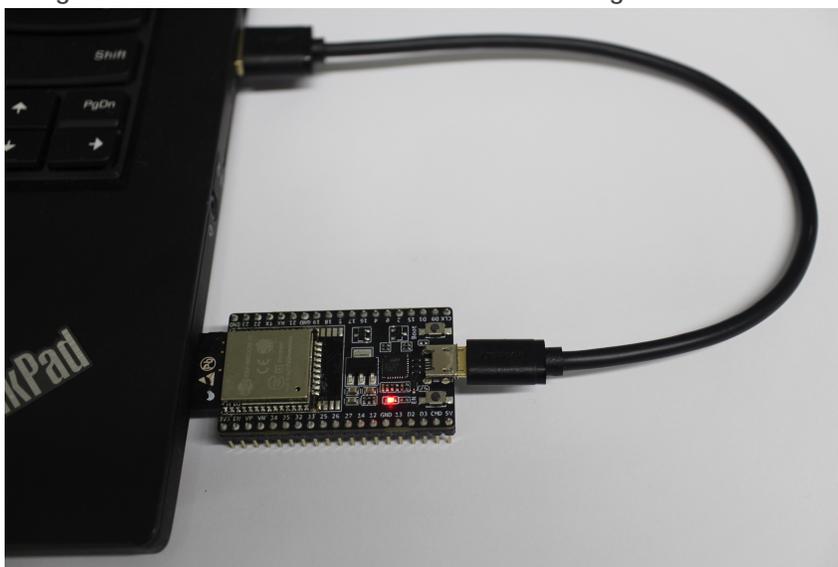


Figure 2-3. Device Connection for Flash Downloading - ESP32 Series



3.

Software

3.1. Software Interface

The main interface of Espressif's Flash Download Tool is shown in Figure 3-1. Choose a download option depending on the model you have.

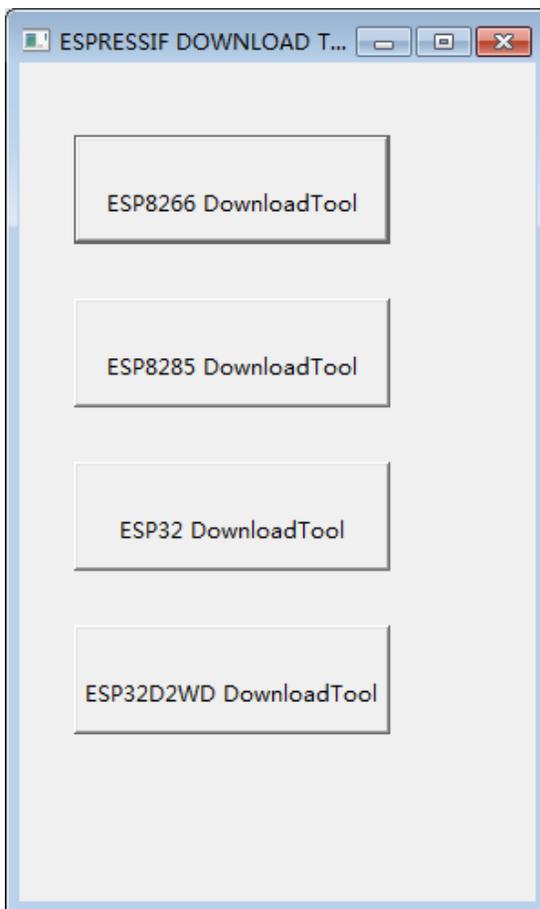


Figure 3-1. The Main Interface of Espressif's Flash Download Tool



For example, Figure 3-2 shows the screen of ESP8266 Download Tool.

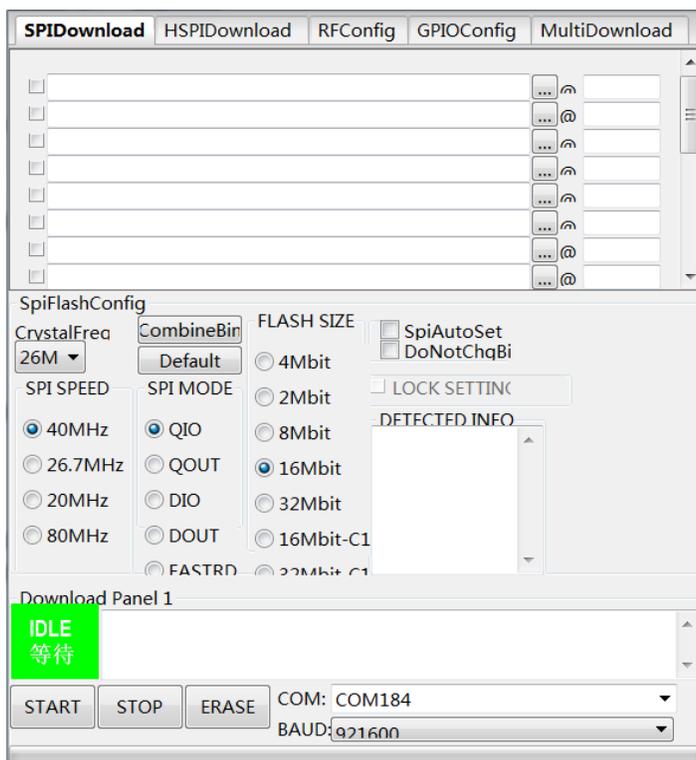


Figure 3-2. ESP8266 Download Screen

You can see five tabs on the *Download* screen:

- *SPIDownload* tab
- *HSPIDownload* tab
- *RFConfig* tab
- *GPIOConfig* tab
- *MultiDownload* tab



3.2. SPIDownload Tab

3.2.1. Common Configuration

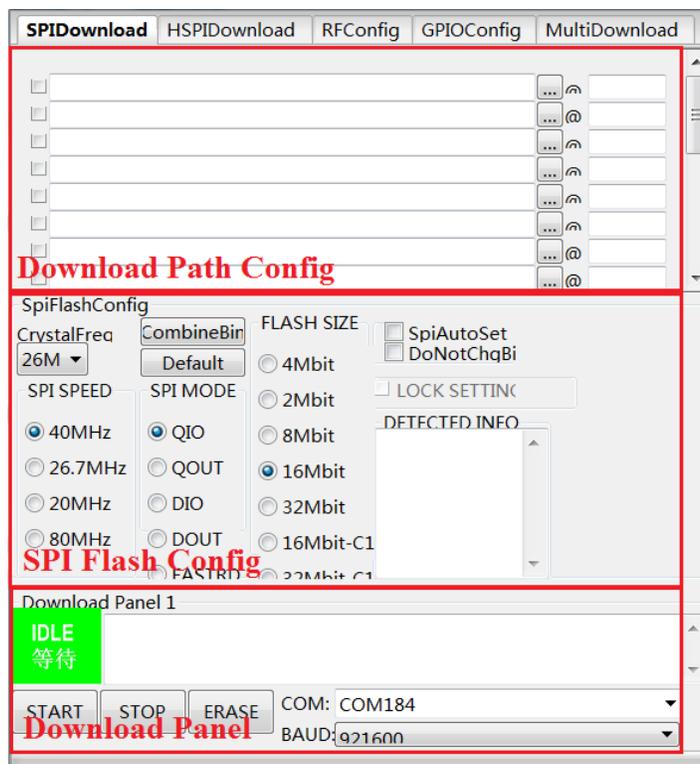


Figure 3-3. SPIDownload Tab

As shown in the figure above, the SPIDownload tab is mainly divided into the following blocks:

- **Download Path Config**, in which the following information should be specified:
 - Path for loading firmware
 - Address for downloading firmware (in hexadecimal format)
 - Path to the **MasterKey** folder (valid only if the Espressif IOT demo is used)
 - Address for downloading the **MasterKey** (in hexadecimal format)
- **SPI Flash Config**, which contains:
 - **CrystalFreq**: crystal oscillator frequency (for ESP32, this option can be ignored)
 - **SPI SPEED**: SPI boot rate
 - **SPI MODE**: SPI boot mode
 - ▶ For ESP8266, QIO is usually selected.
 - ▶ For ESP8285, DOUT must be always selected.
 - ▶ For ESP32, please fill in as per the specific compilation or select DoNotChqBi.



- **FLASH SIZE**: size of the flash being used in Mbits (For ESP8266, 16Mbit-C1 means that the flash mapping is 8 Mbit + 8 Mbit, and 32 Mbit-C means 4 Mbit + 4 Mbit)
- **DETECTED INFO**: flash & crystal oscillator information that is detected automatically
- Other: to be introduced in Section 3.2.2
- **Download Panel**, which contains:
 - **START**: start downloading
 - **STOP**: stop downloading
 - **ERASE**: erase the flash
 - **COM**: serial port used for downloading
 - **BAUD**: baud rate

3.2.2. Other Configuration

3.2.2.1. CombineBin Button

The **CombineBin** function can be used to combine all the selected firmware in **Download Path Config** into one firmware. The combined firmware contains all independent elements of the firmware, download addresses and **SPI Flash Config** information. For this reason, before using the **CombineBin** function, you should check if the address and the configuration of **SPI Flash Config** of each individual firmware are correct. The **CombineBin** function will combine all the selected firmware according to their download addresses and any unused areas will be filled with 0xff.

When you click the **Combine** button, the generated firmware will be saved as **./combine/target.bin**.

Note:

If you click the **CombineBin** button more than once, the newly generated bin file overwrites the previous target.bin file. Therefore, the generated target.bin should be renamed and properly saved in a timely manner.

3.2.2.2. DoNotChgBin Option

If the **DoNotChgBin** option is checked, **SPI Flash Config** will download the current flash configuration setting into the flash to ensure that the flash size and flash mode of the firmware are consistent with the user's settings.

Note:

This option is disabled by default. For ESP32, it's recommended to use other options to implement the complied configuration.

3.2.2.3. Default Button

The **Default** function restores the SPI configuration to default values.

**Note:**

It is not recommended to users. Currently, no flash used by Espressif module supports this function.

3.2.2.4. SpiAutoSet Option

This option automatically sets the SPI download configuration. When this option is enabled, the SPI will be configured with the flash information automatically detected during downloading.

Note:

Generally, this option is not recommended, because the actual hardware memory may use a different allocation logic from the system memory.

3.2.2.5. LOCK SETTINGS Option

This option locks the configuration interface, which will appear in gray after being clocked. It's mainly used for batch downloading and can prevent users from making any unintended changes. To enable this option, please do the followings:

Open the configuration file `./configure/esp**/spi_download.conf`, and under [LOCK], modify

```
lock_settings = 0
```

to:

```
lock_settings = 1
```

3.3. HSPIDownload Tab

The Download Screens for **HSPIDownload** and **SPIDownload** are identical. Please see the details in Section 3.2.



3.4. *MultiDownload* Tab

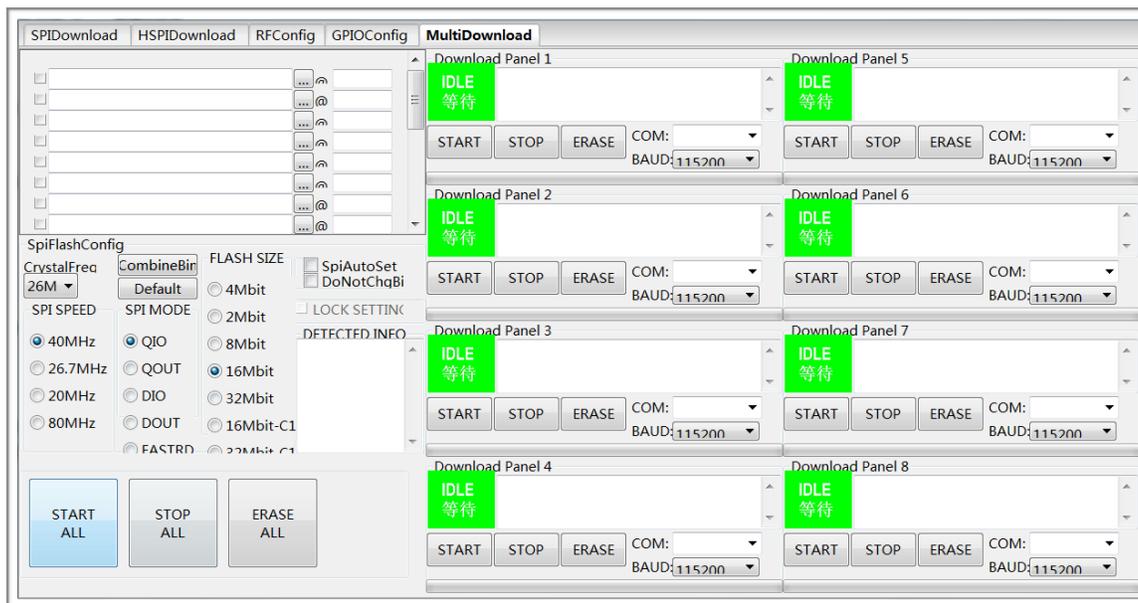


Figure 3-4. *MultiDownload* Tab

The *MultiDownload* screen is basically the same as the *SPIDownload* screen, except that you need to configure the serial port and baud rate of each channel separately.

3.5. *RFConfig* Tab

As this tab is for our internal debugging, users are recommended not to use it.

3.6. *GPIOConfig* Tab

This function modifies the level of the power-on pin by changing the bootloader and generating a new file.

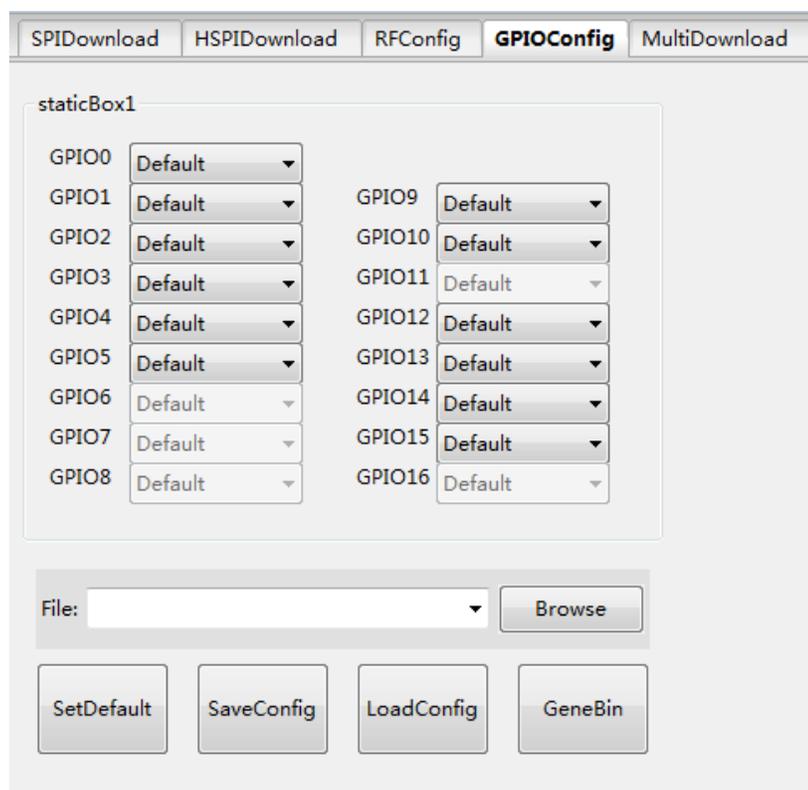


Figure 3-5. *GPIOConfig* Tab

As shown in Figure 3-5, the *GPIOConfig* tab is divided into the *staticsBox1* (pin level configuration), *File* (firmware load address), and *four function keys* described below:

- ***SetDefault***: restores all GPIOs to default values.
- ***SaveConfig***: saves the current GPIO configuration to the configuration file.
- ***LoadConfig***: loads the existing GPIO configuration from the configuration file.
- ***GeneBin***: saves the current GPIO configuration to the loaded bootloader file, and saves a new bootloader in the same directory.

Note:

1. This function is applicable only to *ESP8266* series.
2. The GPIO pins that are allowed to be modified in the *ESP8266* series do not include *IO6*, *IO7*, *IO8*, *IO11* and *IO16*.



4.

Download

4.1. ESP32 Series Products

4.1.1. ESP32 Series Products that Use External Flash

The download process for ESP32 modules that use different external flash is exactly the same. Since both ESP32-WROOM-32 and ESP32-WROVER use external flash, ESP32-WROOM-32 will be used here as an example.

Note:

1. If users want to download firmware to a customized module with PSRAM, IO12 must be set to pull-down and the downloading voltage of the flash must be set to 1.8 V when the module is powered up. Please note that the flash may be damaged if the operating voltage exceeds the maximum operating voltage of the module.
2. In this user guide, the bin files of the **AT** in IDF are used as the bin files to be downloaded.
3. The make commands can be used in the project to obtain the bin file to be downloaded, the address from which to download the bin file and the path to the bin file after the compilation.

1. Start the Flash Download Tool. The main interface will appear, as shown in Figure 4-1. Click **ESP32 DownloadTool** to access the interface for downloading firmware to an ESP32 module that uses external flash.



Figure 4-1. The Download Interface - ESP32



2. Provide information about the bin file to be downloaded and the address from which this bin will be downloaded. Also, configure the **CrystalFreq**, **SPI SPEED**, **SPI MODE**, **FLASH SIZE**, **COM** and **BAUD** settings, based on your actual case.

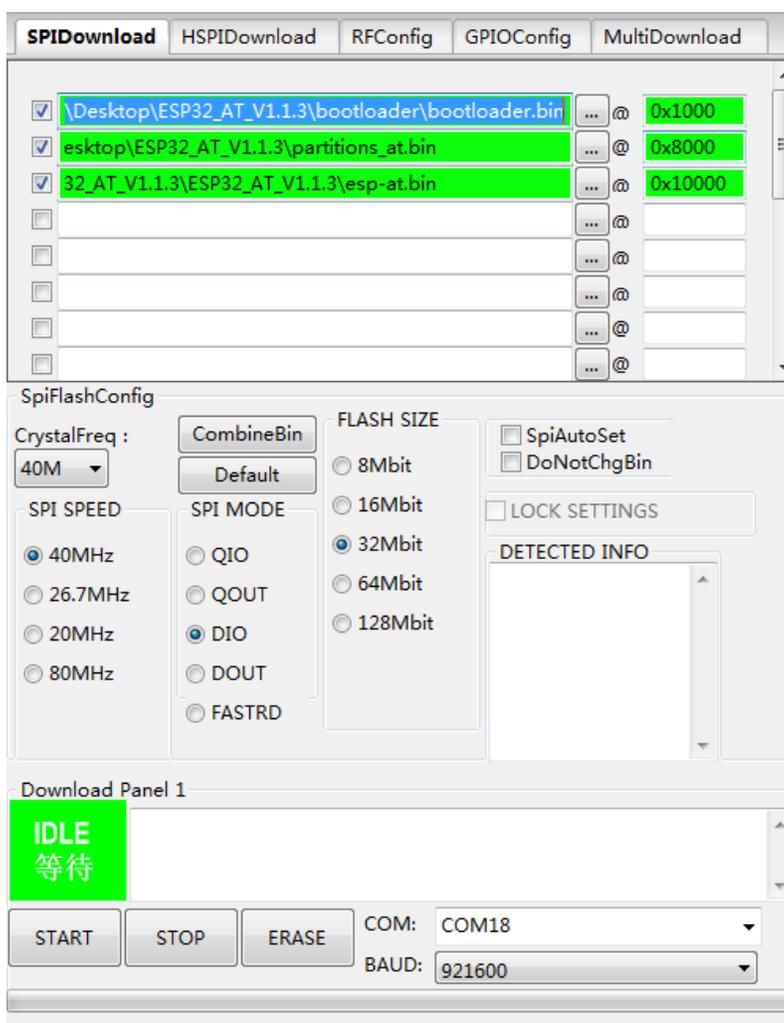


Figure 4-2. ESP32 DownloadTool – SPIDownload



3. Click **START** to start the download process. While downloading, the tool reads the flash information and MAC address of the product, as shown in Figure 4-3.

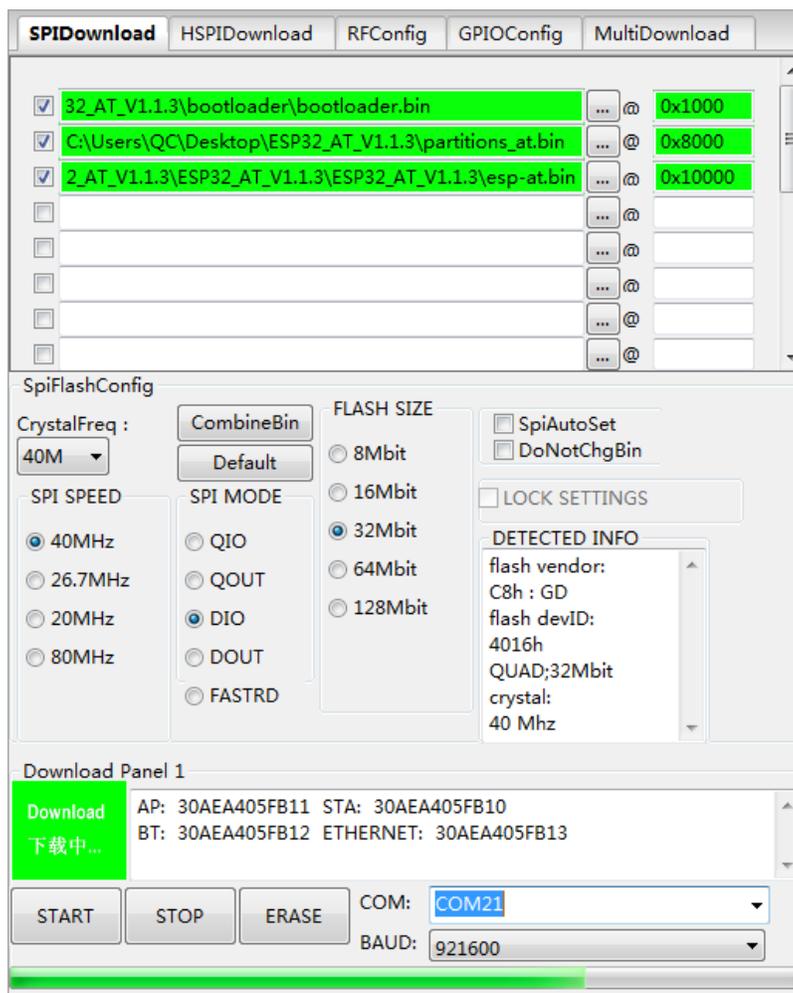


Figure 4-3. Downloading



- When the downloading is completed. You can see the green "**FINISH /完成**" icon in the lower left corner of the interface.

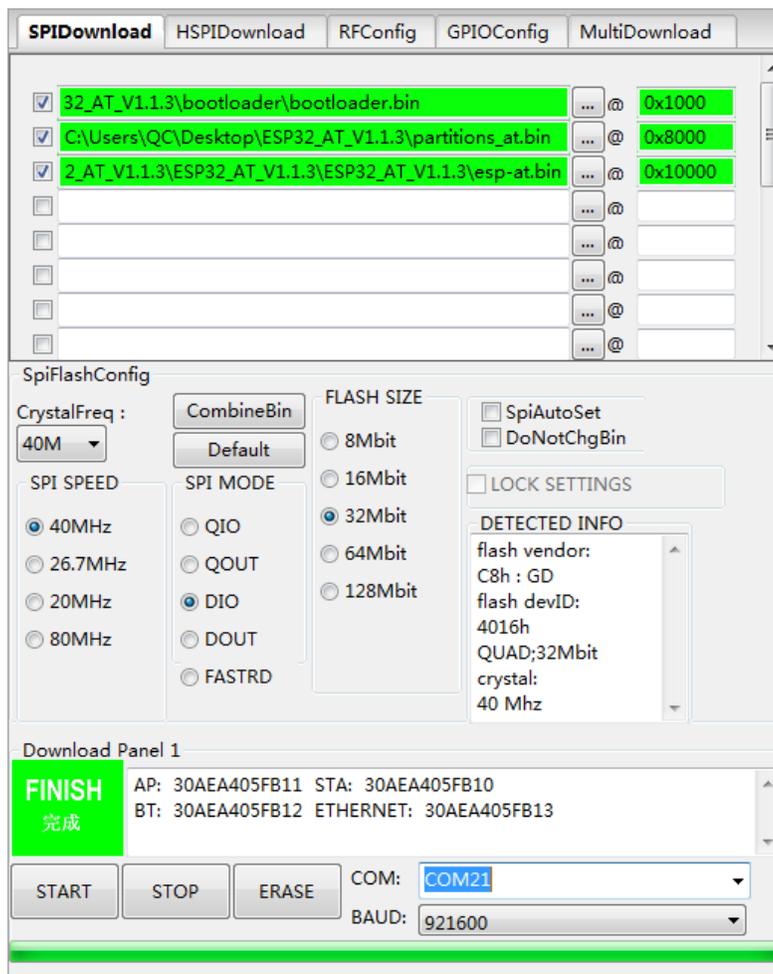


Figure 4-4. Finish Downloading



5. Open the serial port tool and power up the product again. The expected power-up log is shown in Figure 4-5.

```
[20:45:13.623] ets Jun  8 2016 00:22:57

rst:0x1 (POWERON_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
configsip: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:2
load:0x3fff0018,len:4
load:0x3fff001c,len:5744
[20:45:13.652] ho 0 tail 12 room 4
load:0x40078000,len:7684
load:0x40080000,len:7392
entry 0x4008039c
[0;32mI (30) boot: ESP-IDF v3.2-dev-43-gcf280e8-dirty 2nd stage bootloader[0m
[0;32mI (30) boot: compile time 14:44:50[0m
[0;32mI (30) boot: Enabling RNG early entropy source...[0m
[0;32mI (37) boot: SPI Speed      : 40MHz[0m
[0;32mI (41) boot: SPI Mode       : DIO[0m
[0;32mI (45) boot: SPI Flash Size : 2MB[0m
[0;32mI (49) boot: Partition Table:[0m
[0;32mI (52) boot: ## Label                Usage            Type ST Offset   Length[0m
[0;32mI (60) boot:  0 nvs                    Wifi data        01 02 00009000 00006000[0m
[0;32mI (67) boot:  1 phy_init              RF data          01 01 0000f000 00001000[0m
[0;32mI (75) boot:  2 factory                factory app      00 00 00010000 00100000[0m
[0;32mI (82) boot: End of partition table[0m
[0;32mI (86) esp_image: segment 0: paddr=0x00010020 vaddr=0x3f400020 size=0x076dc ( 30428)
map[0m
[0;32mI (106) esp_image: segment 1: paddr=0x00017704 vaddr=0x3ffb0000 size=0x022b4 ( 8884)
load[0m
[0;32mI (110) esp_image: segment 2: paddr=0x000199c0 vaddr=0x3ffb22b4 size=0x00000 (   0)
load[0m
[0;32mI (113) esp_image: segment 3: paddr=0x000199c8 vaddr=0x40080000 size=0x00400 ( 1024)
load[0m
[0;32mI (122) esp_image: segment 4: paddr=0x00019dd0 vaddr=0x40080400 size=0x06240 ( 25152)
load[0m
[0;32mI (141) esp_image: segment 5: paddr=0x00020018 vaddr=0x400d0018 size=0x13418 ( 78872)
```

Figure 4-5. The Expected Power-up log



4.1.2. ESP32 Series Products that Use Integrated Flash

The download process for ESP32 modules that use different types of integrated flash is exactly the same. Both ESP32-PICO-D4 and ESP32-D2WD use integrated flash. Here, ESP320-D2WD will be used as an example.

1. Start the Flash Download Tool. The main interface will appear, as shown in Figure 4-6. Choose **ESP32D2WD DownloadTool** to access the interface for downloading firmware to an ESP32 module that uses integrated flash.



Figure 4-6. ESP32D2WD Download Tool

2. The subsequent steps are identical to those for ESP32 products that use external flash. For details, see Section 4.1.1.

4.1.3. Enable Encryption for Firmware Downloading

1. To configure the encryption function, open the configuration file `./configure/esp32/security.conf` in Notepad. The following options are available to be configured (True: enable; False: disable):
 - [DEBUG MODE]



In this mode, you can specify the key file path, so that all products can share the identical key.

- debug_enable = False (Disable debug mode)
- debug_pem_path = (key file path)
- [SECURE BOOT]
secure boot related configurations:
 - secure_boot_en = False (Disable secure boot)
 - burn_secure_boot_key = False (Disable burning the secure boot key)
 - secure_boot_force_write = False (Disable force writing the secure boot key)
 - secure_boot_rw_protect = False (Disable read/write protection after writing the key)
- [FLASH ENCRYPTION]
flash encryption related configurations:
 - flash_encryption_en = False (Disable flash encryption)
 - burn_flash_encryption_key = False (Disable burning the flash encryption key)
 - flash_encrypt_force_write = False (Disable force writing the flash encryption key)
 - flash_encrypt_rw_protect = False (Disable read/write protection after burning the flash encryption key)
 - reserved_burn_times = 0 (How many times [0 in this case] are reserved for the burning operation.)

See the typical configuration below:

```
[DEBUG MODE]
debug_enable = False
debug_pem_path =

[SECURE BOOT]
secure_boot_en = True
burn_secure_boot_key = True
secure_boot_force_write = True
secure_boot_rw_protect = True

[FLASH ENCRYPTION]
flash_encryption_en = True
burn_flash_encryption_key = True
flash_encrypt_force_write = True
flash_encrypt_rw_protect = True
reserved_burn_times = 0

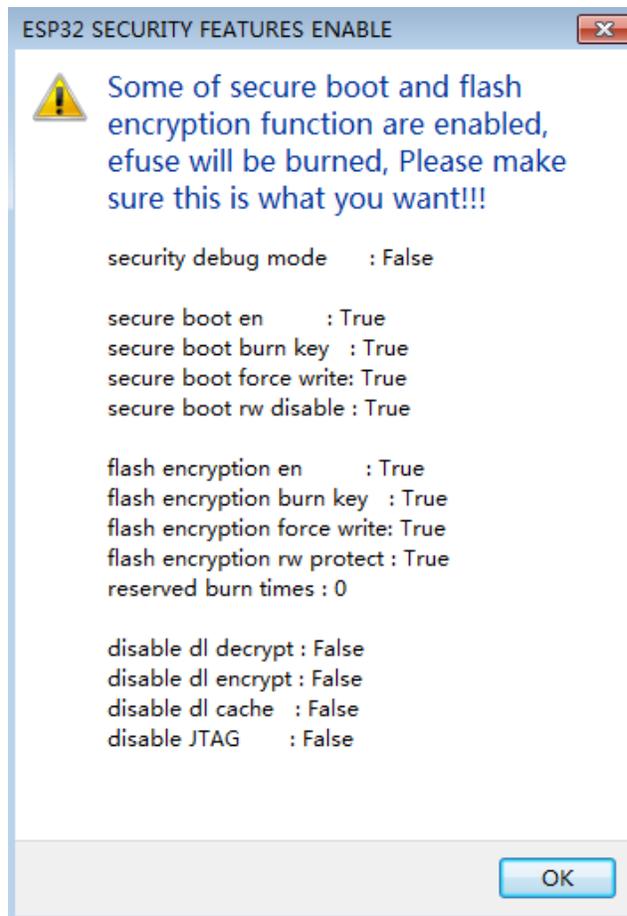
[AES KEY]
aes_key_en = False
burn_aes_key = False

[DISABLE FUNC]
jtag_disable = False
dl_encrypt_disable = False
dl_decrypt_disable = False
dl_cache_disable = False
```



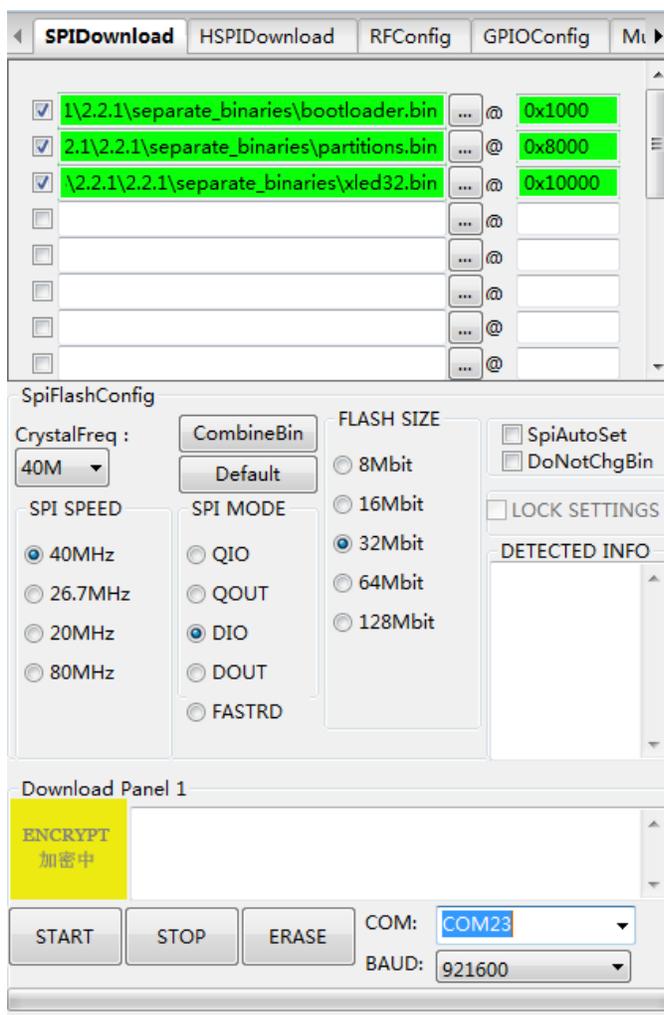
2. Burning process

- There will be a prompt message (shown below) when the tool is running. Check if the message is correct.



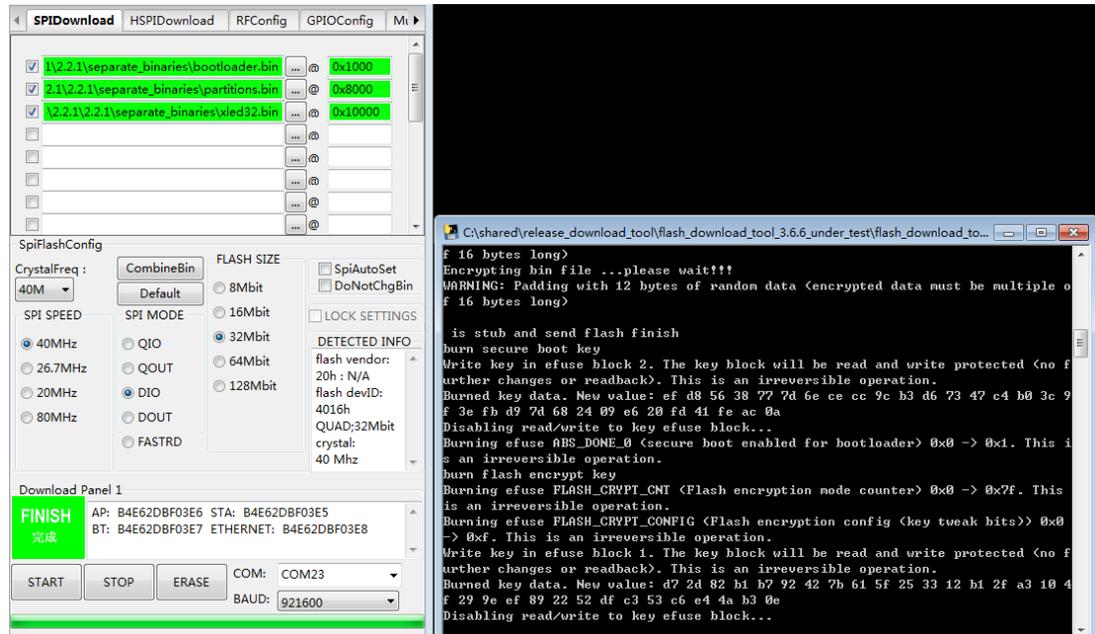


- Click **START** to begin downloading. The downloading process can be divided into two stages, namely the encryption subprocess that may take a while, and the after-encryption downloading subprocess.





- When firmware is successfully downloaded, the key-related information will be burned to the efuse of the chip. When it's done, a green **"FINISH /完成"** icon will appear in the lower left corner of the interface, as shown in the figure below.



Note:

Prior to downloading, the tool verifies the efuse information, so as to prevent re-downloading to and damaging the encrypted module.



4.2. ESP8266 Series Products

4.2.1. ESP-WROOM-02

1. Start the Flash Download Tool. The main interface will appear, as shown in Figure 4-7. Choose **ESP8266 DownloadTool** to access the interface for downloading firmware to an ESP8266 module.



Figure 4-7. ESP8266 DownloadTool



2. Provide information about the bin file to be downloaded and the address from which this bin will be downloaded. Also, configure the **CrystalFreq**, **SPI SPEED**, **SPI MODE**, **FLASH SIZE**, **COM** and **BAUD** settings, based on your actual case.

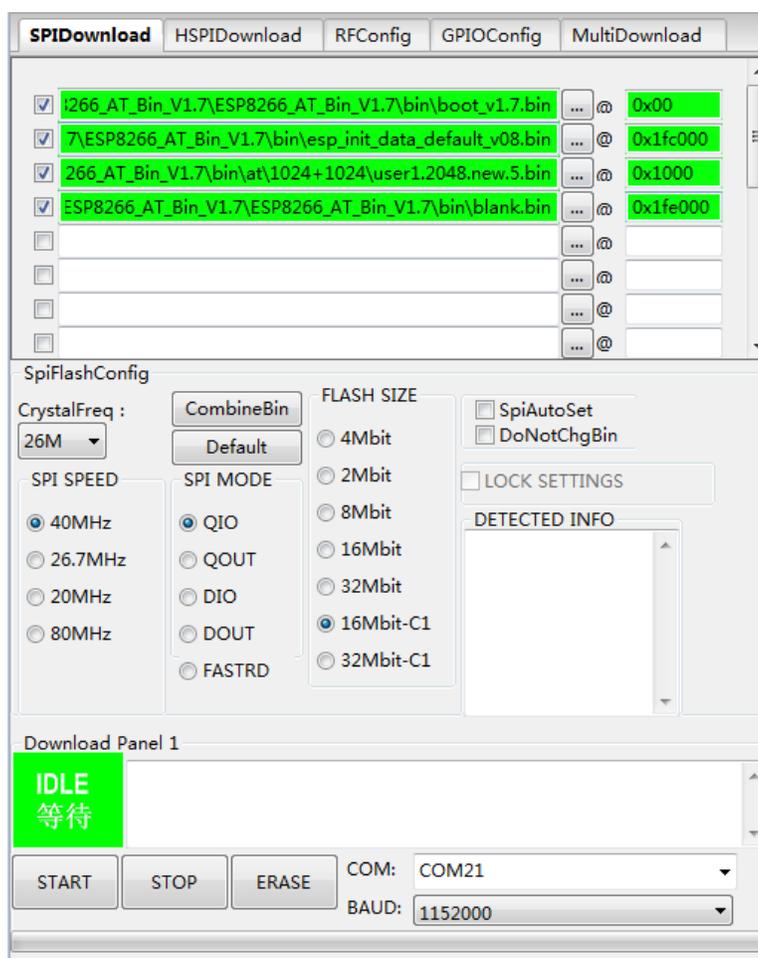


Figure 4-8. ESP8266 DownloadTool — SPIDownload

Note:

1. The parameters in the **SpiFlashConfig** block will be written to flash. The flash boot mode will be configured according to these parameters while powering up.
2. In this user guide, the bin files of AT firmware are used as the bin files to be downloaded.
3. Here, the **CrystalFreq** should be set to 26 M and the **SPI MODE** should be set to QIO or DIO based on your actual case. Otherwise, the downloaded firmware may not be running as expected.



3. Make sure the module is in download mode (In order to do this, open the serial port tool and check if "*ets Jan 8 2013,rst cause:1, boot mode:(1,2)*" is displayed.
4. Disconnect the serial port from other software programs, otherwise the downloading fails
5. Check the firmware to be downloaded in the **Download Path Config** area.
6. Configure the **SpiFlashConfig** block settings based on your actual case, as described in Section 3.1
7. Set the **COM** port and **BAUD** rate for downloading.
8. Click **START** to start the download process.
9. While downloading, the tool reads the flash information and MAC address of the product, as shown in Figure 4-9.

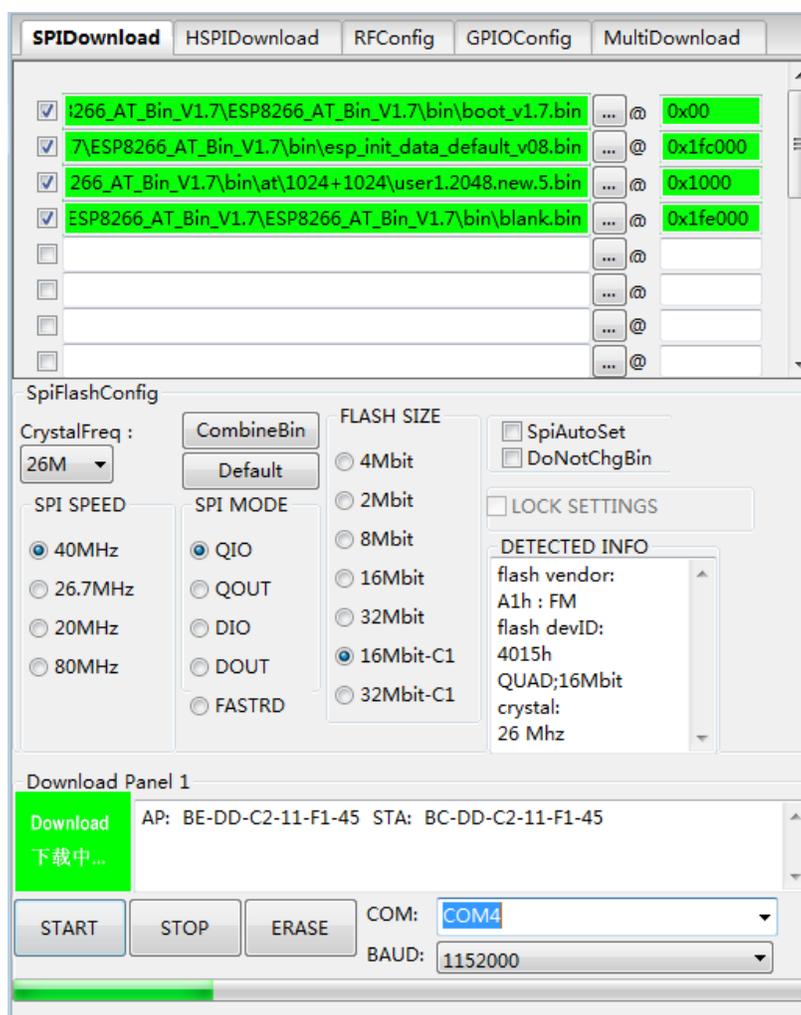


Figure 4-9. Downloading



10. When the downloading is completed, you can see a green **"FINISH /完成"** icon in the lower left corner of the interface.

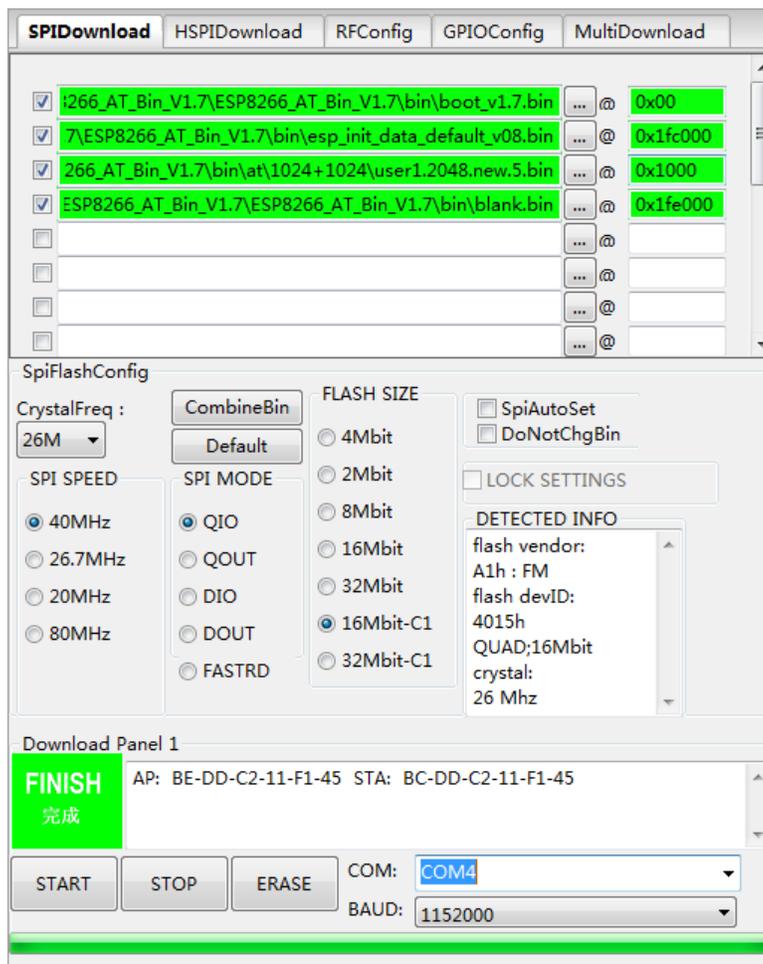


Figure 4-10. Finish downloading



11. Open the serial port tool, and power up the product again. The expected power-up log is shown in Figure 4-11. (The return message is garbled, because the baud rate during the powering-up is 74880, while AT commands are only able to receive the baud rate of 115200):

```
[21:13:57.235] jPY5dT9 5 q姪
A
一蟹姪
A倍囊T%jPY5dT9 5 q菘(便P` )道X)道PVPY!INIJa裔)媳跨q嗽蛸坳%菩滋镍媳籠I\YJ!* 晃%菘-y
A* 晃%菘55* 晃%菘)R5!* 晃%菘)Q
5h!uAA(便P`E)道X)道PVPY!INIJa涎)媳論q 蛸坳%菩滋镍媳籠I\YJ!姪
AZll|'JPY5dT9 5 Q健
A蛸鸵篤PJRA蛸奋PJRA蛸圓蛸 R@9E泥K2Q R5男P籤' @ @ @ @ @
@)@ @ @ @ @ @ @ @ @ @ @ @ @ @ @
@ @ @ @ @ JPY5dT9 5 q菘(5便P` )道X)道PVPY!INIJa裔)媳論q嗽蛸坳%
菩滋镍媳籠I\YJ!* %菘-y
A* %菘55* %菘)R5!* %菘)Q
5h!u1AA(便P`e)道X)道PVPY!INIJa裔)媳論q嗽蛸坳%菩滋镍媳籠I\YJ!@ @ @ @ @
@)@ @ @ @ @ @ @ @ @ @ @ @ @ @ @
@ @ @ @ @ @PJRA蛸峪嶺才RA蛸灶宜1 5 勳 才RA
一蝎R T1 5 媼 JPY5dT9 5 qJPY5dT9 5 q菘(便P` )道X)道PVPY!INIJa裔)媳論q嗽蛸坳%菩滋镍媳
物I\YJ!* 時%菘-y
A* 時%菘55* 時%菘)R5!* 踏%菘)Q
5h!uAA(便P` )道X)道PVPY!INIJa裔)媳論q嗽蛸坳%菩滋镍媳媼I\YJ!姪
A
一蟹姪
A倍囊JPY5dT9 5 qJPY5dT9 5 q菘(便Ph)道X)道PVPY!INIJa裔)媳論q 蛸坳%菩滋镍媳籠I\YJ!* 嫖%菘-y
A* 嫖9%菘55* 嫖%菘)R5!* 嫖%菘)Q
5H!uAA(便Ph)道X)道PVPY!INIJa裔)媳論q嗽蛸坳%菩滋镍媳籠I\YJ!姪
AZ囉\健
A蛸鸵篤PJRA蛸奋PJRA蛸圓蛸 R@9E泥K2Q R5男P況' @ @ @ @ @
@ @ @ @ @ @ @ @ @ @ @ @ @ @ @
@ @ @ @ @ @ @ @ @ @ @ @ @ @ @
@ @ @ @ @ @PJRA蛸峪嶺才RA蛸灶宜1 5 勳 才RA
蚧蝎R T1 5 媼 JPY5dT9 5 qJPY5dT9 5 q菘"(淳Ph)道X)道PVPY!INIJa裔)媳論q嗽蛸坳%菩滋镍媳籠
I\YJ!* 汝%菘-y
A* 汝%菘55* 汝%菘)R5!* 泐%菘)Q
5h!uAA(便PhE)道X)道PVPY!INIJa裔)媳論q嗽蛸坳%菩滋镍媳籠I\YJ!姪
```

Figure 4-11. The Expected Power-up log



4.2.2. ESP-WROOM-S2

The flash download steps for ESP-WROOM-S2 are similar to those of ESP-WROOM-02, except for the following items:

- IO15 should be connected to Ground before powering-up.
- After the module is powered up, IO15 should be set to floating and the power should stay on
- Here, the HSPIDownload tab should be selected, **CrystalFreq** should be set to 26 M and the **SPI MODE** should be set to **QIO** or **DIO** based on your actual case, as shown in Figure 4-12. Otherwise, the downloaded firmware may not be running as expected.

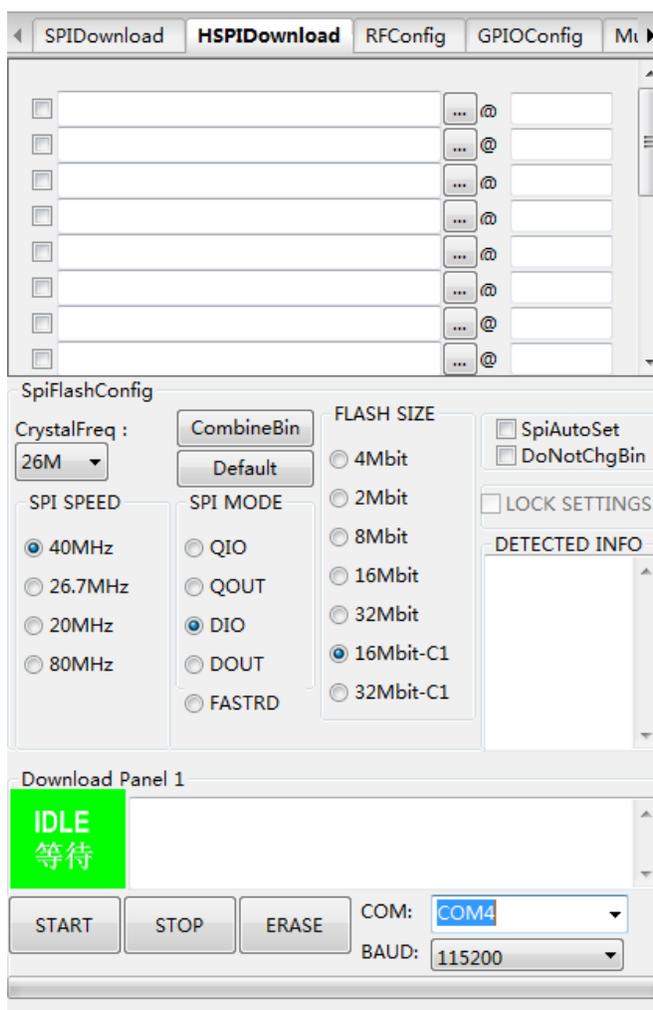


Figure 4-12. ESP8266 DownloadTool — HSPIDownload

- The subsequent steps are identical to those for ESP-WROOM-02. For details, see Section 4.2.1.

Note:

The firmware may not work properly if there are any configuration errors.



4.3. ESP8266 Series Products

1. Start the Flash Download Tool. The main interface appears, as shown in Figure 4-13. Choose **ESP8285 DownloadTool** to access the interface for downloading firmware to an ESP8285 module.



Figure 4-13. ESP8285 DownloadTool

Note:

For ESP8285 modules, the **FLASH SIZE** must be set to 8 Mbit and the **SPI MODE** must be configured to the **DOUT** mode. Please note that some of these options are auto-filled in the tool and cannot be modified).

2. The subsequent steps are identical to those for ESP-WROOM-02. For details, see Section 4.2.1.



5.

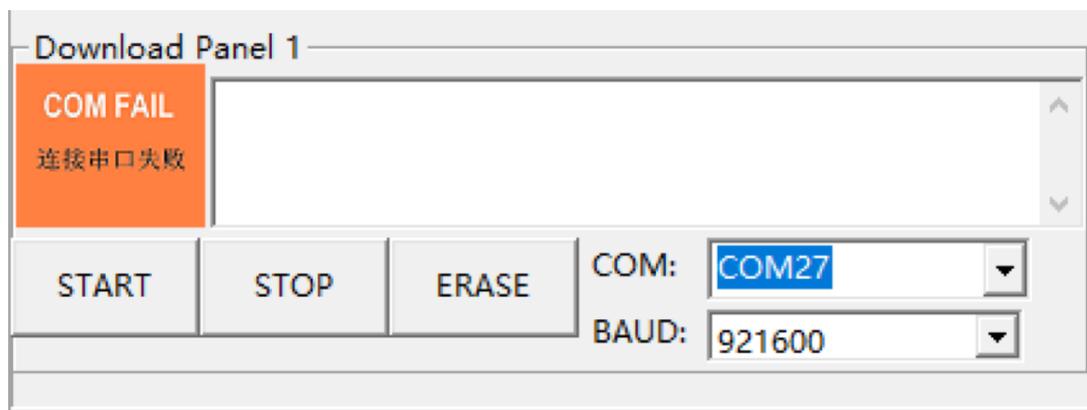
FAQs

5.1. COM Related Errors

1. I cannot find the serial port that I am using in the **COM** drop-down menu when I open the Flash Download Tool.

A: First go to the device manager and check if the serial port has been successfully installed. If not, check the driver for any possible issues.

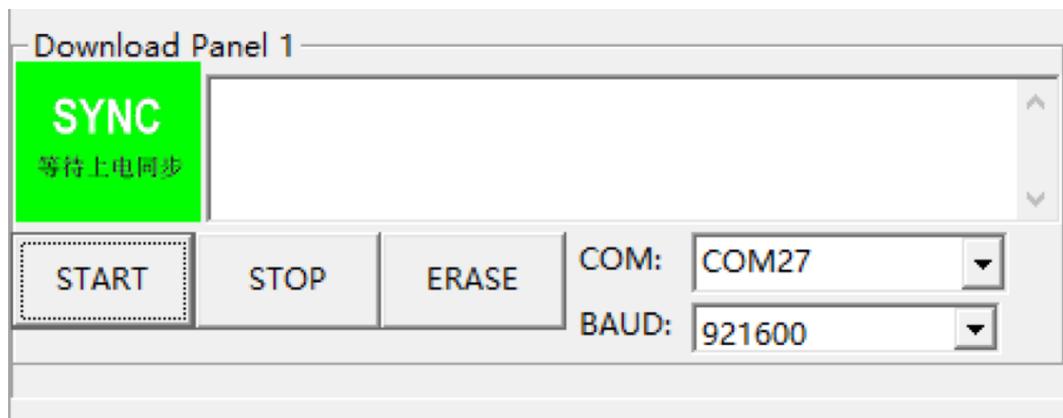
2. I get the "**COM FAIL/连接串口失败**" error message, as shown in the Figure below:



A: Firstly, make sure the selected COM is correct; Then, check if the COM is already occupied by another thread.

5.2. Synchronization Related Errors

1. The Flash Download Tool is stuck at the step shown in the figure below. How can I fix this?



A: This may happen for the reasons given below.

- Hardware:



- ▶ A jumper cap should be added to the USB to Serial Port test board.
- ▶ As discussed in Chapter 3, the module should be in download mode.
- ▶ The flash you are using is not supported by this tool.
- Software:
 - ▶ The module selected in the tool is not the one you are actually using.

5.3. Efuse Related Errors

1. I click the **START** button, and get the error shown in Figure 5-3.



A: You will get the **"ESP8266 Chip efuse check error esp_check_mac_and_efuse"** message when there are errors related to the eFuse. The possible causes are as follows:

- The eFuse is Ok, but the module selected in the tool is not the one that is actually being used. In this situation, please select the module type based on your actual case.
- There are problems with the eFuse of the module. In this case, please contact Espressif to obtain the required esptool.exe and operating instructions. Also, send the data read from eFuse to Espressif for further debugging.

5.4. Download Related Errors

1. Errors during downloading

A: Please check the following:

- The TX/RX of the module is not used by other software programs.
- The module flash size is no less than the size of firmware to be downloaded.
- If there is MD5 verification error, erase the entire flash and try downloading again.

5.5. Operation related error

1. The module crashes when powered on again after the firmware has been downloaded

A: If the downloaded firmware works fine, then please check the following:

- The module selected in the tool is not the one you are actually using.
- The selected flash boot mode is wrong.
- The selected flash download mode is wrong.



A. Appendix - Contents of the Flash Download Tool Folder

The main folder of the Flash Download Tool contains an executable file, function libraries, and a couple of subfolders, as shown in the figure below:

 configure	2019/1/9 13:16	文件夹	
 dl_temp	2019/1/9 13:16	文件夹	
 init_data	2019/1/9 13:16	文件夹	
 RESOURCE	2019/1/9 13:16	文件夹	
 secure	2019/3/5 16:22	文件夹	
 flash_download_tool.exe	2019/1/9 13:14	应用程序	13,472 KB

- **configure** folder: stores the download related configuration files.
- **dl_temp** folder: stores the download related temporary bin files.
- **init_data** folder: stores the initial RF values.
- **RESOURCE** folder: stores graphic images for the tool interface.
- **secure** folder: stores the intermediate files in the downloading process when encryption is enabled.
- **flash_download_tool.exe**: executable file of the Flash Download Tool.

 **Note:**

Please DO NOT modify the contents of this folder, unless it is absolutely necessary.



Espressif IoT Team
www.espressif.com

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. The Bluetooth logo is a registered trademark of Bluetooth SIG.

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

Copyright © 2019 Espressif Inc. All rights reserved.