

# eFuse ERROR Screenshot of ESP32

ESPflash\_download\_tool\_3.9.2\flash\_download\_tool\_3.9.2\flash\_download\_tool\_3.9.2.exe

```
test offset : 32768 0x8000
case ok
test offset : 1048576 0x100000
case ok
test offset : 65536 0x10000
case ok
test offset : 61440 0xF000
case ok
test offset : 131072 0x20000
case ok
test offset : 147456 0x24000
case ok
test offset : 233472 0x39000
case ok
test offset : 155648 0x26000
case ok
test offset : 163840 0x28000
case ok
test offset : 188416 0x2e000
case ok
test offset : 196608 0x30000
case ok
test offset : 135168 0x21000
case ok

uploading stub...
Running stub...
Stub running...
[2023-03-24 16:48:09.194][ESP8266Loader_spi[1]][espDownloader.py][line:594][ERROR]: ESP32 Chip efuse check error esp_check_mac_and_efuse.
```

SPIFlash Config

SPI SPEED	SPI MODE	DoNotChgBin	LOCK SETTING	CombineBin	Default	DETECTED INFO
<input checked="" type="radio"/> 40MHz	<input type="radio"/> QIO	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="button" value="CombineBin"/>	<input type="button" value="Default"/>	
<input type="radio"/> 26.7MHz	<input type="radio"/> QOUT					
<input type="radio"/> 20MHz	<input checked="" type="radio"/> DIO					
<input type="radio"/> 80MHz	<input type="radio"/> DOUT					
	<input type="radio"/> FASTRD					

Download Panel 1

**ERROR**  
错误

START STOP ERASE COM: COM6 BAUD: 115200

COM6

5-chip efuse check fail

Tip: Please confirm whether the chip type to be downloaded is correct and whether it is the official chip of espressif

OK

# eFuse log of MAC address: 9c:9c:1f:da:5f:74 (7737)

Welcome to minicom 2.7.1

OPTIONS: l18n

Compiled on Dec 23 2019, 02:06:26.

Port /dev/ttyUSB0, 18:12:40

Press CTRL-A Z for help on special keys

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:0x3fff00b0,len:7300

load:0x40078000,len:15588

load:0x40080400,len:3864

entry 0x40080650

W (66) boot.esp32: eFuse virtual mode is enabled. If Secure boot or Flash encry!

W (90) efuse: Loading virtual efuse blocks from real efuses

I (47) boot: ESP-IDF v5.0.1-dirty 2nd stage bootloader

I (47) boot: compile time 17:40:06

I (47) boot: chip revision: v1.0

I (51) boot.esp32: SPI Speed : 40MHz

I (55) boot.esp32: SPI Mode : DIO

I (60) boot.esp32: SPI Flash Size : 2MB

I (64) boot: Enabling RNG early entropy source...

I (70) boot: Partition Table:

I (73) boot: ##	Label	Usage	Type	ST	Offset	Length
I (81) boot: 0	nvs	WiFi data	01 02	00009000	00006000	
I (88) boot: 1	phy_init	RF data	01 01	0000f000	00001000	
I (96) boot: 2	factory	factory app	00 00	00010000	00100000	

I (103) boot: End of partition table

I (107) esp\_image: segment 0: paddr=00010020 vaddr=3f400020 size=093f8h ( 37880p

I (130) esp\_image: segment 1: paddr=00019420 vaddr=3ffb0000 size=01ebch ( 7868d

I (133) esp\_image: segment 2: paddr=0001b2e4 vaddr=40080000 size=04d34h ( 19764d

I (144) esp\_image: segment 3: paddr=00020020 vaddr=400d0020 size=16b2ch ( 92972p

I (177) esp\_image: segment 4: paddr=00036b54 vaddr=40084d34 size=06ae0h ( 27360d

I (195) boot: Loaded app from partition at offset 0x10000

I (195) boot: Disabling RNG early entropy source...

I (206) cpu\_start: Pro cpu up.

I (207) cpu\_start: Starting app cpu, entry point is 0x4008104c

I (0) cpu\_start: App cpu up.

I (221) cpu\_start: Pro cpu start user code

I (221) cpu\_start: cpu freq: 160000000 Hz

I (221) cpu\_start: Application information:

I (226) cpu\_start: Project name: efuse

I (230) cpu\_start: App version: 1

I (235) cpu\_start: Compile time: Mar 25 2023 17:41:33  
I (241) cpu\_start: ELF file SHA256: d71fd671ff979164...  
I (247) cpu\_start: ESP-IDF: v5.0.1-dirty  
I (252) cpu\_start: Min chip rev: v0.0  
I (257) cpu\_start: Max chip rev: v3.99  
I (262) cpu\_start: Chip rev: v1.0  
I (267) heap\_init: Initializing. RAM available for dynamic allocation:  
I (274) heap\_init: At 3FFAE6E0 len 00001920 (6 KiB): DRAM  
I (280) heap\_init: At 3FFB28C0 len 0002D740 (181 KiB): DRAM  
I (286) heap\_init: At 3FFE0440 len 00003AE0 (14 KiB): D/IRAM  
I (292) heap\_init: At 3FFE4350 len 0001BCB0 (111 KiB): D/IRAM  
I (299) heap\_init: At 4008B814 len 000147EC (81 KiB): IRAM  
I (306) spi\_flash: detected chip: generic  
I (310) spi\_flash: flash io: dio  
W (314) spi\_flash: Detected size(4096k) larger than the size in the binary image.  
W (327) cpu\_start: eFuse virtual mode is enabled. If Secure boot or Flash encryption!  
W (341) efuse: Loading virtual efuse blocks from real efuses  
I (348) cpu\_start: Starting scheduler on PRO CPU.  
I (0) cpu\_start: Starting scheduler on APP CPU.  
I (358) example: Start eFuse example  
I (358) example: Coding Scheme NONE  
I (368) example: read efuse fields  
I (368) example: 1. read MAC address: 9c:9c:1f:da:5f:74  
I (378) example: 2. read secure\_version: 0  
I (378) example: 3. read custom fields  
I (388) example: module\_version = 0  
I (388) example: device\_role = None  
I (398) example: setting\_1 = 16  
I (398) example: setting\_2 = 2  
I (398) example: custom\_secure\_version = 1  
W (408) example: This example does not burn any efuse in reality only virtually  
W (418) example: Write operations in efuse fields are performed virtually  
I (418) example: write custom efuse fields  
W (428) efuse: Virtual efuses enabled: Not really burning eFuses  
W (438) efuse: Virtual efuses enabled: Not really burning eFuses  
W (438) efuse: Virtual efuses enabled: Not really burning eFuses  
W (448) efuse: Virtual efuses enabled: Not really burning eFuses  
W (458) efuse: Virtual efuses enabled: Not really burning eFuses  
I (458) example: module\_version = 1  
I (468) example: device\_role = Slave  
I (468) example: setting\_1 = 19  
I (468) example: setting\_2 = 6  
I (478) example: custom\_secure\_version = 6  
I (478) example: Done

# eFuse log of MAC address: c8:f0:9e:f2:af:24 (4291)

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:0x3fff00b0,len:7300
load:0x40078000,len:15588
load:0x40080400,len:3864
entry 0x40080650
W (66) boot.esp32: eFuse virtual mode is enabled. If Secure boot or Flash encry!
W (90) efuse: Loading virtual efuse blocks from real efuses
I (47) boot: ESP-IDF v5.0.1-dirty 2nd stage bootloader
I (47) boot: compile time 17:40:06
I (47) boot: chip revision: v1.0
I (51) boot.esp32: SPI Speed      : 40MHz
I (55) boot.esp32: SPI Mode      : DIO
I (60) boot.esp32: SPI Flash Size : 2MB
I (64) boot: Enabling RNG early entropy source...
I (70) boot: Partition Table:
I (73) boot: ## Label        Usage        Type ST Offset  Length
I (81) boot: 0 nvs           WiFi data    01 02 00009000 00006000
I (88) boot: 1 phy_init      RF data      01 01 0000f000 00001000
I (95) boot: 2 factory       factory app  00 00 00010000 00100000
I (103) boot: End of partition table
I (107) esp_image: segment 0: paddr=00010020 vaddr=3f400020 size=093f8h ( 37880p
I (129) esp_image: segment 1: paddr=00019420 vaddr=3ffb0000 size=01ebch ( 7868d
I (133) esp_image: segment 2: paddr=0001b2e4 vaddr=40080000 size=04d34h ( 19764d
I (143) esp_image: segment 3: paddr=00020020 vaddr=400d0020 size=16b2ch ( 92972p
I (177) esp_image: segment 4: paddr=00036b54 vaddr=40084d34 size=06ae0h ( 27360d
I (195) boot: Loaded app from partition at offset 0x10000
I (195) boot: Disabling RNG early entropy source...
I (206) cpu_start: Pro cpu up.
I (206) cpu_start: Starting app cpu, entry point is 0x4008104c
I (0) cpu_start: App cpu up.
I (221) cpu_start: Pro cpu start user code
I (221) cpu_start: cpu freq: 160000000 Hz
I (221) cpu_start: Application information:
I (225) cpu_start: Project name:   efuse
I (230) cpu_start: App version:   1
I (235) cpu_start: Compile time:   Mar 25 2023 17:41:33
I (241) cpu_start: ELF file SHA256: d71fd671ff979164...
I (247) cpu_start: ESP-IDF:      v5.0.1-dirty
I (252) cpu_start: Min chip rev:   v0.0
I (257) cpu_start: Max chip rev:   v3.99
I (262) cpu_start: Chip rev:      v1.0
```

I (267) heap\_init: Initializing. RAM available for dynamic allocation:  
I (274) heap\_init: At 3FFAE6E0 len 00001920 (6 KiB): DRAM  
I (280) heap\_init: At 3FFB28C0 len 0002D740 (181 KiB): DRAM  
I (286) heap\_init: At 3FFE0440 len 00003AE0 (14 KiB): D/IRAM  
I (292) heap\_init: At 3FFE4350 len 0001BCB0 (111 KiB): D/IRAM  
I (299) heap\_init: At 4008B814 len 000147EC (81 KiB): IRAM  
I (306) spi\_flash: detected chip: generic  
I (310) spi\_flash: flash io: dio  
W (313) spi\_flash: Detected size(4096k) larger than the size in the binary imag.  
W (326) cpu\_start: eFuse virtual mode is enabled. If Secure boot or Flash encry!  
W (341) efuse: Loading virtual efuse blocks from real efuses  
I (348) cpu\_start: Starting scheduler on PRO CPU.  
I (0) cpu\_start: Starting scheduler on APP CPU.  
I (358) example: Start eFuse example  
I (358) example: Coding Scheme NONE  
I (368) example: read efuse fields  
I (368) example: 1. read MAC address: c8:f0:9e:f2:af:24  
I (378) example: 2. read secure\_version: 1  
I (378) example: 3. read custom fields  
I (388) example: module\_version = 0  
I (388) example: device\_role = None  
I (398) example: setting\_1 = 0  
I (398) example: setting\_2 = 0  
I (398) example: custom\_secure\_version = 0  
W (408) example: This example does not burn any efuse in reality only virtually  
W (418) example: Write operations in efuse fields are performed virtually  
I (418) example: write custom efuse fields  
W (428) efuse: Virtual efuses enabled: Not really burning eFuses  
W (438) efuse: Virtual efuses enabled: Not really burning eFuses  
W (438) efuse: Virtual efuses enabled: Not really burning eFuses  
W (448) efuse: Virtual efuses enabled: Not really burning eFuses  
W (458) efuse: Virtual efuses enabled: Not really burning eFuses  
I (458) example: module\_version = 1  
I (468) example: device\_role = Slave  
I (468) example: setting\_1 = 3  
I (468) example: setting\_2 = 4  
I (478) example: custom\_secure\_version = 5  
I (478) example: Done