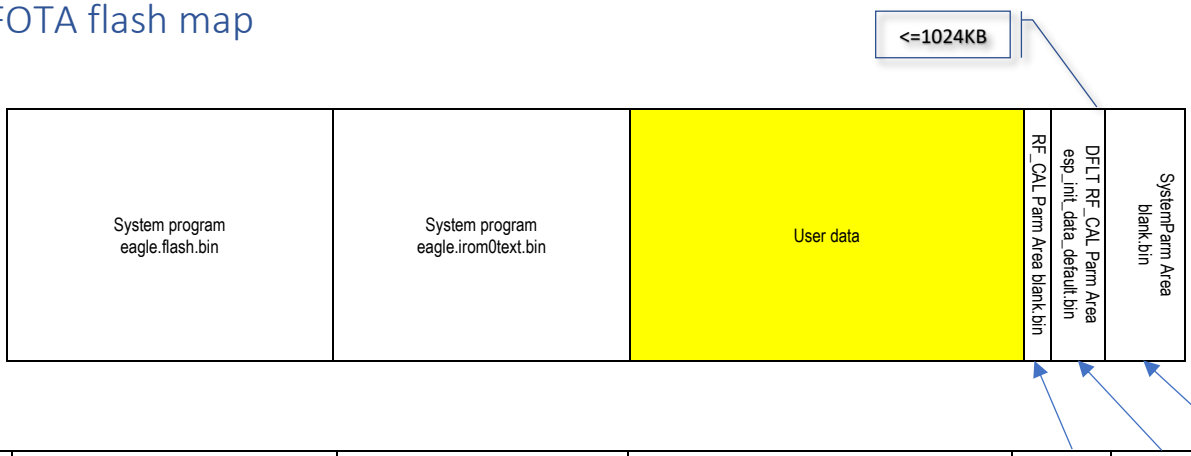
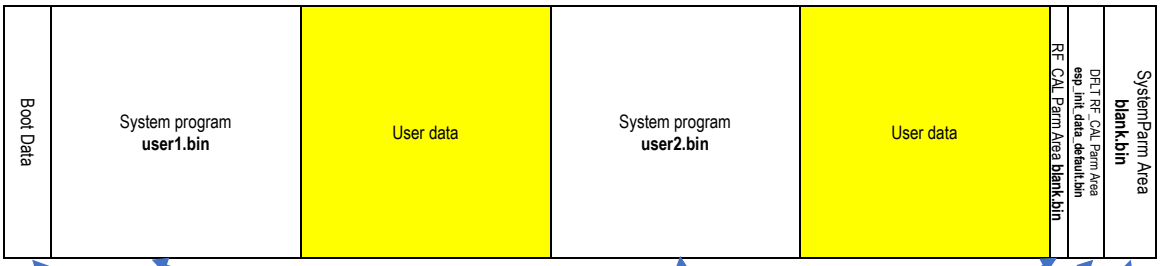
NonFOTA flash map



Flash						
512	<=64KB 0x00000	<=368KB 0x10000	>=60KB	4KB 0x7B000	4KB 0x7C000	12KB 0x7E000
1024	<=64KB 0x00000	<=752KB 0x10000	>=176KB	4KB 0xFB000	4KB 0xFC000	12KB 0xFE000
2048	<=64KB 0x00000	<=768KB 0x10000	>=176KB	4KB 0x1FB000	4KB 0x1FC000	12KB 0x1FE000
4096	<=64KB 0x00000	<=768KB 0x10000	>=176KB	4KB 0x3FB000	4KB 0x3FC000	12KB 0x3FE000
8192	<=64KB 0x00000	<=768KB 0x10000	>=176KB	4KB 0x7FB000	4KB 0x7FC000	12KB 0x7FE000
16384	<=64KB 0x00000	<=768KB 0x10000	>=176KB	4KB 0xFFB000	4KB 0xFFC000	12KB 0xFFE000

Use esptool to load 8 and 16 MB flash

FOTA flash map



Flash							User data
512	4KB 0x00000	<=232KB 0x01000	<=232KB 0x41000	4KB 0x7B000	4KB 0x7C000	12KB 0x7E000	>=0
1024	4KB 0x00000	<=488KB 0x01000	<=488KB 0x81000	4KB 0xFB000	4KB 0xFC000	12KB 0xFE000	>=0
2048 Part1 = 512KB	4KB 0x00000	<=488KB 0x01000	<=488KB 0x81000	4KB 0x1FB000	4KB 0x1FC000	12KB 0x1FE000	>=1024KB
2048 Part1 = 1024KB	4KB 0x00000	<=1000KB 0x01000	<=1000KB 0x101000	4KB 0x1FB000	4KB 0x1FC000	12KB 0x1FE000	>=0
4096 Part1 = 512KB	4KB 0x00000	<=488KB 0x01000	<=488KB 0x81000	4KB 0x3FB000	4KB 0x3FC000	12KB 0x3FE000	>=3072KB
4096 Part1 = 1024KB	4KB 0x00000	<=1000KB 0x01000	<=1000KB 0x101000	4KB 0x3FB000	4KB 0x3FC000	12KB 0x3FE000	>=2048KB
8192 Part1 = 1024KB	4KB 0x00000	<=1000KB 0x01000	<=1000KB 0x101000	4KB 0x7FB000	4KB 0x7FC000	12KB 0x7FE000	>=6144KB
16384 Part1 = 1024KB	4KB 0x00000	<=1000KB 0x01000	<=1000KB 0x101000	4KB 0xFFB000	4KB 0xFFC000	12KB 0xFFE000	>=14336KB

Use esptool to load 8 and 16 MB flash

Please note – presented above memory sizes and location addresses are taken from official Espressif documentation but they do not match each other. SystemParm Area and DFLT RF_CAL Parm Area have 8KB and RF_CAL Parm Area has 4KB. Much better source of these values are examples from SDK – see below.

Following sizes and addresses are based on code from SDK examples - FOTA

```
#elif (SPI_FLASH_SIZE_MAP == 6)
#define SYSTEM_PARTITION_OTA_SIZE
#define SYSTEM_PARTITION_OTA_2_ADDR
#define SYSTEM_PARTITION_RF_CAL_ADDR
#define SYSTEM_PARTITION_PHY_DATA_ADDR
#define SYSTEM_PARTITION_SYSTEM_PARAMETER_ADDR
#else
#error "The flash map is not supported"
#endif

static const partition_item_t at_partition_table[] = {
    { SYSTEM_PARTITION_BOOTLOADER,            0x0,
    { SYSTEM_PARTITION_OTA_1,                 0x1000,
    { SYSTEM_PARTITION_OTA_2,
    { SYSTEM_PARTITION_RF_CAL,
    { SYSTEM_PARTITION_PHY_DATA,
    { SYSTEM_PARTITION_SYSTEM_PARAMETER,
};
```

Defined in project

Bootloader always first with length 0x1000

OTA1 after bootloader

```
0x6A000
0x101000
0x3fb000
0x3fc000
0x3fd000
```

Size: 4KB

```
0x1000},
SYSTEM_PARTITION_OTA_SIZE},
SYSTEM_PARTITION_OTA_SIZE},
0x1000},
0x1000},
0x3000},
```

Size: 12KB

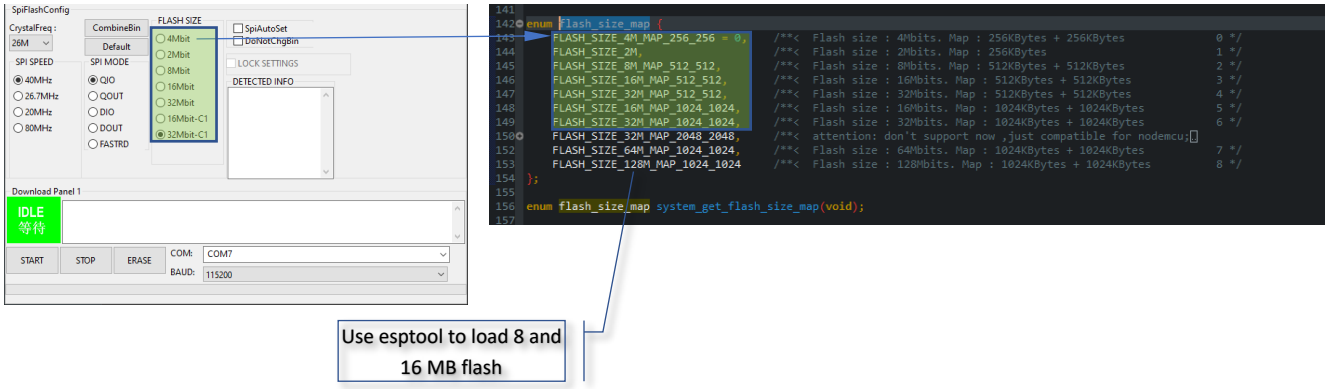
```
SYSTEM_PARTITION_OTA_2_ADDR,
SYSTEM_PARTITION_RF_CAL_ADDR,
SYSTEM_PARTITION_PHY_DATA_ADDR,
SYSTEM_PARTITION_SYSTEM_PARAMETER_ADDR,
```

bootloader	OTA1	OTA2	RF_CAL	PHY_DATA	SYSTEM_PARAMETER
0x1000	0x6A000	0x6A000	0x1000	0x1000	0x3000

0x000000		0x1000		0x101000		0x3FB000		0x3FC000		0x3FD000		0x400000

Original ESP8266_NONOS_SDK Code/Documentation/Licence:
https://github.com/espressif/ESP8266_NONOS_SDK

Why download tool can't be used to load firmware to 8 and 16MB flash



Quick cheat-sheet

```
1 MEGABIT    - 128  kByte  - 131072  Byte  - 1024 kbit  - 1048576  bit
1 MEGABYTE   - 1024 kByte  - 1048576 Byte  - 8192 kbit  - 8388608  bit
1 KILOBIT    - 1/8  kByte  - 128     Byte  - 1    kbit  - 1024     bit
1 KILOBYTE   - 1    kByte  - 1024    Byte  - 8    kbit  - 8192     bit
```

- **System Program:** This area stores the firmware necessary for the system to run.
- **User Data:** If the system data do not take up all the flash memory, the remaining area can be used to store user data. It is recommended that the user reserve at least 12 KB in the user data area to store user parameters.
- **RF_CAL Parameter:** The system automatically stores the calibrated RF parameters in this area.
- Default RF Parameter: Download esp_int_data_default.bin in this area to store the default RF parameters.
- **System Parameter Area:** This area stores the system parameters.
- **Boot Data:** It is located in Partition 1 of the FOTA firmware, and stores boot data.

Predefined partitions / User-defined partitions

user_interface.h

```
typedef enum {
    SYSTEM_PARTITION_INVALID = 0,
    SYSTEM_PARTITION_BOOTLOADER,            /* user can't modify this partition address, but can modify size */
    SYSTEM_PARTITION_OTA_1,                 /* user can't modify this partition address, but can modify size */
    SYSTEM_PARTITION_OTA_2,                 /* user can't modify this partition address, but can modify size */
    SYSTEM_PARTITION_RF_CAL,                /* user must define this partition */
    SYSTEM_PARTITION_PHY_DATA,              /* user must define this partition */
    SYSTEM_PARTITION_SYSTEM_PARAMETER,      /* user must define this partition */
    SYSTEM_PARTITION_AT_PARAMETER,
    SYSTEM_PARTITION_SSL_CLIENT_CERT_PRIVKEY,
    SYSTEM_PARTITION_SSL_CLIENT_CA,
    SYSTEM_PARTITION_SSL_SERVER_CERT_PRIVKEY,
    SYSTEM_PARTITION_SSL_SERVER_CA,
    SYSTEM_PARTITION_WPA2_ENTERPRISE_CERT_PRIVKEY,
    SYSTEM_PARTITION_WPA2_ENTERPRISE_CA,

    SYSTEM_PARTITION_CUSTOMER_BEGIN = 100,  /* user can define partition after here */
        …
        …
        …
    SYSTEM_PARTITION_MAX
} partition_type_t;
```

Used for enctypted connection

Used for WPA Enterprise connection

Used for user partitions