

Gateway OTA

一、给网关本身 OTA

1. 记得烧录 `ota_boot.bin`；如果忘记烧录，在 OTA 开始的时候，gateway 指示灯会一直闪烁，并且不会回复任何 UART 信息。
2. OTA 数据结构：E0 19 00 + par(参考以下 `ble_ota_data_t`) + 串口 crc

```
typedef struct{
    u8 ota_type;    // 0 表示给网关本身进行升级，1 表示给开关进行升级
    u16 index;      // 小端对齐
    u8 data[16];
    u16 crc;
}ble_ota_data_t;
```

即：

ID + LEN + MAC + OTA_TYPE + INDEX(2byte, 小端) + ota data(16byte) + crc_ota_data(2byte) + uart crc

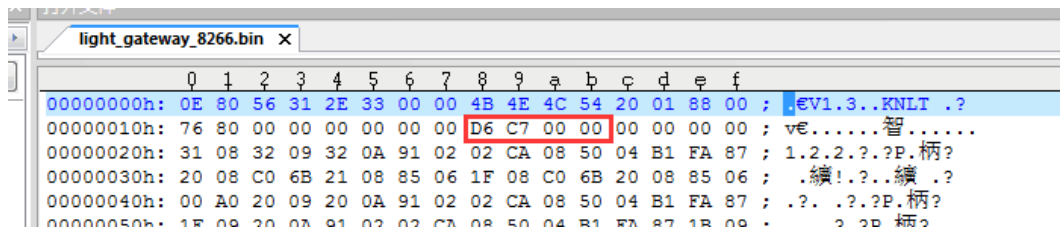
举例如下：

e0 19 00 00 02 00 31 08 32 09 32 0a 91 02 02 ca 08 50 04 b1 fa 87 8c 26 4a E0 19 00

红色字体“02 00”表示 index 是 2，小端对齐。

红色字体“E0 19”表示 `ble_ota_data_t` 里面的 crc 的值，由附录的 `crc16()` 函数算出来
`crc16(&ble_ota_data_t->index, 18)`

3. Firmware 的长度可以从 bin 文件的 0x18--0x1b 字段获取，比如：



表示文件长度是：0x0000C7D6 (51158 byte)

4. Firmware 数据的最后一个包，不足 16 个时，补 0xff
5. 所有的 firmware 数据发送完成后，发送 OTA 结束命令，格式如下：

ID + LEN + MAC + 串口 crc

即：e1 04 00 e5

6. 8266 gateway 收到 OTA 结束命令后，开始软重启，并启用新的 firmware，最后发送 OTA 结果的命令给主机，格式如下：

ID + LEN + MAC + result + 串口 crc

其中 result: 0 表示成功，1 表示失败。

即：

E2 05 00 00 E7 表示成功

E2 05 00 01 E8 表示失败

7.

附录： crc 函数

```
unsigned short crc16 (unsigned char *pD, int len)
```

```
{
```

```
    static unsigned short poly[2]={0, 0xa001};           //0x8005 <==> 0xa001
```

```
    unsigned short crc = 0xffff;
```

```
    int i,j;
```

```
    for(j=len; j>0; j--)
```

```
    {
```

```
        unsigned char ds = *pD++;
```

```
        for(i=0; i<8; i++)
```

```
        {
```

```
            crc = (crc >> 1) ^ poly[(crc ^ ds ) & 1];
```

```
            ds = ds >> 1;
```

```
        }
```

```
    }
```

```
    return crc;
```

```
}
```