**B6x升级协议说明**

Flash 256KB存储说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Flash地址** | | **下位机(蓝牙模块)** | | Pgidx\_F |
| 0x1800\_0000 | | magicCode(4B) | Flash Jump(0xAA55A004) | 0x0000 |
| .  .  .  .  .  . | 1800\_0004 | codeLen(4B) | 程序长度 |
| 1800\_0008 | codeAddr(4B) | 程序起始地址(Code1) |
| 1800\_000C | flashAddr(4B) | flash运行地址(Code1 或0xFFFFFFFF) |
| 1800\_0F00 | trimData(256B) | 校准值数据, 此页写保护 | 0x000F |
| 1800\_1000 | userData(12KB) | 用户数据起始地址 | 0x0010 |
| 1800\_4000 | Code1(112KB) | 下位机代码起始地址 | 0x0040 |
|  | 1802\_0000 | Code2(128KB) | OTA升级代码备份地址 | 0x0200 |
| 0x1803\_FFFF | | 256KB Flash结束地址 | | 0x03FF |

注:Flash大小根据实际封装芯片确定。

以下数据均为小端模式：

1. **BOOT UART (PA06\_TX/PA07\_RX)**

串口协议可实现对芯片的烧录功能。需进入Uart Boot，方式为：

1) 进入Boot 模式方式1

1. 上电后，同时拉低Uart 引脚:Tx(PA06)/Rx(PA07)和 RST 复位引脚;

2. 15ms 后，先恢复RST 复位引脚;

3. 20ms 后，再恢复Uart引脚:

4. Boot Uart发送”B9”，需要回复”9B”.(波特率115200,停止位1,数据位:8,校验位:无)

5. 100ms 后，未收到回复，退出进入Boot模式.

2) 进入Boot模式方式2 (nRST引脚复用为IO时, 需完全掉电)

1. 上电前，拉低 Uart 引脚:Tx(PA06)/Rx(PA07);

2. 上电 20ms 后，恢复 Uart 引脚;

3. Boot Uart发送”B9”，需要回复”9B”.(波特率115200,停止位1,数据位:8,校验位:无)

4. 100ms 后，未收到回复，退出进入Boot模式.

Boot进入后:

测试指令:5A 2A 00 00

响应指令:A5 2A 01 00 A3

* 1. **芯片Uart Boot协议**

进入Boot后，需将驱动程序(chipset.bin)加载到Sram中运行，才能将固件烧录到Flash中，此时Uart支持一下协议：

指令说明：

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Uart Boot串口数据格式(0x)** | | | | | | | | | | | **备注** |
| 主机 | Byte0 | Byte1 | Byte2 | Byte3 | Byte4 | Byte5 | Byte6 | Byte7 | Byte8 | Byte(n) | 字节位 |
| 5A | 2A | 00 | 00 | NULL | | | | | | 同步指令 |
| 5A | 08 | Len | | Addr\_L | | | | Data | CRC | 加载指令 |
| 5A | 0B | Len | | Baud | | | | NULL | CRC | 速率指令 |
| 5A | 13 | Len | | Addr\_S | | | | NULL | CRC | 跳转指令 |
|  | | | | | | | | | | | |
| 从机 | A5 | 2A | 00 | 01 | A3 | NULL | | | | | 同步响应 |
| A5 | 88 | 00 | 01 | Status | NULL | | | | | 加载响应 |
| A5 | 8B | 00 | 01 | Status | NULL | | | | | 速率响应 |
| A5 | 93 | 00 | 01 | Status | NULL | | | | | 跳转响应 |

字段说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字节位** | **字段** | **说明** | **备注** | |
| Byte0 | 5A | 下发指令头部 | 每个下发指令数据包都会以它开头(**下同**) | |
| A5 | 响应指令头部 | 每个响应指令数据包都会以它开头(**下同**) | |
| Byte1 | 2A | 同步字 | 用于主机、下位机握手(**下同**) | |
| 08 | 加载字 | 用于将驱动程序(chipset.bin)加载到Sram中 | |
| 0B | 速率字 | 用于设置串口波特率 | |
| 13 | 跳转字 | 用于跳转指定位置执行代码 | |
| Byte2 | Len | 长度 | 字节位Byte3后面数据的总长度(含CRC) | |
| Byte3 |
| Byte4 | Addr\_L(4B) | 加载地址 | 每帧数据在Sram中的位置，一般起始地址：0x20003600 | |
| Baud(4B) | 波特率 | 以115200为单位，最大8倍：921600 | |
| Addr\_S(4B) | 跳转地址 | 一般为加载地址的起始地址 | |
| A3 | 特殊响应 | Boot模式下的特殊同步指令响应状态(Status) | |
| Status(1B) | 状态 | 00:PT\_OK | 指令执行正常 |
| A0:PT\_ERR\_CRC | CRC校验错误 |
| A1:PT\_ERR\_LEN | 指令长度错误 |
| A2:PT\_ERR\_CODE | 指令错误 |
| A3:PT\_ERR\_HEAD | 指令头部错误 |
| A4:PT\_ERR\_TIMEOUT | 指令接收超时 |
| A4:PT\_ERR\_VERIFY | 下位机校验Flash失败 |
| A4:PT\_ERR\_STATUS | 下位机状态出错 |
| NULL(0B) | 空白 | 无数据(**下同**) | |
| Byte8 | Data(xB) | 固件数据 | 写入从机Sram中的驱动程序，x可根据Len等计算 | |
| Byte(n) | CRC | 校验码 | Byte4~Byte(n-1)的数据校验码 | |

* 1. **驱动程序串口协议**

加载驱动程序，并成功跳转运行后，此时Uart支持以下协议：

指令说明：

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Uart 串口数据格式(0x)** | | | | | | | | | | | **备注** |
| 主机 | Byte0 | Byte1 | Byte2 | Byte3 | Byte4 | Byte5 | Byte6 | Byte7 | Byte8 | Byte(n) | 字节位 |
| 5A | 2A | 00 | 00 | NULL | | | | | | 同步指令 |
| 5A | 32 | Len | | Mode | Mcnt | Maps | | | CRC | 擦除指令 |
| 5A | 34 | Len | | Pgidx\_F | | Data | | | CRC | 下载指令 |
| 5A | 36 | Len | | Addr\_M | | | | NULL | CRC | 修改指令 |
|  | | | | | | | | | | | |
| 从机 | A5 | 2A | 00 | 01 | Status | NULL | | | | | 同步响应 |
| A5 | 32 | 00 | 01 | Status | NULL | | | | | 擦除响应 |
| A5 | 34 | 00 | 01 | Status | NULL | | | | | 下载响应 |
| A5 | 36 | 00 | 01 | Status | NULL | | | | | 修改响应 |

字段说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字节位** | **字段** | **说明** | **备注** | |
| Byte1 | 32 | 擦除字 | 用于擦除从机Flash中的数据 | |
| 34 | 下载字 | 用于将固件写入从机Flash中 | |
| 36 | 修改字 | 用于修改Flash中指定位置的数据(MAC，名称等) | |
| Byte2 | Len | 长度 | **同上** | |
| Byte3 |
| Byte4 | Mode(1B) | 擦除模式 | 0:FLASH\_PAGE | 页擦除(256B)(默认) |
| 1:FLASH\_SECTOR | 扇擦除(4KB) |
| 2:FLASH\_CHIP | 全片擦除(256KB) |
| 3:FLASH\_BLOCK32 | 块擦除 1(32KB) |
| 4:FLASH\_BLOCK64 | 块擦除 2(64KB) |
| Pgidx\_F(2B) | 页编号 | Flash中，每256字为一页，首页Pgidx为0 | |
| Addr\_M(4B) | 修改地址 | 需要修改Flash中数据的0x18xxxxxx地址 | |
| Status(1B) | 状态 | **同上** | |
| Byte5 | mcnt(1B) | 擦除块总数 | 擦除非连续地址区的值，如：APROM 与 DATA | |
| Byte6 | \*maps(mcnt\*4B) | 擦除块表 | 存储扇区或者页的起始索引与总量 | |
| Data(xB) | 固件数据 | 写入从机 Flash 中的数据，x 可根据 Len 等计算 | |
| Byte... |
| Byte(n) | CRC | 校验码 | **同上** | |

1. **空中(OTA)升级协议**

[服务UUID] 0xFF50

[通知UUID] 0xFF51

[写入UUID] 0xFF52

空中数据接口协议, 同上”1.2驱动程序串口协议”。Flash小于128KB, 无法通过OTA升级.

1. **CRC校验方式**

#define PKT\_CRC\_SEED (0xFF)

uint8\_t pkt\_crc8(uint8\_t \*buff, uint16\_t len)

{

uint16\_t i;

uint8\_t crc = PKT\_CRC\_SEED;

for (i = 0; i < len; i++)

{

crc ^= buff[i];

}

return crc;

}