UHF Hand Reader 读写器

用户手册V1.2

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 修订日期 | 版本 | 摘要 | 涉及内容 | 修订人 | 备注 |
| 20240819 | V1.2 | 增加指令 | 新添H103，H104指令 | 董玉龙 |  |

目录

[1对接通讯协议 2](#_Toc692220314)

[1.1 串口通讯参数 2](#_Toc1147051578)

[1.2 数据帧格式 2](#_Toc1837356589)

[2 命令帧集 4](#_Toc1967546439)

[2.1 命令列表 4](#_Toc364387210)

[2.2 模块通用控制命令 5](#_Toc646679682)

[2.2.1 RFM\_MODULE\_INT（初始化设备） 5](#_Toc1163403882)

[2.2.2 RFM\_REBOOT（设备恢复出厂设置） 5](#_Toc1642371726)

[2.2.3 RFM\_GET\_DEVICEINFO（获取模块信息） 5](#_Toc1319990529)

[2.2.4 RFM\_SET\_ALL\_PARAM (设置所有可配置参数) 7](#_Toc1438468866)

[2.2.5 RFM\_GET\_ALL\_PARAM (读取所有可配置参数) 1](#_Toc1618424549)1

[2.2.6 RFM\_SET\_GET\_S-PERMISSION\_PARAM (设置读取权限) 12](#_Toc1699317138)

[2.2.7 RFM\_GET\_BATTERY\_CAPACITY (获取电池电量) 13](#_Toc491754965)

[2.2.8 RFM\_SET\_GET\_BLUETOOTH\_NAME（设置/获取蓝牙设备名称） 14](#_Toc298918352)

[2.2.9 RFM\_SET\_BLU\_HID\_SEND（设置蓝牙键盘输出） 15](#_Toc992099532)

[2.3.0 RFM\_SET\_GET\_OUTPUTMODE（设置/获取选择蓝牙输出模式） 17](#_Toc1967290994)

[2.3.1 RFM \_REPORT\_KEYSTA（上报按键开始结束状态） 18](#_Toc1567807396)

[2.3.2 RFM\_SET\_GET\_READMODE（设置/获取读码模式） 19](#_Toc1994766691)

[2.4 国际标准(ISO 18000-6C)协议相关命令 20](#_Toc131007982)

[2.4.1 RFM\_INVENTORYISO\_CONTINUE (标签盘点) 21](#_Toc1569795530)

[2.4.2 RFM\_INVENTORY\_STOP (停止盘点) 22](#_Toc1684329084)

[2.4.3 RFM\_READISO\_TAG (读取标签数据) 23](#_Toc1440863005)

[2.4.4 RFM\_WRITEISO\_TAG (写入标签数据) 2](#_Toc121419102)4

[2.4.5 RFM\_LOCKISO\_TAG (锁定标签数据) 2](#_Toc121419103)6

[2.4.6 RFM\_KLENLISO\_TAG (灭活标签) 2](#_Toc121419104)7

[2.4.7 RFM\_SETISO\_SELECTMASK (选择需要操作的标签) 2](#_Toc292312038)8

[附录A. 标签返回的操作状态 2](#_Toc1108675776)9

[附录B.CRC16校验参考C代码 2](#_Toc2008972112)9

[附录C. STATUS的定义。 3](#_Toc1361361747)0

# 1对接通讯协议

本模块可以通过BLE/USB/USB虚拟串口通讯。

## **1.1** 串口通讯参数

模块通过串口与外部应用进行通讯

表格A-1 模块对接串口参数

|  |  |  |
| --- | --- | --- |
| 项目 | 参数 | 备注 |
| 波特率 | 115200bps |  |
| 数据位 | 8 |  |
| 停止位 | 1 |  |
| 奇偶校验位 | None |  |

## **1.2** 数据帧格式

上层应用下发的数据包以下称“命令”，而模组返回给上层应用的数据包以下称“响应”。以下所有数据段的长度单位均为字节。

数据传输顺序：对每个由多个字节构成的数据项，先发送最高有效字节，字节内先发送最低有效位。

命令的数据帧格式见表格A-2：

表格A-2 命令的数据帧格式

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **MSB** |  |  |  |  | **LSB** |
|  | Control Field | | | Information Field | EpLENogue Field |
| HEAD | ADDR | CMD | LEN | Data[] | CHECK |
| 1Byte | 1Byte | 2Bytes | 1Byte | 0~255Byte | 2Byte |

响应的数据帧格式见表格A-3：

表格A-3 响应的数据帧格式

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **MSB** |  |  | |  | | **LSB** |
|  | Control Field | | | Information Field | | EpLENogue Field |
| HEAD | ADDR | CMD | LEN | STATUS | Data[] | CHECK |
| 1Byte | 1Byte | 2Bytes | 1Byte | 1Byte | 0~254Byte | 2Byte |

数据帧结构包括以下三大部分：帧控制域(Control Field)、信息数据域(Information Field) 和帧尾域(EpLENogue Field)。

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | 长度  (字节) | 说明 |
| Control Field | HEAD | 1 | HEAD固定为十六进制数0xCF，该字节用于数据帧同步。 |
| ADDR | 1 | 读写器地址。地址范围：0x00~0xFE，0xFF为广播地址，读写器只响应和自身地址相同及地址为0xFF的命令。读写器出厂时地址为0x00。 |
| CMD | 2 | 命令代码，命令字定义见表格A-7。 |
| LEN | 1 | 该长度为实际Information Field的所有字节数据长度，最大数据长度为255Bytes（0xFF）。 |
| Information Field | STATUS | 1 | STATUS 是模块的响应中包含的对上位机命令的执行状态。STATUS只存在模块返回的响应数据帧中，上层应用下发的命令帧中没有STATUS部分。STATUS为0表示命令执行成功，此处执行成功只表示模块成功接收到标签响应，如果标签响应中有标签执行状态，则还应进一步判断标签执行状态是否正确。STATUS的定义见表格A-6。 |
| Data[] | 不定 | 参数域。在实际命令中，可以不存在。Data[]是需要传递的实际数据，在各命令格式中将具体定义Data[]有效字节。 |
| EpLENogue Field | MSB-CRC16 | 1 | CRC16高字节。CRC16是从HEAD到Data[]的CRC16值 |
| LSB-CRC16 | 1 | CRC16低字节。 |

CRC16参考代码见 附录B.

# 2 命令帧集

超高频RFID读写器模块，在以下的描述中简称为“模块”。

## **2.1** 命令列表

表格A-7控制命令示例列表

|  |  |  |  |
| --- | --- | --- | --- |
| **命令名称** | **CMD控制字节** | **功能描述** | **备注** |
| **EPC指令** | | | |
| RFM\_INVENTORYISO\_CONTINUE | 0x0001 | 标签盘点(流程) |  |
| RFM\_INVENTORYISO\_STOP | 0x0002 | 停止标签盘点 |  |
| RFM\_READISO\_TAG | 0x0003 | 读取标签数据 |  |
| RFM\_WRITEISO\_TAG | 0x0004 | 写入标签数据(流程) |  |
| RFM\_LOCKISO\_TAG | 0x0005 | 锁定标签数据(流程) |  |
| RFM\_KLENLISO\_TAG | 0x0006 | 灭活标签(流程) |  |
| RFM\_SETISO\_SELECTMASK | 0x0007 | 设置标签EPC号码 |  |
| **模块自定义指令** | | | |
| RFM\_MODULE\_INT | 0X0050 | 初始化设备 | 停止所有动作 |
| RFM\_GET\_DEVICEINFO | 0x0051 | 蓝牙获取所有模块版本号和sn码 |  |
| RFM\_REBOOT | 0x0052 | 恢复出厂设置 | 恢复默认值 |
| RFM\_GET\_BATTERY\_CAPACITY | 0x0083 | 获取电池电量 |  |
| RFM\_SET\_ALL\_PARAM | 0x0071 | 设置所有基础参数 | 将所有基础参数打包，一次性全设置 |
| RFM\_GET\_ALL\_PARAM | 0x0072 | 读取所有基础参数 | 将所有基础参数打包，一次性全部读取 |
| RFM\_SET\_GET\_S-PERMISSION\_PARAM | 0X0076 | 特定读取权限设置 | H103 |
| RFM\_SET\_GET\_BLUETOOTH\_NAME | 0x0086 | 设置蓝牙设备名称 |  |
| RFM\_SET\_BLU\_HID\_SEND | 0x0087 | 蓝牙键盘输出 |  |
| RFM\_SET\_GET\_OUTPUTMODE | 0x0088 | 设置/获取选择输出模式 |  |
| RFM \_REPORT\_KEYSTA | 0x0089 | 上报按键开始结束状态 |  |
| RFM\_SET\_GET\_READMODE | 0x008E | 设置/获取读码模式 | H103,H104 |

## 2.2 模块通用控制命令

### 2.2.1 RFM\_MODULE\_INT（初始化设备）

设备按照存储的参数进行初始化配置。

* 命令格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | CHECK |
| 0xCF | 0xFF | 0x0050 | 0x00 | 2Byte |

* 响应格式及状态字节

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | STATUS | CHECK |
| 0xCF | 0x00 | 0x0050 | 0x01 | 1Byte | 2Byte |

 STATUS：该命令只会返回执行成功（值为0x00），其他值无效；

### 2.2.2 RFM\_REBOOT（设备恢复出厂设置）

恢复出厂设置命令，当该命令被成功执行时，模块恢复出厂设置（是否需要重启？）。

* 命令格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | CHECK |
| 0xCF | 0xFF | 0x0052 | 0x00 | 2Byte |

* 响应格式及状态字节

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | STATUS | CHECK |
| 0xCF | 0x00 | 0x0052 | 0x01 | 1Byte | 2Byte |

 STATUS：该命令只会返回执行成功（值为0x00），其他值无效；

### 2.2.3 RFM\_GET\_DEVICEINFO（获取模块信息）

该命令用于获取当前设备的版本信息，包括CP模块和RFID模块的硬件版本号、固件版本号和SN号。

* 命令格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | CHECK |
| 0xCF | 0xFF | 0x0070 | 0x00 | 2Byte |

* 响应格式及状态字节

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | STATUS | PAYLOAD | | |
| CPHardVer | CPFirmVer | CPSN\_code |
| 0xCF | 0x00 | 0x0070 | 0xAC | 1Byte | 32Bytes | 32Bytes | 12Bytes |

|  |  |  |  |
| --- | --- | --- | --- |
| PAYLOAD | | | CHECK |
| RFIDModeVer | RFIDModeName | RFIDMode\_SNCode |
| 32Bytes | 32Bytes | 12Bytes | 2Byte |

各字节定义：

|  |  |  |
| --- | --- | --- |
| 字段 | 字节 | 描述 |
| STATUS | 1 | 该命令只会返回执行成功（值为0x00），其他值无效； |
| CPHardVer | 32 | CP模块硬件版本号，格式为ASCII 码，以’\0’为结尾，实际的硬件版本号可能不足32字节，没有用到的其他字节以’\0’补充； |
| CPFirmVer | 32 | CP模块固件版本号和bootloader版本号，格式与硬件版本号相同； |
| CP\_SNCode | 12 | CP模块12字节序列号，格式为ASCII码，没有用到的其他字节以’\0’补充； |
| RFIDModeVer | 32 | RFID模块版本号，格式为ASCII 码，以’\0’为结尾。实际的版本号为2字节，不足32字节，没有用到的其他字节以’\0’补充。第一字节为主版本，第二字节为子版本；示例：Byte0=0x0、Byte1=0x1e，版本号为3.30。 |
| RFIDModeName | 32 | RFID模块型号，格式为ASCII 码，以’\0’为结尾。实际的型号不足32字节，没有用到的其他字节以’\0’补充。 |
| RFIDMode\_SNCode | 12 | RFID模块序列号，格式为ASCII码，实际的序列号为4字节，不足32字节，没有用到的其他字节以’\0’补充； |

### 2.2.4 RFM\_SET\_ALL\_PARAM (设置所有可配置参数)

该命令用于设置设备的所有可配置参数。用户使用模块对标签进行操作前需要用该命令设置设备的所有可配置参数。若用户没有进行该设置，设备工作时将使用默认设置。

* 命令格式

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | PAYLOAD | | | | | | | |
| Addr | RFIDPRO | Work  Mode | Interface | Baudrate | WGSet | Ant | RfidFreq |
| 0xCF | 0xFF | 0x0071 | 0x19 | 1Byte | 1Byte | 1Byte | 1Byte | 1Byte | 1Byte | 1Byte | 8Bytes |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PAYLOAD | | | | | | | | | | CHECK |
| RfidPower | InquiryArea | QValue | Session | AcsAddr | AcsDataLen | FilterTime | TriggerTime | BuzzerTime | Polling  Interval |
| 1Byte | 1Byte | 1Byte | 1Byte | 1Byte | 1Byte | 1Byte | 1Byte | 1Byte | 1Byte | 2Bytes |

* + Addr：设备的通信地址，默认为0x00。这个地址不能为0xFF。如果设置为0xFF，则读写模块将返回参数出错信息。
  + RFIDPRO：设备射频RFID的协议标准规范，0x00：ISO 18000-6C； 0x01：GB/T 29768；0x02：GJB 7377.1；当前仅支持 ISO 18000-6C。
  + WorkMode：设备的工作模式，默认值0。

|  |  |
| --- | --- |
| WorkMode | 工作模式 |
| 0 | 应答模式 |
| 1 | 主动模式 |
| 2 | 触发模式 |

* + Interface：设备的通信接口，默认值0x80，具体释义如下：

|  |  |
| --- | --- |
| Interface | 接口类型 |
| 0x80 | RS232 |
| 0x40 | RS485 |
| 0x20 | RJ45 |
| 0x10 | WiFi |
| 0x01 | USB |
| 0x02 | keyboard |
| 0x04 | CDC\_COM |

* + Baudrate：串口波特率，默认值为4，具体释义如下：

|  |  |
| --- | --- |
| baudrate | 实际波特率 |
| 0 | 9600bps |
| 1 | 19200 bps |
| 2 | 38400 bps |
| 3 | 57600 bps |
| 4 | 115200 bps |

* + WGSet：韦根数据输出接口的配置参数，默认值0x00，具体释义如下：

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| WGSet | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
| Bit位释义 | 0：关闭韦根输出  1：开启韦根输出 | 0：wg26  1：wg34 | 0：低位在前  1：高位在前 | 备用 | 备用 | 备用 | 备用 | 备用 |

* + Ant：设备所有的天线号，按位表示选择使用的天线，对应Bit位值为1则表示使用该天线，值为0则表示不使用该天线；从低位开始，第0位表示1号天线，第1位表示2号天线，以此类推，最多能表示8个天线；不同模块支持不同个天线，视具体情况而定；默认值0x01，表示1号天线。
  + RfidFreq：设备的RFID频率相关参数，用于选择频段及各频段中的上限频点和下限频点，长度8Bytes，具体参数如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| RfidFreq | | | | |
| REGION | STRATFREI | STRATFRED | STEPFRE | CN |
| 1 Byte | 2Byte | 2Byte | 2Byte | 1Byte |

各字节定义：

设置频率先按照REGION确定频段范围，再参考STRATFREI、STRATFRED、STEPFRE、CN 计算具体的频率范围。

|  |  |  |
| --- | --- | --- |
| 字段 | 字节 | 描述 |
| REGION | 1 | 各国频段范围：  0x00: 用户根据需求自定义；  0x01：US [902.75~927.25]  0x02：Korea [917.1~923.5]  0x03：EU [865.1~868.1]  0x04：JAPAN [952.2~953.6]  0x05：MALAYSIA [919.5~922.5]  0x06：EU3 [865.7~867,5]  0x07：CHINA\_BAND1 [840.125~844.875]  0x08：CHINA\_BAND2 [920.125~924.875] |
| STRATFREI | 2 | 兆赫兹起始频率的整数部分；如920.125MHz，STRATFREI = 920 = 0x0398，高字节=0x03，低字节=0x98； |
| STRATFRED | 2 | 兆赫兹起始频率的小数部分；如 920.125MHz，STRATFRED =125，高字节=0x00，低字节=0x7D |
| STEPFRE | 2 | 频率步进（KHz），需参考各频段计算公式；如125KHz，STEPFRE =125，高字节=0x00，低字节=0x7D； |
| CN | 1 | 信道数； |

频率计算公式： 最小频点：Fmin = STRATFREI + STRATFRED/1000 （单位：MHz）

最大频点：Fmax = Fmin + STEPFRE\*CN/1000 （单位：MHz）

各频段计算公式：(设置的频率范围须在各国标准内)

Chinese band2： Fs = 920.125 + CN \* 0.25 (MHz) 其中CN∈[0, 19]。

US band： Fs = 902.75 + CN \* 0.5 (MHz) 其中CN∈[0,49]。

Korean band： Fs = 917.1 + CN \* 0.2 (MHz) 其中CN∈[0, 31]。

EU band: Fs = 865.1 + CN\*0.2(MHz) 其中CN∈[0, 14]。

Ukraine band: Fs = 868.0 + CN\*0.1(MHz) 其中CN∈[0, 6]。

Peru band： Fs = 916.2 + CN\*0.9(MHz) 其中CN∈[0, 11]。

Chinese band1： Fs = 840.125 + CN \* 0.25 (MHz) 其中CN∈[0, 19]。

EU3 band： Fs = 865.7 + CN \* 0.6(MHz) 其中CN∈[0, 3]。

US band3： Fs = 902 + CN \* 0.5 (MHz) 其中CN∈[0,52]。

Taiwan band： Fs = 922.25 + CN \* 0.5 (MHz) 其中CN∈[0,11]。

* + RfidPower：设备的RFID输出功率，单位为：dBm，取值范围为：[0, 30]dBm，其他无效。
  + InquiryArea：设备要访问标签的存储区区域。0x00：保留区；0x01(默认)：EPC存储区；0x02：TID存储区；0x03：USER存储区；0x04：EPC+TID；0x05：EPC+USER；0x06：EPC+TID+USER；其它值保留。若命令中出现了其它值，将返回参数出错的消息。
  + QValue：询查EPC标签时使用的初始Q值，Q值的设置应为场内的标签数量约等于2Q。Q值的默认值为4，Q值的范围为0～15，若命令中出现了其它值，将返回参数出错的消息。
  + Session：询查EPC标签时使用的Session值，默认为0，取值范围[0，3]，其它值，将返回参数出错的消息。

|  |  |
| --- | --- |
| Session | 注释 |
| 0 | Session使用S0 |
| 1 | Session使用S1 |
| 2 | Session使用S2 |
| 3 | Session使用S3 |
| 0xFD | Auto-3 |
| 0xFE | Auto-2 |
| 0xFF | Auto-1 |

* + AcsAddr：设备要访问标签存储区的起始地址，单位 ：Byte，默认值：0x00：访问EPC区时，0x00表示访问EPC区除CRC和PC段的EPC号码段起始地址；访问其他存储区时，0x00表示该存储区的起始地址。
  + AcsDataLen：设备要访问标签存储区的数据长度，单位 ：Byte，默认值：0x00。
  + FilterTime：过滤时间，在读取成功一张标签数据后的该值时间内，过滤掉有相同数据的标签。单位为：S，取值范围为：[0, 255]，其他无效；默认值为0，没有过滤。
  + TriggerTime：设备收到触发信号后的询查时长，单位为：S，默认值为1，取值范围为：[0, 255]，其他无效。
  + BuzzerTime：设备执行成功后蜂鸣器鸣叫时长，单位为：10ms，取值范围为：[0, 255]，其他无效；默认为1，当为0时，表示蜂鸣器不鸣叫。
  + PollingInterval：询查间隔时间，单位为：10ms，取值范围为：[0, 255]，其他无效，默认为1。
* 响应格式及状态字节

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | STATUS | CHECK |
| 0xCF | 0xXX | 0x0071 | 0x01 | 1Byte | 2Byte |

* + STATUS：
  1. 0x00：命令执行成功；
  2. 0x01：参数错误；
  3. 其他值：见目录C；

### 2.2.5 RFM\_GET\_ALL\_PARAM (读取所有可配置参数)

该命令用于读取模块设备的所有可配置参数。

* 命令格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | CHECK |
| 0xCF | 0xFF | 0x0072 | 0x00 | 2Byte |

* 响应格式及状态字节

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | PAYLOAD | | | | | | | | |
| STATUS | Addr | RFIDPRO | Work  Mode | Interface | Baudrate | WGSet | Ant | RfidFreq |
| 0xCF | 0xFF | 0x0072 | 0x1A | 1Byte | 1Byte | 1Byte | 1Byte | 1Byte | 1Byte | 1Byte | 1Byte | 8Bytes |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PAYLOAD | | | | | | | | | | CHECK |
| RfidPower | InquiryArea | QValue | Session | AcsAddr | AcsDataLen | FilterTime | TriggerTime | BuzzerTime | Polling  Interval |
| 1Byte | 1Byte | 1Byte | 1Byte | 1Byte | 1Byte | 1Byte | 1Byte | 1Byte | 1Byte | 2Bytes |

* + STATUS：
  1. 0x00：命令执行成功；
  2. 其他值：见目录C；
  + 其余参数参照RFM\_SET\_ALL\_PARAM。

### 2.2.6 RFM\_SET\_GET\_S-PERMISSION\_PARAM (设置读取权限)

该命令用于设置密码与掩码询查权限参数、继电器使用参数、缓存开关参数和公共协议使用参数。

* 命令格式

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| HEAD | Control Field | | | PAYLOAD | | | | | | | | CHECK |
| ADDR | CMD | LEN | OPTION | Code  EN | Codes | Mask  EN | Start  Add | Mask  Len | Mask  Data | Mask  Condition |
| 0xCF | 0xFF | 0x0076 | 0x29 | 1Byte | 1Byte | 4Bytes | 1Byte | 1Byte | 1Byte | 31Bytes | 1Byte | 2Bytes |

参数释义：

Option：命令控制选项，0x01设置，0x02读取，其它值无效。

CodeEN：密码功能使能参数，长度1字节。0x01启用，0x00不启用。默认0x00。

Codes ：标签的访问密码，长度4字节。默认[0x00,0x00,0x00,0x00]。

MaskEN：掩码功能使能参数，长度1字节。0x01启用，0x00不启用。默认0x00。

StartAdd：掩码起始地址，长度1字节，单位字节。默认0x00。

Mask Len：掩码长度，长度1字节，单位字节，最大31。默认0x00。

MaskData：掩码数据，长度31字节，掩码长度不足31字节时，后面字节数据以0补充。

默认全为0。

MaskCondition：掩码条件，长度1字节。0x00：密码或者掩码符合；0x01：密码和掩码同时符合。默认0x00。

* 响应格式及状态字节

设置响应：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | STATUS | Option | CHECK |
| 0xCF | 0x00 | 0x0073 | 1Byte | 1Byte | 1Byte | 2Byte |

获取响应：

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| HEAD | Control Field | | | PAYLOAD | | | | | | | | | CHECK |
| ADDR | CMD | LEN | STATUS | OPTION | Code  EN | Codes | Mask  EN | Start  Add | Mask  Len | Mask  Data | Mask  Condition |
| 0xCF | 0xFF | 0x0076 | 0x29 | 1Byte | 1Byte | 0xFF | 4  0xFF | 1Byte | 1Byte | 1Byte | 31Bytes | 1Byte | 2Bytes |

* + STATUS：

1）0x00：命令执行成功；

1. 其他值：无效；

### 2.2.7 RFM\_GET\_BATTERY\_CAPACITY (获取电池电量)

该命令获取电池电量

* 命令格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | CHECK |
| 0xCF | 0xFF | 0x0083 | 0x00 | 2Byte |

* 响应格式及状态字节

获取响应：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | STATUS | Battery | CHECK |
| 0xCF | 0x00 | 0x0083 | 0x02 | 1Byte | 1Byte | 2Byte |

|  |  |  |
| --- | --- | --- |
| 字段 | 字节 | 描述 |
| STATUS | 1 | 0x00：命令执行成功；0x01：参数错误。 |
| Battery | 1 | 若电池电量为百分之20数据为[0x14] |

### 2.2.8 RFM\_SET\_GET\_BLUETOOTH\_NAME（设置/获取蓝牙设备名称）

该命令设置/获取蓝牙设备名称

* 命令格式

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | PAYLOAD | | CHECK |
| Option | BleName |
| 0xCF | 0xFF | 0x0086 | 1Byte | 1Byte | NC or 1-20Byte | 2Byte |

* + Option：命令控制选项

0x01：设置，其后接1Byte长度的RFID ；

0x02：读取，其后不接RFID；

0x03: 0x03

其他值：见目录C；

|  |  |  |
| --- | --- | --- |
| 字段 | 字节 | 描述 |
| Option | 1 | 0x01设置，0x02获取，0x03收到应答指令名称设置成功之后，下发Option为0x03指令即可 |
| BleName | 1-20 | 若蓝牙名称为CF-H100最终数据为{0x43,0x46,0x2D,0x48,0x31,0x30,0x30} |

* 响应格式及状态字节

设置响应：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | STATUS | Option | CHECK |
| 0xCF | 0x00 | 0x0086 | 1Byte | 1Byte | 1Byte | 2Byte |

获取响应：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | PAYLOAD | | | CHECK |
| STATUS | Opion | BleName |
| 0xCF | 0x00 | 0x0086 | 1Byte | 1Byte | 1Byte | 1-20Byte | 2Byte |

|  |  |  |
| --- | --- | --- |
| 字段 | 字节 | 描述 |
| Option | 1 | 0x01设置，0x02获取 |
| STATUS | 1 | 0x00：命令执行成功；0x01：参数错误。 |
| BleName | 1-20 | 例如数据为{0x43,0x46,0x2D,0x48,0x31,0x30,0x30}那么名称就为CF-H100 |

### 2.2.9 RFM\_SET\_BLE\_HID\_SEND（发送蓝牙键盘数据）

该命令为在配置为HID模式之后，发送键盘数据的格式；需要在之前配置RFM\_SET\_GET\_OUTPUTMODE指令；

* 命令格式

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | KEY BOARD | | | | | | | | CHECK |
| KEY0 | KEY1 | KEY2 | KEY3 | KEY4 | KEY5 | KEY6 | KEY7 |
| 0xCF | 0xFF | 0x0087 | 0x08 | 1Byte | 1Byte | 1Byte | 1Byte | 1Byte | 1Byte | 1Byte | 1Byte | 2Byte |

/\*键盘发送给PC的数据每次8个字节

KEY0 KEY1 KEY2 KEY3 KEY4 KEY5 KEY6 KEY7

定义分别是：

|  |  |  |
| --- | --- | --- |
| 字段 | 字节 | 描述 |
| KEY0 | 1 | |--bit0: Left Control是否按下，按下为1  |--bit1: Left Shift 是否按下，按下为1  |--bit2: Left Alt 是否按下，按下为1  |--bit3: Left GUI 是否按下，按下为1  |--bit4: Right Control是否按下，按下为1  |--bit5: Right Shift 是否按下，按下为1  |--bit6: Right Alt 是否按下，按下为1  |--bit7: Right GUI 是否按下，按下为1 |
| KEY1 | 1 | 0x00 |
| KEY2 | 1 | 普通按键 |
| KEY3 | 1 | 普通按键 |
| KEY4 | 1 | 普通按键 |
| KEY5 | 1 | 普通按键 |
| KEY6 | 1 | 普通按键 |
| KEY7 | 1 | 普通按键 |

### 2.3.0 RFM\_SET\_GET\_OUTPUTMODE（设置/获取选择蓝牙输出模式）

该命令设置选择蓝牙输出模式，可以配置设备为蓝牙通讯或者键盘仿真输出的模式；

* 命令格式

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | Option | MODE | CHECK |
| 0xCF | 0xFF | 0x0088 | 0x02 | 1Byte | 1Byte | 2Byte |

* 响应格式及状态字节

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | STATUS | CHECK |
| 0xCF | 0x00 | 0x0088 | 0x01 | 1Byte | 2Byte |

该命令获取蓝牙输出模式

* 命令格式

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | Option | CHECK |
| 0xCF | 0xFF | 0x0088 | 0x01 | 1Byte | 2Byte |

* 响应格式及状态字节

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | STATUS | MODE | CHECK |
| 0xCF | 0x00 | 0x0088 | 0x02 | 1Byte | 1Byte | 2Byte |

|  |  |  |
| --- | --- | --- |
| 字段 | 字节 | 描述 |
| Option | 1 | 0x01：设置命令 0x02获取命令 |
| STATUS | 1 | 0x00：命令执行成功 0x01参数错误； |
| MODE | 1 | 0x01：蓝牙透传输出；0x00：蓝牙HID输出。 |

### 2.3.1 RFM \_REPORT\_KEYSTA（上报按键开始结束状态）

该命令上报按键开始/结束状态，为提示上位机准备接收数据；

* 命令格式

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | START | CHECK |
| 0xCF | 0x00 | 0x0089 | 0x01 | 0x01 | 2Byte |

* 命令格式

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | END | CHECK |
| 0xCF | 0x00 | 0x0089 | 0x01 | 0x02 | 2Byte |

### 2.3.2 RFM\_SET\_GET\_READMODE（设置/获取读码模式）

该命令设置/获取 读码模式；用于不同模块切换，条码和RFID模块；

* 命令格式

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | Option | RAEDMODE | CHECK |
| 0xCF | 0xFF | 0x008E | 0x02 | 1Byte | 1Byte | 2Byte |

* 响应命令格式及状态字节

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | STATUS | Option | CHECK |
| 0xCF | 0x00 | 0x008E | 0x02 | 1Byte | 1Byte | 2Byte |

* 获取命令格式

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | Option | CHECK |
| 0xCF | 0xFF | 0x008E | 0x01 | 1Byte | 2Byte |

* 响应获取格式及状态字节

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | STATUS | Option | RAEDMODE | CHECK |
| 0xCF | 0x00 | 0x008E | 0x03 | 1Byte | 1Byte | 1Byte | 2Byte |

|  |  |  |
| --- | --- | --- |
| 字段 | 字节 | 描述 |
| Option | 1 | 0x01：设置命令 0x02获取命令 |
| STATUS | 1 | 0x00：命令执行成功 0x01参数错误； |
| RAEDMODE | 1 | 0x00: RF读卡 ，0x01扫描头读卡 |

## 2.4 国际标准(ISO 18000-6C)协议相关命令

### 2.4.1 RFM\_INVENTORYISO\_CONTINUE (标签盘点)

该命令为启动国际标准多标签盘点命令，盘点功能具有防碰撞算法。

该命令可以指定盘点时长，如果输入盘点次数为0，则表示持续盘点标签，直到接收到停止盘点命令。 执行盘点过程中，每成功盘点到一个标签，都会通过一个“STATUS”值为“0”的命令响应返回新盘点到的标签信息。

当盘点命令成功执行完成后，最后会返回的响应“STATUS”值为“0x12”的命令响应，用于通知盘点命令已执行完成。

* 命令格式

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | PAYLOAD | | CHECK |
| InvType | InvParam |
| 0xCF | 0xFF | 0x0001 | 0x05 | 1Byte | 4Byte | 2Byte |

* + InvType：盘点方式：

0x00：按时间盘点标签，在执行指定时间后停止盘点或接收到停止盘点命令后停止盘点；

0x01: 按照循环次数盘点，在执行指定次数的轮询后或者接到停止盘点指令后停止盘点；

* + InvParam ：盘点方式参数：

1.若InvType为0x00：

InvParam表示盘点时间，单位为：秒，如果该值为0，则表示持续盘点标签，直到接收到停止盘点命令；

2.若InvType为0x01：

InvParam表示盘点次数，单位为：次，该值必须大于0；

* 响应格式及状态字节

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | STATUS | PAYLOAD | | | | | CHECK |
| RSSI | Antenna | Channel | EPC LEN | EPC NUM |
| 0xCF | 0x00 | 0x0001 | 1Byte | 1Byte | 2Bytes | 1 Byte | 1 Byte | 1Byte | N Bytes | 2Byte |

各字节定义：

|  |  |  |
| --- | --- | --- |
| 字段 | 字节 | 描述 |
| STATUS | 1 | 0x00：成功盘点到标签，PAYLOAD中包含该标签信息；  0x01：Q、MemBank参数值错误或Length与Mask数据长度不一致；  0x02：由于模块内部错误导致的命令执行失败；  0x12：没有盘点到标签或整个盘点命令执行完成；  0x17：标签数据超出串口最大传输长度；  其他值：无效； |
| RSSI | 2 | 标签ACK响应的RSSI，单位为dBm，带符号数，负数使用补码格式； |
| Channel | 1 | 从哪个信道接收到的标签数据，值从0开始，0表示0信道，1表示1信道，以此类推； |
| EPC LEN | 1 | 标签的EPC号码长度（字节）； |
| EPC NUM | N Byte | 标签的EPC号码； |

### 2.4.2 RFM\_INVENTORY\_STOP (停止盘点)

该命令用户主动停止国际标准多标签防碰撞盘点流程。

* 命令格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | CHECK |
| 0xCF | 0xFF | 0x0002 | 0x00 | 2Byte |

* 响应格式及状态字节

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | STATUS | CHECK |
| 0xCF | 0x00 | 0x0002 | 0x01 | 1Byte | 2Byte |

 STATUS：该命令只会返回执行成功(值为0x00)，其他值无效；

### 2.4.3 RFM\_READISO\_TAG (读取标签数据)

该命令帧为模块读取国际标准协议标签存储区数据。

调用该方法之前，请先通过RFM\_SETISO\_SELECTMASK命令设置盘点条件(用于选择标签)。 进行国际标准协议标签数据操作(读取标签数据、写入标签数据、锁定标签、灭活标签)时标签会返回标签操作状态，这时模块的响应中含有一个字节的标签操作状态。标签操作状态的定义见附录A

* 命令格式

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN |  | PAYLOAD | | |  | CHECK |
| Option | AccPwd | MemBank | WordPtr | WordCount |
| 0xCF | 0xFF | 0x0003 | 1Byte | 1Byte | 4 Bytes | 1Byte | 2 Bytes | 1 Bytes | 2Byte |

各字段定义：

|  |  |  |
| --- | --- | --- |
| 字段 | 字节 | 描述 |
| Option | 1 | 命令选项（该字段未使用，无效，默认0x00） ； |
| AccPwd | 4 | 访问口令，用于标签进入安全态，默认0x00000000； |
| MemBank | 1 | 所要读取标签的存储区，值列表如下：  0x00：Reserved； 0x01：EPC； 0x02：TID； 0x03：User； |
| WordPtr | 2 | 指向逻辑存储区的读取起始地址(字)； |
| WordCount | 1 | 需要读取的字个数，不能为0，默认为4，取值范围[1,120] |

* 响应格式及状态字节

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | STATUS | PAYLOAD | | | | | | | | CHECK |
| TagStatus | Antenna | CRC | PC | EPC LEN | EPC NUM | WordCount | Data |
| 0xCF | 0x00 | 0x0003 | 1Byte | 1Byte | 1Byte | 1Byte | 2Byte | 2Bytes | 1Byte | N  Bytes | 1Bytes | N-Byte | 2Byte |

各字段定义：

根据STATUS和TagStatus判断命令是否执行成功。

|  |  |  |
| --- | --- | --- |
| 字段 | 字节 | 描述 |
| STATUS | 1 | 参考表格A-6 STATUS定义 |
| TagStatus | 1 | 标签返回的操作状态，具体定义见附录A； |
| Antenna | 1 | 从哪根天线中读出 |
| CRC | 2 | 标签响应数据中的CRC数据 |
| PC | 2 | 标签响应数据中的PC数据 |
| EPC LEN | 1 | 标签的EPC号码长度（字节）； |
| EPC NUM | N Byte | 标签的EPC号码； |
| WordCount | 1 | 成功读取到的标签数据字个数； |
| Data | N Byte | 成功读取到的标签数据，长度为WordCount×2个字节； |

### 2.4.4 RFM\_WRITEISO\_TAG (写入标签数据)

该命令为模块向标签存储区写入数据。

调用该方法之前，请先通过RFM\_SETISO\_SELECTMASK命令设置盘点条件(用于选择标签)。

* 命令格式

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | PAYLOAD | | | | | | CHECK |
| Option | AccPwd | MemBank | WordPtr | WordCount | Data |
| 0xCF | 0xFF | 0x0004 | 1Byte | 1Byte | 4Byte | 1Byte | 2Byte | 1Byte | N-Byte | 2Byte |

各字段定义：

|  |  |  |
| --- | --- | --- |
| 字段 | 字节 | 描述 |
| Option | 1 | 命令选项（该字段未使用，无效，默认0x00） ； |
| AccPwd | 4 | 访问口令，用于标签进入安全态，默认0x00000000； |
| MemBank | 1 | 所要读取标签的存储区，值列表如下：  0x00：Reserved； 0x01：EPC； 0x02：TID； 0x03：User； |
| WordPtr | 2 | 指向逻辑存储区的读取起始地址（字）； |
| WordCount | 1 | 需要写入标签的数据字个数(1个字为两个字节)，必须大于0； |
| Data | N-Byte | 需要写入标签的数据，长度必须为字的整数倍，长度为1 ~ WordCount个字 |

* 响应格式及状态字节

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | STATUS |  | | PAYLOAD | | | | CHECK |
| TagStatus | Antenna | CRC | PC | EPC LEN | EPC NUM |
| 0xCF | 0x00 | 0x0004 | 1Byte | 1Byte | 1Byte | 1Byte | 2Byte | 2Bytes | 1Byte | N  Bytes | 2Byte |

* + TagStatus：标签返回的操作状态，具体定义见附录A，只有在STATUS字段值为0时，响应数据中才会包含该字段；

各字段定义：

根据STATUS和TagStatus判断命令是否执行成功。

|  |  |  |
| --- | --- | --- |
| 字段 | 字节 | 描述 |
| STATUS | 1 | 参考表格A-6 STATUS定义 |
| TagStatus | 1 | 标签返回的操作状态，具体定义见附录A； |
| Antenna | 1 | 从哪个天线端口接收到的标签数据，值范围为：1~4，分别表示1~4号天线 |
| CRC | 2 | 标签响应数据中的CRC数据 |
| PC | 2 | 标签响应数据中的PC数据 |
| EPC LEN | 1 | 标签的EPC号码长度（字节）； |
| EPC NUM | N Byte | 标签的EPC号码； |

### 2.4.5 RFM\_LOCKISO\_TAG (锁定标签数据)

该命令对指定国际标准协议标签的指定存数据区进行锁定操作。

调用该方法之前，请先通过RFM\_SETISO\_SELECTMASK命令设置盘点条件(用于选择标签)。

* 命令格式

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN |  | PAYLOAD | | CHECK |
| AccPwd | Area | Action |
| 0xCF | 0xFF | 0x0005 | 0x06 | 4Byte | 1Byte | 1Byte | 1Byte |

各字段定义：

|  |  |  |
| --- | --- | --- |
| 字段 | 字节 | 描述 |
| AccPwd | 4 | 访问口令，用于标签进入安全态，默认0x00000000 |
| Area | 1 | 需要锁定的区域，值列表如下：  0x00：灭活密码区；0x01：访问密码区； 0x02：EPC； 0x03：TID； 0x04：User； |
| Action | 1 | 锁定操作类型，值列表如下：  0x00：开放； 0x01：永久开放； 0x02：锁定； 0x03：永久锁定； |

* 响应格式及状态字节

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | STATUS |  | PAYLOAD | | | | | CHECK |
| TagStatus | Antenna | CRC | PC | EPC LEN | EPC NUM |
| 0xCF | 0x00 | 0x0005 | 1Byte | 1Byte | 1Byte | 1Byte | 2Bytes | 2Bytes | 1Byte | N  Bytes | 1Byte |

各字段定义：

根据STATUS和TagStatus判断命令是否执行成功。

|  |  |  |
| --- | --- | --- |
| 字段 | 字节 | 描述 |
| STATUS | 1 | 参考表格A-6 STATUS定义 |
| TagStatus | 1 | 标签返回的操作状态，具体定义见附录A； |
| Antenna | 1 | 从哪个天线端口接收到的标签数据，值范围为：1~4，分别表示1~4号天线 |
| CRC | 2 | 标签响应数据中的CRC数据 |
| PC | 2 | 标签响应数据中的PC数据 |
| EPC LEN | 1 | 标签的EPC号码长度（字节）； |
| EPC NUM | N Byte | 标签的EPC号码； |

### 2.4.6 RFM\_KLENLISO\_TAG (灭活标签)

该命令为模块灭活标签。

调用该方法之前，请先通过RFM\_SETISO\_SELECTMASK命令设置盘点条件(用于选择标签)。

灭活，只支持非0灭活密码.

* 命令格式

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | PAYLOAD | CHECK |
| KLENlPwd |
| 0xCF | 0xFF | 0x0006 | 0x04 | 4Byte | 1Byte |

* + KLENlPwd：KLENl指令密码；
* 响应格式及状态字节

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | STATUS |  | | PAYLOAD | | | | CHECK |
| TagStatus | Antenna | CRC | PC | EPC LEN | EPC NUM |
| 0xCF | 0x00 | 0x0006 | 1Byte | 1Byte | 1Byte | 1Byte | 2Byte | 2Byte | 1Byte | N  Bytes | 1Byte |

各字段定义：

根据STATUS和TagStatus判断命令是否执行成功。

|  |  |  |
| --- | --- | --- |
| 字段 | 字节 | 描述 |
| STATUS | 1 | 参考表格A-6 STATUS定义 |
| TagStatus | 1 | 标签返回的操作状态，具体定义见附录A； |
| Antenna | 1 | 从哪个天线端口接收到的标签数据，值范围为：1~4，分别表示1~4号天线 |
| CRC | 2 | 标签响应数据中的CRC数据 |
| PC | 2 | 标签响应数据中的PC数据 |
| EPC LEN | 1 | 标签的EPC号码长度（字节）； |
| EPC NUM | N Byte | 标签的EPC号码； |

### 2.4.7 RFM\_SETISO\_SELECTMASK (选择需要操作的标签)

该命令用于设置操作标签（读，写，锁和灭活）时所需要的标签EPC号码。

* 命令格式

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN |  | PAYLOAD | | CHECK |
| Pointer | Length | Mask |
| 0xCF | 0xFF | 0x0007 | 1Byte | 2Bytes | 1Byte | N-Bytes | 2Byte |

* + Pointer：保留，默认值为0x0000；
  + Length：需要匹配的EPC号码Bit位长度，默认值为0x00。
  + Mask：需要进行匹配的数据，有效数据长度为Length位，如果Length是奇数，则需在掩码的低位补0；
* 响应格式及状态字节

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| HEAD | ADDR | CMD | LEN | STATUS | CHECK |
| 0xCF | 0x00 | 0x0007 | 0x01 | 1Byte | 2Byte |

* + STATUS：
  1. 0x00：命令执行成功；
  2. 0x01：参数错误；
  3. 其他值：见目录C；

# 附录**A.** 标签返回的操作状态

标签返回的操作状态编码共8位，见表格A-1。

表格A-1 标签返回的操作状态

|  |  |  |  |
| --- | --- | --- | --- |
| 操作状态编码 | 操作状态 | 描述 | 错误优先级 |
| 0x81 | 其他错误 | 标签返回的未知错误 |  |
| 0x82 | 存储区溢出 | 目标存储区不存在 |  |
| 0x83 | 存储区锁定 | 对被锁定为不可写的存储区进行写操  作或者擦除操作，对被锁定为不可读的存储区进行读操作 |  |
| 0x84 | 功率不足 | 标签没有足够的能量完成操作 |  |
| 0x85 | 非特定错误 | 标签返回的未知错误 |  |
|  |  |  |  |

# 附录B.CRC16校验参考C代码

如下：

#define PRESET\_VALUE 0xFFFF

#define POLYNOMIAL 0x8408

unsigned int uiCrc16Cal(unsigned char const \* pucY, unsigned char ucX)

{

unsigned char ucI,ucJ;

unsigned short int uiCrcValue = PRESET\_VALUE;

for(ucI = 0; ucI < ucX; ucI++)

{

uiCrcValue = uiCrcValue ^ \*(pucY + ucI);

for(ucJ = 0; ucJ < 8; ucJ++)

{

if(uiCrcValue & 0x0001)

{

uiCrcValue = (uiCrcValue >> 1) ^ POLYNOMIAL;

}

else

{

uiCrcValue = (uiCrcValue >> 1);

}

}

}

return uiCrcValue;

}

# 附录C. STATUS的定义。

表格A-6 STATUS定义

|  |  |
| --- | --- |
| STATUS | 错误描述 |
| 0x00 | 执行成功（此处只表示模块成功接收到标签响应数据，如果标签响应中有标签执行状态，则还应该进一步判断标签执行状态是否正确） |
| 0x01 | 参数值错误或越界，或者模块不支持该参数值 |
| 0x02 | 由于模块内部错误导致的命令执行失败 |
| 0x03 | 保留 |
| 0x12 | 没有盘点到标签或整个盘点命令执行完成 |
| 0x14 | 标签响应超时 |
| 0x15 | 解调标签响应错误 |
| 0x16 | 协议认证失败 |
| 0x17 | 口令错误 |
| 0xFF | 没有更多数据了 |