

# 门锁通讯协议

V2.7

## 版本信息

| 版本   | 时间         | 更新内容   | 更新者               |
|------|------------|--|-------------------|
| V1.1 |            | 飞比与中控讨论版本  | Daniel            |
| V1.2 | 2016.08.29 | 所有协议增加回复，增加门锁主动获取时间<br>修改本地开锁上报（包含胁迫报警）  | Daniel            |
| V1.3 | 2016.09.04 | 增加启用开门常开/取消门锁常开/双人验证/胁迫报警<br>/验证进入管理员菜单等状态到本地开锁上报<br>兼容老的胁迫报警协议 0x25 命令<br>修改模块发送给门锁的常开命令到 0x86，原 0x53 | Parke             |
| V1.4 | 2016.09.29 | 远程开锁回复命令，增加开锁持续时间  | Parke             |
| V1.5 | 2016.11.30 | 修正老胁迫报警命令中用户 ID 解释错误问题<br>增加部分的详细备注描述，增加远程加密方法描述<br>增加报警解除的命令  | Ricky<br>Nicholas |
| V1.6 | 2017.05.15 | 增加门铃通讯协议。  | Ricky             |
| V1.7 | 2017.08.08 | 增加联动开锁协议。  | Ricky             |
| V1.8 | 2017.08.14 | 增加联动开锁校验码。   | Ricky             |
| V1.9 | 2018.05.21 | 增加密码操作协议   | Moon              |
| V2.0 | 2018.08.14 | 增加密钥信息上报协议、请求历史开锁记录协议、门锁<br>常用模式设置协议、锁键盘报警   | Moon              |
| V2.1 | 2018.08.15 | 增加其他用户非法操作报警   | Moon              |
| V2.2 | 2018.08.16 | 增加其他用户用户信息上报、密钥信息上报改为一次可<br>传 8 个  | Moon              |
| V2.3 | 2018.08.30 | 补充密码操作功能   | Moon              |
| V2.4 | 2018.11.23 | 添加门锁情景模块上报、设置  | Moon              |
| V2.5 | 2018.11.26 | 添加开锁记录、密钥信息的分包应答   | Moon              |
| V2.6 | 2018.12.27 | 整合开锁上报的类型，并扩展  | Moon              |
| V2.7 | 2019.01.18 | 扩展 0x40 的入网指令，增加锁版本号字段   | Moon              |
| V2.8 | 2019.03.04 | 增加动态密码种子下发及开锁上报  | Moon              |

## 目录

|          |                                 |          |
|----------|---------------------------------|----------|
| <b>1</b> | <b>接口定义.....</b>                | <b>4</b> |
| 1:       | VCC（电池正） .....                  | 4        |
| 2:       | GND（电池负） .....                  | 4        |
| 3:       | 唤醒输出（无线模块对锁主板的唤醒）P0.6 默认为低..... | 4        |
| 4:       | 唤醒输入（锁主板对无线模块的唤醒）P0.7 上拉.....   | 4        |
| 5:       | TXD（串口发送信号） .....               | 4        |
| 6:       | RXD（串口接收信号） .....               | 4        |
| <b>2</b> | <b>数据交互.....</b>                | <b>4</b> |
| <b>3</b> | <b>数据包格式.....</b>               | <b>5</b> |
| 3.1      | 发送数据包.....                      | 5        |
| 3.2      | 回复确认包.....                      | 6        |
| <b>4</b> | <b>具体通讯数据.....</b>              | <b>6</b> |
| 4.1      | 无线加网（门锁发送给模块） .....             | 6        |
| 4.2      | 无线退网（门锁发送给模块） .....             | 7        |
| 4.3      | 非法操作报警（门锁发送给模块） .....           | 7        |
| 4.4      | 强拆报警（门锁发送给模块） .....             | 8        |
| 4.5      | 假锁报警（门锁发送给模块） .....             | 8        |
| 4.6      | 未关门报警（门锁发送给模块） .....            | 9        |
| 4.7      | 胁迫报警（ 门锁发送给模块） .....            | 9        |
| 4.8      | 低压报警（门锁发送给模块） .....             | 10       |
| 4.9      | 本地开锁上报（门锁发送给模块） .....           | 10       |
| 4.10     | 门锁唤醒模块（门锁发送给模块） .....           | 12       |
| 4.11     | 远程密码开锁（模块发送给门锁） .....           | 12       |
| 4.12     | 远程密码开锁回复（门锁发送给模块） .....         | 13       |
| 4.13     | 时间同步（模块发送给门锁） .....             | 14       |
| 4.14     | 模块入网退网提示（模块发送给门锁） .....         | 14       |
| 4.15     | 模块网络状态显示（模块发送给门锁） .....         | 15       |
| 4.16     | 门锁获取时间（门锁发送给模块） .....           | 16       |
| 4.17     | 门锁常开（模块发送给门锁） .....             | 16       |
| 4.18     | 报警解除（门锁发送给模块） .....             | 17       |
| 4.19     | 门铃功能（门锁发送给模块） .....             | 18       |
| 4.20     | 联动开锁（模块发送给门锁） .....             | 18       |
| 4.21     | 联动开锁校验（门锁发送给模块） .....           | 19       |
| 4.22     | 密钥操作（模块发送给门锁） .....             | 20       |
| 4.23     | 请求密钥信息（模块发送给门锁） .....           | 22       |
| 4.24     | 请求历史开锁记录（模块发送给门锁） .....         | 23       |
| 4.25     | 门锁常用模式设置（模块发送给门锁） .....         | 24       |
| 4.26     | 锁键盘报警（门锁发送给模块） .....            | 25       |
| 4.27     | 其他用户非法操作报警（门锁发送给模块） .....       | 25       |

|      |                                  |    |
|------|----------------------------------|----|
| 4.28 | 用户信息变动上报（门锁发送给模块） .....          | 26 |
| 4.29 | 门锁本地常用模式设置上报（门锁发送给模块） .....      | 27 |
| 4.30 | 时间查询（模块发送给门锁） .....              | 28 |
| 4.31 | 指纹输入次数过多，禁止指纹输入报警（门锁发送给模块） ..... | 28 |
| 4.32 | 门内（内保险打开）反锁报警（门锁发送给模块） .....     | 29 |
| 4.33 | 门内（内保险关闭）解除反锁报警（门锁发送给模块） .....   | 30 |
| 4.34 | 卡片输入次数过多，禁止卡片输入报警（门锁发送给模块） ..... | 30 |
| 4.35 | 机械钥匙开锁，无用户编号（门锁发送给模块） .....      | 31 |
| 4.36 | 门锁激活（模块发给门锁） .....               | 31 |
| 4.36 | 用户种子下发（模块发给门锁） .....             | 32 |
| 4.37 | 删除用户种子（模块发给门锁） .....             | 32 |
| 4.38 | 查询用户种子 ID 列表（模块发给门锁） .....       | 33 |

## 1 接口定义

需提供 6 线接口：

1: VCC（电池正）

2: GND（电池负）

3: 唤醒输出（无线模块对锁主板的唤醒）P0.6 默认为低

4: 唤醒输入（锁主板对无线模块的唤醒）P0.7 上拉

5: TXD（串口发送信号）

6: RXD（串口接收信号）

通讯参数：波特率 57600，数据位 8，无校验，1 停止位，无流控。

唤醒信号线及串口都为 3.3V 电平。

考虑到功耗问题，唤醒信号线需上拉为高电平。

ZIGBEE 模块与电子锁主板采用串口通讯，进行数据交互。

ZIGBEE 模块，以下简称模块，门锁电路板，以下简称门锁。

## 2 数据交互

门锁发送数据给模块：

1) 门锁先给 WakeUp\_Mod 一个 10ms 的高电平来唤醒模块，20ms 之后在发送串口数据；

2) 模块接收到正确协议数据后，回复“确认指令”协议数据，若接收协议数据不正确，回复“确认错误指令”数据协议，再次等待门锁发送数据，若三次接收数据错误则自动进入

睡眠；

3) 门锁接收到“确认正确指令”协议数据后，进入休眠状态；若接收到“确认错误指令”协议数据或没有收到回复确认数据，则再次发送协议数据，三次重发，每次间隔 100ms，三次重发结束门锁也进入睡眠。（重发时不需要再发激活信号）

模块发送数据给门锁：

1) 模块先给 WakeUp\_MCU 一个 250ms 的高电平唤醒门锁之后直接发送数据，之后再每 800ms 向门锁发送串口数据，模块每次发送数据前均会发送激活信号 250ms；

2) 模块发送 7 次每 800ms 发送一次，若收到回复确认指令就停止发送进入睡眠，若失败或者没有收到则 7 次发送完成后模块进入睡眠。

### 3 数据包格式

协议定义如下：

协议数据为十六进制。

#### 3.1 发送数据包

|         |       |      |         |     |         |         |         |
|---------|-------|------|---------|-----|---------|---------|---------|
| 开始标记    | 有效数据  | 命 令  | 命令字自    | 应 答 | 有效数据(N) | 异或校验(1) | 结束标记    |
| 0xAA(1) | 长度(1) | 字(1) | 增 ID(4) | (1) |         |         | 0x55(1) |

开始标记：固定为 0xaa，二进制为 10101010，在电路上表现为整齐的峰谷峰谷波形，容易与干扰数据做区别。

命令字：执行操作的命令。

命令字自增 ID：每次命令自增 1，保证每条命令是唯一的，保证发出去的命令和收到的应答是对应的，范围 0x00000001~0xFFFFFFFF；注意此字段为（低字节在后，高字节在前）；

应答：0x00 表示发送数据包

有效数据长度：有效数据部分的字节数 N

有效数据：具体的操作数据(低字节在前，高字节在后)。

异或校验：对从”开始标记”开始到”有效数据”结束的内容进行异或计算的结果值。

结束标记：固定为 0x55，二进制为 1010101

3.2 回复确认包

|         |        |       |         |     |         |         |         |
|---------|--------|-------|---------|-----|---------|---------|---------|
| 开始标记    | 有效数据   | 命令    | 命令字自    | 应 答 | 有效数据(1) | 异或校验(1) | 结束标记    |
| 0xAA(1) | 长度 (1) | 字 (1) | 增 ID(4) | (1) |         |         | 0x55(1) |

开始标记：固定为 0xaa, 二进制为 10101010, 在电路上表现为整齐的峰谷峰谷波形，容易与干扰数据做区别。

命令字：执行操作的命令，与数据包的命令字一致。

命令字自增 ID：每次命令自增 1，保证每条命令是唯一的，保证发出去的命令和收到的应答是对应的，该 ID 与接收到的数据包 ID 保持一致（对应的数据包和确认包的 ID 一致）

应答：0x01 表示回复确认包

有效数据长度：有效数据部分的字节数，默认 0x01

有效数据：0x00 表示接收成功，0x01~0xFE 表示接收失败。

异或校验：对从”开始标记”开始到”有效数据”结束的内容进行异或计算的结果值。

结束标记：固定为 0x55，二进制为 1010101

4 具体通讯数据

4.1 无线加网（门锁发送给模块）

表 4-1-1 无线加网数据包

| 开始<br>标记<br>(1) | 有效数<br>据长度<br>(1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)                        | 异惑<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|-------------------|------------|-----------------|-----------|------------------------------------|-----------------|-------------|
| 0xAA            | 0x0A/<br>0x18     | 0x40       | 0x00 00 00 01   | 0x00      | 0x00 00 00 00 00 00 00 00 00 00 00 | 0xE1            | 0x55        |

命令字为 0x40，应答为 0x00，命令字自增从 0x01 开始累加，有效数据位全为 0。

第一~二字节：0x00 00~0xFFFF，表示厂商代码，填写为 0x11 80 将动态修改 APP 端显示的 SNID 名称。

第三字节：保留

第四、五字节：锁板版本号，这 2 个 ASCII 字符将显示为 SNID 的第 11、12 个字符，请用 ASCII 字符，其 SNDI 显示为 “FNB56-DOR23xx1.5”。

第六~九字节：保留

第十~二十四字节（可选字段）：若有 CMEI 码需要上报，则填为 15 字节 CMEI 码；若

无此需求请不要填写并将有效数据长度填为 0x0A。

表 4-1-2 无线加网回复包

| 开始<br>标记<br>(1) | 有效数<br>据长度<br>(1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异惑<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|-------------------|------------|-----------------|-----------|-------------|-----------------|-------------|
| 0xAA            | 0x01              | 0x40       | 0x00 00 00 01   | 0x01      | 0x00        | 0xEB            | 0x55        |

命令字，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位 0x00 表示数据接收成功，0x01~0xfe 表示接收数据失败。

## 4.2 无线退网（门锁发送给模块）

表 4-2-1 无线退网数据包

| 开始<br>标记<br>(1) | 有效数<br>据长度<br>(1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)                     | 异惑<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|-------------------|------------|-----------------|-----------|---------------------------------|-----------------|-------------|
| 0xAA            | 0x0A              | 0x41       | 0x00 00 00 01   | 0x00      | 0x00 00 00 00 00 00 00 00 00 00 | 0xE0            | 0x55        |

命令字为 0x41，应答为 0x00，命令字自增从 0x01 开始累加，有效数据位全为 0。

表 4-2-2 无线退网回复包

| 开始<br>标记<br>(1) | 有效数<br>据长度<br>(1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异惑<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|-------------------|------------|-----------------|-----------|-------------|-----------------|-------------|
| 0xAA            | 0x01              | 0x41       | 0x00 00 00 01   | 0x01      | 0x00        | 0xEA            | 0x55        |

命令字，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位 0x00 表示数据接收成功，0x01~0xfe 表示接收数据失败。

## 4.3 非法操作报警（门锁发送给模块）

表 4-3-1 非法操作数据包

| 开始<br>标记<br>(1) | 有效数<br>据长度<br>(1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)                     | 异惑<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|-------------------|------------|-----------------|-----------|---------------------------------|-----------------|-------------|
| 0xAA            | 0x0A              | 0x20       | 0x00 00 00 01   | 0x00      | 0x00 00 00 00 00 00 00 00 00 00 | 0x81            | 0x55        |

命令字为 0x20，应答为 0x00，命令字自增从 0x01 开始累加，有效数据位全为 0。

表 4-3-2 非法操作回复包

| 开始<br>标记<br>(1) | 有效数<br>据长度<br>(1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异惑<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|-------------------|------------|-----------------|-----------|-------------|-----------------|-------------|
| 0xAA            | 0x01              | 0x20       | 0x00 00 00 01   | 0x01      | 0x00        | 0x8B            | 0x55        |

命令字，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位 0x00 表示数据接收成功，0x01~0xfe 表示接收数据失败。

#### 4.4 强拆报警（门锁发送给模块）

表 4-4-1 强拆报警数据包

| 开始<br>标记<br>(1) | 有效数<br>据长度<br>(1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)                     | 异惑<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|-------------------|------------|-----------------|-----------|---------------------------------|-----------------|-------------|
| 0xAA            | 0x0A              | 0x22       | 0x00 00 00 01   | 0x00      | 0x00 00 00 00 00 00 00 00 00 00 | 0x83            | 0x55        |

命令字为 0x22，应答为 0x00，命令字自增从 0x01 开始累加，有效数据位全为 0。

表 4-4-2 强拆报警回复包

| 开始<br>标记<br>(1) | 有效数<br>据长度<br>(1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异 惑<br>校 验<br>(1) | 结束标记<br>(1) |
|-----------------|-------------------|------------|-----------------|-----------|-------------|-------------------|-------------|
| 0xAA            | 0x01              | 0x22       | 0x00 00 00 01   | 0x01      | 0x00        | 0x89              | 0x55        |

命令字，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位 0x00 表示数据接收成功，0x01~0xfe 表示接收数据失败。

#### 4.5 假锁报警（门锁发送给模块）

表 4-5-1 假锁报警数据包

| 开始<br>标记<br>(1) | 有效数<br>据长度<br>(1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)                     | 异惑<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|-------------------|------------|-----------------|-----------|---------------------------------|-----------------|-------------|
| 0xAA            | 0x0A              | 0x23       | 0x00 00 00 01   | 0x00      | 0x00 00 00 00 00 00 00 00 00 00 | 0x82            | 0x55        |

命令字为 0x23，应答为 0x00，命令字自增从 0x01 开始累加，有效数据位全为 0。

表 4-5-2 假锁报警回复包

| 开始<br>标记<br>(1) | 有效数<br>据长度<br>(1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异惑<br>校验<br>(1) | 结 束 标<br>记 (1) |
|-----------------|-------------------|------------|-----------------|-----------|-------------|-----------------|----------------|
| 0xAA            | 0x01              | 0x23       | 0x00 00 00 01   | 0x01      | 0x00        | 0x88            | 0x55           |

命令字，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位 0x00 表示



数据接收成功，0x01~0xfe 表示接收数据失败。

## 4.6 未关门报警（门锁发送给模块）

表 4-6-1 未关门数据包

| 开始<br>标记<br>(1) | 有效数<br>据长度<br>(1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)                     | 异惑<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|-------------------|------------|-----------------|-----------|---------------------------------|-----------------|-------------|
| 0xAA            | 0x0A              | 0x24       | 0x00 00 00 01   | 0x00      | 0x00 00 00 00 00 00 00 00 00 00 | 0x85            | 0x55        |

命令字为 0x24，应答为 0x00，命令字自增从 0x01 开始累加，有效数据位全为 0。

表 4-6-2 未关门回复包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异 惑<br>校 验<br>(1) | 结束标记<br>(1) |
|-----------------|----------------|------------|-----------------|-----------|-------------|-------------------|-------------|
| 0xAA            | 0x01           | 0x24       | 0x00 00 00 01   | 0x01      | 0x00        | 0x8F              | 0x55        |

命令字，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位 0x00 表示数据接收成功，0x01~0xfe 表示接收数据失败。

## 4.7 胁迫报警（ 门锁发送给模块）

表 4-7-1 胁迫报警数据包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)                     | 异惑<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|----------------|------------|-----------------|-----------|---------------------------------|-----------------|-------------|
| 0xAA            | 0x0A           | 0x25       | 0x00 00 00 01   | 0x00      | 0x00 01 03 01 00 00 00 00 00 00 | 0x87            | 0x55        |

命令字为 0x25， 应答为 0x00， 命令字自增从 0x01 开始累加， 有效数据位上传：用户编号 0x0100（ 2byte）（低字节在前，高位在后），验证方式(1byte， 1 为密码， 2 为卡， 3 为指纹)以及电量（ 1byte， 电量为 01-04， 表示电量从高到低， 04 代表低压， 作用同低压报警）。表中有效数据含义： 用户编号 0x0100， 验证方式 0x03， 电量 0x01， 其他数据保留默认为 0。（注：此条协议仅为兼容老版本智能锁，此协议有缺陷如没有用户编号，开锁时长，没有开锁时间等，并且发送此条后还必须发送一条本地开锁记录，因此将本条与本地开锁上报整合到一起，新锁设计请参考本地开锁上报中的胁迫报警协议，不再使用此条老协议）

表 4-7-2 胁迫报警回复包

| 开始标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异惑<br>校验<br>(1) | 结束标记<br>(1) |
|-------------|----------------|------------|-----------------|-----------|-------------|-----------------|-------------|
| 0xAA        | 0x01           | 0x25       | 0x00 00 00 01   | 0x01      | 0x00        | 0x8E            | 0x55        |

命令字，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位 0x00 表示数据接收成功，0x01~0xfe 表示接收数据失败。

## 4.8 低压报警（门锁发送给模块）

表 4-8-1 低压报警数据包

| 开始标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)                     | 异惑<br>校验<br>(1) | 结束标记<br>(1) |
|-------------|----------------|------------|-----------------|-----------|---------------------------------|-----------------|-------------|
| 0xAA        | 0x0A           | 0x30       | 0x00 00 00 01   | 0x00      | 0x00 00 00 00 00 00 00 00 00 00 | 0x91            | 0x55        |

命令字为 0x30，应答为 0x00，命令字自增从 0x01 开始累加，有效数据位全为 0。

表 4-8-2 低压报警回复包

| 开始标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异惑<br>校验<br>(1) | 结束标记<br>(1) |
|-------------|----------------|------------|-----------------|-----------|-------------|-----------------|-------------|
| 0xAA        | 0x01           | 0x30       | 0x00 00 00 01   | 0x01      | 0x00        | 0x9B            | 0x55        |

命令字，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位 0x00 表示数据接收成功，0x01~0xfe 表示接收数据失败。（此条可尽量与本地开锁上报一起发送）

## 4.9 本地开锁上报（门锁发送给模块）

表 4-9-1 本地开锁数据包

| 开始标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)                     | 异惑<br>校 验<br>(1) | 结束标记<br>(1) |
|-------------|----------------|------------|-----------------|-----------|---------------------------------|------------------|-------------|
| 0xAA        | 0x0A           | 0x80       | 0x00 00 00 01   | 0x00      | 0x01 00 02 04 05 98 c0 1b 7d 1f | 0x02             | 0x55        |

命令字为 0x80，应答为 0x00；命令字自增 ID 为 0x00 00 00 01（高位在前，低位在后）；

有效数据分别包括：

- 1、开锁的用户编号：0x00 01（2byte）（高位在后，低位在前）；

2、验证方式：0x02 (1byte, 0x01 为密码, 0x02 为卡, 0x03 为指纹, 0x04 多重验证, 0x05 为人脸, 0x06 为虹膜, 0x07 为指静脉, 0x08 为掌纹, 0x09 为掌静脉, 0x0A 为声纹识别, 0x0B 为 RF 感应, 0x0C 为蓝牙, 0x0D 为时效密码, 0x0E 为临时密码, 0x0F 为动态密码)

(多重验证: 是指要求用户开门时必须多次验证, 比如要求必须使用该用户的指纹和密码同时验证通过才能开锁);

3、电量等级: 0x04 (1byte, 电量为 01-04, 表示电量从高到低, 04 代表低压, 作用同低压报警);

4、开锁持续时间: 0x05 (1byte, 单位秒, 可以为 0x00) (为开锁后门锁隔多长时间自动关闭的时间, 模块会在此时间后自动发送门锁关闭数据包给网关, 如果此字节为 0x00: 则 Zigbee 模块不会自动发送门锁关闭数据包给网关, 节省电量);

5、门锁状态: 1byte, 0x98=1001 1000 位模式, 即验证进入管理员菜单时, 胁迫报警, 双人验证模式, 其他 Bit 定义如下:

Bit7: 胁迫报警;

Bit4: 双人验证模式;

Bit3: 验证管理员进入菜单;

Bit1: 取消门锁常开;

Bit0: 启用门锁常开;

例如: 0x98: 胁迫报警, 双人验证模式, 验证进入管理员菜单时;

0x11: 双人验证模式, 启用门锁常开模式;

0x12: 双人验证模式, 取消门锁常开模式;

0x88: 胁迫报警, 单人验证模式, 验证进入管理员菜单时;

0x90: 普通开门记录, 胁迫报警, 双人验证模式)

注: (启用门锁常开时: 开锁持续时间必须发 0x00, 模块不能再往网关发送已开锁信息; 取消门锁常开时: 门锁会在开锁持续时间后关锁, 可以通过管理员菜单取消门锁常开, 或在常开状态下使用正确的指纹密码卡验证成功后同时取消门锁常开状态) (双人验证模式: 门锁默认为单人验证模式, 如果用户选择使用双人验证模式, 则要求必须两个用户同时验证通过后方能开锁, 不限制用户验证方式, 但必须两个用户同时验证通过, 比如 A 用户指纹与 B 用户密码同时验证通过, 门锁发送两条开锁记录给模块, 但这两条开锁记录会标记为双人验证模式) (Bit3 置位代表此条信息不是开锁记录而是验证管理员进入菜单) (Bit1 取消与 Bit0 启用门锁常开不能同时置位);

6、时间戳 0x1F7D1BC0, (4byte, UTC 时间, 低位在前, 高位在后), 即 2016/08/26 12/00/00, UTC 时间转换, 具体转换公式请看附录 1。

表 4-9-2 本地开锁回复包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答(1) | 有效数据<br>(n) | 异感<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|----------------|------------|-----------------|-------|-------------|-----------------|-------------|
| 0xAA            | 0x01           | 0x80       | 0x00 00 00 01   | 0x01  | 0x00        | 0x2B            | 0x55        |

命令字, 命令字自增与接收到的协议数据包一致, 应答为 0x01, 有效数据位 0x00 表示数据接收成功, 0x01~0xfe 表示接收数据失败。

备注: 状态字节含义, 组合含义如下: (用户编号, 验证方式, 电量等按实际情况填写)

## 4.10 门锁唤醒模块 (门锁发送给模块)

表 4-10-1 门锁唤醒模块数据包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)                        | 异感<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|----------------|------------|-----------------|-----------|------------------------------------|-----------------|-------------|
| 0xAA            | 0x0A           | 0x77       | 0x00 00 00 01   | 0x00      | 0x00 00 00 00 00 00 00 00 00 00 00 | 0xD6            | 0x55        |

命令字为 0x77, 应答为 0x00, 命令字自增从 0x01 开始累加, 有效数据位全为 0。

在远程操作前, 需先唤醒模块才能远程操作, 模块唤醒后的一分钟内若接收到正确操作指令, 待处理完成操作命令 2 秒之后进入睡眠, 若没有收到任何操作则一分钟之后自动进入睡眠。

表 4-10-2 门锁唤醒模块回复包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异感<br>校验<br>(1) | 结束标<br>记 (1) |
|-----------------|----------------|------------|-----------------|-----------|-------------|-----------------|--------------|
| 0xAA            | 0x01           | 0x77       | 0x00 00 00 01   | 0x01      | 0x00        | 0xDC            | 0x55         |

命令字, 命令字自增与接收到的协议数据包一致, 应答为 0x01, 有效数据位 0x00 表示数据接收成功, 0x01~0xFE 表示接收数据失败。

## 4.11 远程密码开锁 (模块发送给门锁)

表 4-11-1 远程密码开锁数据包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异感<br>校验<br>(1) | 结束标<br>记 (1) |
|-----------------|----------------|------------|-----------------|-----------|-------------|-----------------|--------------|
|                 |                |            |                 |           |             |                 |              |

|      |      |      |               |      |                                 |      |      |
|------|------|------|---------------|------|---------------------------------|------|------|
| 0xAA | 0x0A | 0x60 | 0x00 00 00 01 | 0x00 | 0x01 02 03 04 05 06 00 00 00 00 | 0xC6 | 0x55 |
|------|------|------|---------------|------|---------------------------------|------|------|

命令字为 0x60，应答为 0x00，命令字自增从 0x01 开始累加，有效数据位：目前密码长度为 6byte，其他为默认为 0。模块及门锁接收到的 6 位密码是经过加密的，请参考本文档最后部分附录 2；

表 4-11-2 远程密码开锁回复包

| 开始标记 (1) | 有效数据长度 (1) | 命令字 (1) | 命令字自增 ID (4)  | 应答 (1) | 有效数据 (n) | 异惑校验 (1) | 结束标记 (1) |
|----------|------------|---------|---------------|--------|----------|----------|----------|
| 0xAA     | 0x01       | 0x60    | 0x00 00 00 01 | 0x01   | 0x00     | 0xCB     | 0x55     |

命令字，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位 0x00 表示数据接收成功，0x01~0xFE 表示接收数据失败。

远程开锁命令过程如下：

1. 门锁发送 0x77 命令唤醒模块，模块回复确认；
2. 模块发送 0x84 显示网络状态，门锁回复确认；
3. 模块醒来一分钟，等待 APP 端输入密码；
4. 然后 APP 端输入密码，发送给模块，模块再发送给门锁（0x60），门锁回复确认；
5. 门锁判断密码是否正确，发送远程开锁回复（0x54），模块回复确认。
6. 如果密码正确，模块直接进入休眠；否则，模块继续等待 APP 端的密码，直到超时。

## 4.12 远程密码开锁回复（门锁发送给模块）

表 4-12-1 远程密码开锁回复数据包

| 开始标记 (1) | 有效数据长度 (1) | 命令字 (1) | 命令字自增 ID (4)  | 应答 (1) | 有效数据 (n)                        | 异惑校验 (1) | 结束标记 (1) |
|----------|------------|---------|---------------|--------|---------------------------------|----------|----------|
| 0xAA     | 0x0A       | 0x54    | 0x00 00 00 01 | 0x00   | 0x00 01 00 00 00 00 00 00 00 00 | 0xF1     | 0x55     |

命令字为 0x54，应答为 0x00，命令字自增从 0x01 开始累加，有效数据位上传门锁状态 0x00(1byte，00 表示密码正确，01 为密码错误，02 为远程开锁未允许(锁本地拨码开关未打上))，用户编号 0x0001 (2byte)，保留 0x00 (1byte)，开锁持续时间 0x05 (1byte，单位秒，此字节直接填 0x00 代表不上传门锁开启时间)，其他字节都为 0x00。有效数据：远程开锁成功 (0x00)，用户编号 (0x0001)，保留字节 (0x00)，开锁持续时间 (0x05)；(备

注：除了模块在未收到确认包会重发外，网关为了可靠也会把此包发给模块及门锁 3 次（间隔 500ms 每次），因此对于此数据包，门锁回复确认包后可能还会收到模块的重发包，这些重发包都需要回复确认包，但不需要重复回复此条开锁成功与失败的数据包给模块，避免 APP 和服务器重复记录开锁数据）

表 4-12-2 远程密码开锁回复确认包

| 开始标记 (1) | 有效数据长度 (1) | 命令字 (1) | 命令字自增 ID (4)  | 应答 (1) | 有效数据 (n) | 异 惑 校 验 (1) | 结束标记 (1) |
|----------|------------|---------|---------------|--------|----------|-------------|----------|
| 0xAA     | 0x01       | 0x54    | 0x00 00 00 01 | 0x01   | 0x00     | 0xFF        | 0x55     |

命令字，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位 0x00 表示数据接收成功，0x01~0xFE 表示接收数据失败。

#### 4.13 时间同步（模块发送给门锁）

表 4-13-1 时间同步数据包

| 开始标记 (1) | 有效数据长度 (1) | 命令字 (1) | 命令字自增 ID (4)  | 应答 (1) | 有效数据 (n)                        | 异惑校验 (1) | 结 束 标 记 (1) |
|----------|------------|---------|---------------|--------|---------------------------------|----------|-------------|
| 0xAA     | 0x0A       | 0x62    | 0x00 00 00 01 | 0x00   | 0xE0 07 07 1A 0B 1E 2D 00 00 00 | 0x01     | 0x55        |

命令字为 0x62，应答为 0x00，命令字自增从 0x01 开始累加，有效数据位包括年(2byte)，月 (1byte)，日 (1byte)，时 (1byte)，分 (1byte)，秒 (1byte)。

有效数据：2016 年 (0x07E0)，7 月 (0x07)，26 日 (0x1A)，12 时 (0x0B)，30 分 (0x1E)，45 秒 (0x2D)。

表 4-13-2 时间同步回复包

| 开始标记 (1) | 有效数据长度 (1) | 命令字 (1) | 命令字自增 ID (4)  | 应答 (1) | 有效数据 (n) | 异惑校验 (1) | 结 束 标 记 (1) |
|----------|------------|---------|---------------|--------|----------|----------|-------------|
| 0xAA     | 0x01       | 0x62    | 0x00 00 00 01 | 0x01   | 0x00     | 0xC9     | 0x55        |

命令字，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位 0x00 表示数据接收成功，0x01~0xFE 表示接收数据失败。

备注：此条命令会在门锁向模块发送门锁获取时间命令后的几秒内回复。

#### 4.14 模块入网退网提示（模块发送给门锁）

表 4-14-1 模块入网退网提示数据包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)                     | 异惑校<br>验 (1) | 结束标<br>记 (1) |
|-----------------|----------------|------------|-----------------|-----------|---------------------------------|--------------|--------------|
| 0xAA            | 0x0A           | 0x83       | 0x00 00 00 01   | 0x00      | 0x01 01 00 00 00 00 00 00 00 00 | 0x22         | 0x55         |

命令字为 0x83，应答为 0x00，命令字自增从 0x01 开始累加，操作方法：1byte 有效数据位，入网操作或退网操作（1byte，0x00 入网，0x01 退网），反馈状态：1byte 有效数据，代表成功还是失败（1byte，0x00 成功，0x01 失败），其它字节默认为 0。有效数据位：退网操作 0x01，退网失败 0x01。

- ◆ 模块收到无线入网命令（4.1）后，会持续找网一分钟，一分钟之内若入网成功则会立即发送入网成功指令给门锁（有效数据 0x00 00 00 00 00 00 00 00 00 00）；若在一分钟之内没有入网成功则会在一分钟结束时发送入网失败指令（有效数据 0x00 01 00 00 00 00 00 00 00 00）；
- ◆ 模块在收到无线退网命令（4.2）后，会在 5 秒之后发送退网状态给门锁。（模块自己清除入网状态并自动重启，网关通过心跳判断是否在线，APP 显示离线需 1 小时以上；）

表 4-14-2 模块入网退网提示回复包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异惑校<br>验 (1) | 结束标<br>记 (1) |
|-----------------|----------------|------------|-----------------|-----------|-------------|--------------|--------------|
| 0xAA            | 0x01           | 0x83       | 0x00 00 00 01   | 0x01      | 0x00        | 0x28         | 0x55         |

命令字，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位 0x00 表示数据接收成功，0x01~0xFE 表示接收数据失败。

## 4.15 模块网络状态显示（模块发送给门锁）

表 4-15-1 模块网络状态提示数据包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)                     | 异惑校<br>验 (1) | 结束标<br>记 (1) |
|-----------------|----------------|------------|-----------------|-----------|---------------------------------|--------------|--------------|
| 0xAA            | 0x0A           | 0x84       | 0x00 00 00 01   | 0x00      | 0x00 00 00 00 00 00 00 00 00 00 | 0x25         | 0x55         |

命令字为 0x84，应答为 0x00，命令字自增从 0x01 开始累加，有效数据位上传网络状态（1byte，0x00 模块在线，0x01 模块掉线），其它字节默认为 0；  
模块在每次被唤醒和心跳时都会去检测网络是否在线，若在线这回发送在线命令给门锁，若掉线则会发送掉线命令。门锁可以将在线状态告知用户。

表 4-15-2 模块网络状态提示回复包

| 开始标记 (1) | 有效数据长度 (1) | 命令字 (1) | 命令字自增 ID (4)  | 应答 (1) | 有效数据 (n) | 异惑校验 (1) | 结束标记 (1) |
|----------|------------|---------|---------------|--------|----------|----------|----------|
| 0xAA     | 0x01       | 0x84    | 0x00 00 00 01 | 0x01   | 0x00     | 0x2F     | 0x55     |

命令字，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位 0x00 表示数据接收成功，0x01~0xFE 表示接收数据失败。

## 4.16 门锁获取时间（门锁发送给模块）

表 4-16-1 门锁获取时间数据包

| 开始标记 (1) | 有效数据长度 (1) | 命令字 (1) | 命令字自增 ID (4)  | 应答 (1) | 有效数据 (n)                        | 异惑校验 (1) | 结束标记 (1) |
|----------|------------|---------|---------------|--------|---------------------------------|----------|----------|
| 0xAA     | 0x0A       | 0x85    | 0x00 00 00 01 | 0x00   | 0x00 00 00 00 00 00 00 00 00 00 | 0x24     | 0x55     |

命令字为 0x85，应答为 0x00，命令字自增从 0x01 开始累加，有效数据位全为 0；模块在收到该命令后，会先从网关或服务器获取时间，然后把时间发送给门锁（4.13）。

表 4-16-2 门锁获取时间回复包

| 开始标记 (1) | 有效数据长度 (1) | 命令字 (1) | 命令字自增 ID (4)  | 应答 (1) | 有效数据 (n) | 异惑校验 (1) | 结束标记 (1) |
|----------|------------|---------|---------------|--------|----------|----------|----------|
| 0xAA     | 0x01       | 0x85    | 0x00 00 00 01 | 0x01   | 0x00     | 0x2E     | 0x55     |

命令字，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位 0x00 表示数据接收成功，0x01~0xFE 表示接收数据失败。

备注：门锁定期自动向模块发此指令，从模块及网关获取最新时间；确保本地时间正确；建议每星期同步一次时间，但门锁不需要主动醒来同步时间（可以在用户开锁时，判断上次时间同步是否超过 1 星期，如果超过 1 星期则在此次开锁成功后马上同步一次时间，同步时间成功后再发送开锁上报）

## 4.17 门锁常开（模块发送给门锁）

表 4-17-1 门锁常开数据包

| 开始标记 (1) | 有效数据长度 (1) | 命令字 (1) | 命令字自增 ID (4)  | 应答 (1) | 有效数据 (n)                        | 异惑校验 (1) | 结束标记 (1) |
|----------|------------|---------|---------------|--------|---------------------------------|----------|----------|
| 0xAA     | 0x0A       | 0x86    | 0x00 00 00 01 | 0x00   | 0xFE FF FF FF 00 00 00 00 00 00 | 0x26     | 0x55     |

命令字为 0x86，应答为 0x00，命令字自增从 0x01 开始累加；有效数据分别包括：门锁



保持开启的时间 (4byte) 高位在后 (0xFFFFFFFF 表示常开), 其他字节默认为 0; 有效数据: 0xFFFFFFFFFE 表示保持开门的时间; 此条数据备解释: 此条主要目的, 想通过 APP 设置门锁常开, APP 发送命令给网关及模块及门锁, 控制门锁处于常开多长时间; 实际不会产生 0xFFFFFFFFFE 这种数据, 但可能是保持常开状态半天.

表 4-17-2 门锁常开回复包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异惑校<br>验 (1) | 结束<br>标记<br>(1) |
|-----------------|----------------|------------|-----------------|-----------|-------------|--------------|-----------------|
| 0xAA            | 0x01           | 0x86       | 0x00 00 00 01   | 0x01      | 0x00        | 0x2D         | 0x55            |

命令字, 命令字自增与接收到的协议数据包一致, 应答为 0x01, 有效数据位 0x00 表示数据接收成功, 0x01~0xFE 表示接收数据失败。

## 4.18 报警解除 (门锁发送给模块)

表 4-18-1 报警解除数据包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)                        | 异惑校<br>验 (1) | 结束<br>标记<br>(1) |
|-----------------|----------------|------------|-----------------|-----------|------------------------------------|--------------|-----------------|
| 0xAA            | 0x01           | 0x29       | 0x00 00 00 01   | 0x00      | 0x87 00 00 00 00 00 00 00 00 00 00 | 0x0F         | 0x55            |

命令字为 0x29, 应答为 0x00, 命令字为 0x00000001, 有效数据分别包括: 解除假锁报警 0x87 (1byte, 0x84 解除防拆报警, 0x85 解除未关锁报警, 0x86 解除胁迫报警, 0x87 解除假锁报警, 0xB3 解除非法操作报警, 其他值暂时无意义), 其他字节全部默认为 0; 有效数据: 解除假锁报警发有效数据 (0x87 00 00 00 00 00 00 00 00 00 00), 解除非法操作报警发有效数据 (0xB3 00 00 00 00 00 00 00 00 00 00); (备注: 请模块朝网关发送数据时务必直接透传此字节; 门锁暂时只需要支持 0x85 及 0x87: 即解除未关锁报警及解除假锁报警)

表 4-18-2 报警解除回复包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异惑校<br>验 (1) | 结束<br>标记<br>(1) |
|-----------------|----------------|------------|-----------------|-----------|-------------|--------------|-----------------|
| 0xAA            | 0x01           | 0x29       | 0x00 00 00 01   | 0x01      | 0x00        | 0x82         | 0x55            |

命令字为 0x29，命令字自增字与接收到的数据包一致，应答为 0x01，有效数据位 0x00 表示数据接收成功，0x01~0xfe 表示接收数据失败。

## 4.19 门铃功能（门锁发送给模块）

表 4-19-1 门铃数据包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)                     | 异惑校<br>验 (1) | 结束<br>标记<br>(1) |
|-----------------|----------------|------------|-----------------|-----------|---------------------------------|--------------|-----------------|
| 0xAA            | 0x01           | 0x2A       | 0x00 00 00 01   | 0x00      | 0x01 3C 00 00 00 00 00 00 00 00 | 0xBD         | 0x55            |

命令字为 0x2A，应答为 0x00，命令字自增从 0x01 开始累加，有效数据分别包括：01 门铃类别，3C（0x3C）表示响铃周期 60S。

表 4-19-2 门铃回复包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异惑校<br>验 (1) | 结束<br>标记<br>(1) |
|-----------------|----------------|------------|-----------------|-----------|-------------|--------------|-----------------|
| 0xAA            | 0x01           | 0x2A       | 0x00 00 00 01   | 0x01      | 0x00        | 0x89         | 0x55            |

命令字为 0x29，命令字自增字与接收到的数据包一致，应答为 0x01，有效数据位 0x00 表示数据接收成功，0x01~0xFE 表示接收数据失败。

## 4.20 联动开锁（模块发送给门锁）

表 4-20-1 联动开锁数据包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)                     | 异惑校<br>验 (1) | 结束标<br>记 (1) |
|-----------------|----------------|------------|-----------------|-----------|---------------------------------|--------------|--------------|
| 0xAA            | 0x0A           | 0x66       | 0x00 00 00 01   | 0x00      | 0x53 79 4D 01 6F 72 74 68 00 00 | 0xC6         | 0x55         |

命令字为 0x66，应答为 0x00，命令字自增从 0x01 开始累加，有效数据位：数据共计 8 个字节，由自增验证码(四个字节)，与”SkyWorth”的前四个字节分别做异或运算。其中自增验证码为：0x00 12 34 56，数据内容：53^00 6B^12 79^34 57^56 6F 72 74 68。

要求自增验证码的数值必须每次增加，否则门锁端不响应本次开锁指令，并返回开锁失

败。后两字节保留(00 00)。

表 4-20-2 联动开锁回复包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 疑惑<br>校验<br>(1) | 结束标<br>记 (1) |
|-----------------|----------------|------------|-----------------|-----------|-------------|-----------------|--------------|
| 0xAA            | 0x01           | 0x66       | 0x00 00 00 01   | 0x01      | 0x00        | 0xCB            | 0x55         |

命令字，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位 0x00 表示数据接收成功，0x01~0xFE 表示开锁失败。

#### 4.21 联动开锁校验（门锁发送给模块）

表 4-21-1 联动开锁数据包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)                     | 疑惑<br>校验<br>(1) | 结束标<br>记 (1) |
|-----------------|----------------|------------|-----------------|-----------|---------------------------------|-----------------|--------------|
| 0xAA            | 0x0A           | 0x67       | 0x00 00 00 01   | 0x00      | 0x00 00 00 00 00 00 00 00 00 00 | 0xC6            | 0x55         |

命令字为 0x67，应答为 0x00，命令字自增从 0x01 开始累加，有效数据全为 0。

此命令由门锁端主动发起查询，随即无线模块先按照协议格式回复应答数据包，然后生成一个随机数作为联动开锁功能的初始验证码，再通过命令字 0x67 将联动开锁校验码返回给门锁。

门锁端需判断验证码是否符合标准(取值范围：0x00 00 00 00 85~0x00FB00FB，其中 bit0、bit3、bit7 位必须置 1，否则无效)，如若不符合要求，则命令无效；并保存此验证码用于联动开锁安全验证。

表 4-21-2 联动开锁校验回复包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)                     | 疑惑<br>校验<br>(1) | 结束标<br>记 (1) |
|-----------------|----------------|------------|-----------------|-----------|---------------------------------|-----------------|--------------|
| 0xAA            | 0x0A           | 0x67       | 0x00 00 00 01   | 0x00      | 0x00 01 00 85 00 00 00 00 00 00 | 0xCB            | 0x55         |

命令字，命令字自增与接收到的协议数据包一致，应答为 0x00。

有效数据位前四个字节表示联动开锁的初始验证码(0x00 01 00 85)，初始验证码取值范围：0x0000000085~0x00FB00FB，其中 bit0、bit3、bit7 位必须置 1，否则无效。

## 4.22 密钥操作（模块发送给门锁）

表 4-9-1 密码操作请求包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答 (1) | 有效数据<br>(n)   | 异惑校<br>验 (1) | 结束标<br>记<br>(1) |
|-----------------|----------------|------------|-----------------|--------|---|--------------|-----------------|
| 0xAA            | xx             | 0x73       | 0x00 00 00 01   | 0x00   | 0x00 01 0x12 04 0x11 22 33 44 0x55<br>66 77 88 0x01 0x02 0x06 0x01 02 03<br>04 05 06 0x06 0x01 02 03 04 05 06 | --           | 0x55            |

命令字为 0x73，应答为 0x00，命令字自增字为 0x00000001，有效数据部分表示：

第 1~2 字节表示：密码用户编号，密码操作时优先选用旧密码索引，当旧密码长度为 0 才选用密码编号索引；创建密码时不读取此字节。

第 3~4 字节表示：Lock code，密码类型，第 3 字节有数，第 4 字节保留，数据如下：

0x01=>管理密码。

0x02=>主人密码(又名普通密码，家庭成员密码)。

0x03=>客人密码。

0x04=>胁迫密码，(又名报警密码，用此密码开锁时会报警)。

0x05=>管理员感应卡。

0x06=>主人感应卡。

0x07=>客人感应卡。

0x08=>胁迫感应卡(又名报警感应卡，用此感应卡开锁时会报警)。

第 5~8 字节表示：密码有效的起始时间，全为 0 表示无时间限制。

第 9~12 字节表示：密码有效的终止时间，全为 0 表示无时间限制。

第 13 字节表示：condition，密码使用次数，0xFF 表示不限次数。

第 14 字节表示：Action，数值如下

0x01=>创建密码，可通过控制后台下发创建密码。

0x02=>修改密码，可通过控制后台修改指定密码。

0x03=>删除密码，失效密码自动删除，可通过控制后台删除指定密码。

0x04=>冻结密码，可通过控制后台冻结指定密码。

0x05=>解冻密码，可通过控制后台解冻指定密码。

0x06=>删除指纹，可通过控制后台删除指定指纹。

0x07=>删除感应卡，可通过控制后台删除指定感应卡。

第 15 字节往后为可变长的密钥数据部分。包括四部分，内容、数值如下：

- 1、旧密码长度或卡片数据长度，数值 0x06~0x0A；
- 2、旧密码数据或卡片数据，占 0 或 6~10 个字节；
- 3、新密码长度，修改密码时有效，数值 0x06~0x0A，卡片操作时无此数据；
- 4、新密码数据，修改密码时有效，占 0 或 6~10 个字节，卡片操作时无此数据。

表 4-9-2 密钥操作请求包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异惑校<br>验 (1) | 结束标<br>记<br>(1) |
|-----------------|----------------|------------|-----------------|-----------|-------------|--------------|-----------------|
| 0xAA            | 0x0D           | 0x73       | 0x00 00 00 01   | 0x01      | 0x00 01     | --           | 0x55            |

命令字为 0x76，命令字自增字与接收到的数据包一致，应答 0x01，有效数据表示如下：

第一个字节：

- 0x00 表示密码操作成功；
- 0x01 表示密码编号索引错误，查无此编号；
- 0x02 表示旧密码索引错误，查无此旧密码；
- 0x03 表示删除指纹操作成功；
- 0x04 表示删除密码指纹操作成功；
- 0x05 表示删除感应卡操作成功；
- 0x06 表示删除指纹操作失败；
- 0x07 表示删除密码指纹操作失败；
- 0x08 表示删除感应卡操作失败；
- 0x09 表示此用户类型不支持修改（管理员用户）；
- 0x0A 表示此用户类型不支持添加（管理员用户）；
- 0x0B 时效密码添加失败，生效时间戳大于结束时间戳；

第二个字节：表示密码编号；密码操作失败时填 0xFF。

第三~十二个字节：仅在应答添加临时密码时需填充，内容为已添加成功的临时密码，不足 10 位的填为 0xFF；应答其他操作时全部填为 0xFF。

## 4.23 请求密钥信息（模块发送给门锁）

表 4-23-1 请求密钥信息数据包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异感<br>校验<br>(1) | 结束标<br>记 (1) |
|-----------------|----------------|------------|-----------------|-----------|-------------|-----------------|--------------|
| 0xAA            | 0x02           | 0xC0       | 0x00 00 00 01   | 0x00      | 可变长         | 0xC6            | 0x55         |

命令字为 0xC0，应答为 0x00，命令字自增从 0x01 开始累加，有效数据为：

第一字节：密码类型：

0x01：常规密码

0x02：常规指纹

0x03：常规卡片

0x09：临时密码

0x0A：胁迫报警密码

0x0B：胁迫报警指纹

0x0C：胁迫报警卡片

0x0D：管理员密码

0x0E：管理员指纹

0x0F：管理员卡片

0xFF：所有密码类型，依次从常规密码上报到胁迫报警密码

第二字节：密钥信息编号，0xFF 表示请求所有的密钥信息；单独查询一条密钥信息是时填具体编号，取值在 0x00~0xFE 间。

表 4-23-2 请求密钥信息回复包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异感<br>校验<br>(1) | 结束标<br>记 (1) |
|-----------------|----------------|------------|-----------------|-----------|-------------|-----------------|--------------|
| 0xAA            | 0x0E           | 0xC0       | 0x00 00 00 01   | 0x01      | ~           | 0xCB            | 0x55         |

命令字，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位：

第一字节：密码总个数：0x00~0xFF

第二字节：指纹总个数：0x00~0xFF

第三字节：卡片总个数：0x00~0xFF

第四字节：密钥信息数据包序列号：0x00~0xFF

第五字节：第 1 个密钥类型，（0x01：常规密码...~..0x0F：管理员卡片）

第六字节：第 1 个密钥编号，0x00~0xFF

第七字节：第 2 个密钥类型，（0x01：常规密码...~..0x0F：管理员卡片）

第八字节：第 2 个密钥编号，0x00~0xFF

第九字节：第 3 个密钥类型，（0x01：常规密码...~..0x0F：管理员卡片）

第十字节：第 3 个密钥编号，0x00~0xFF

..... 直至

第十九字节：第 8 个密钥类型，（0x01：常规密码...~..0x0F：管理员卡片）

第二十字节：第 8 个密钥编号，0x00~0xFF

表 4-23-3 密钥信息接收成功，请发下一个包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异惑<br>校验<br>(1) | 结束标<br>记 (1) |
|-----------------|----------------|------------|-----------------|-----------|-------------|-----------------|--------------|
| 0xAA            | 0x02           | 0xC0       | 0x00 00 00 01   | 0x01      | 0xFF FF     | -               | 0x55         |

命令字，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位：0xFF FF  
表示成功接收到了上一个密钥信息包，可以继续发一下个密钥信息包（若密钥信息包尚未全部发送完毕）。

4.24 请求历史开锁记录（模块发送给门锁）

表 4-24-1 请求历史开锁记录

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异惑<br>校验<br>(1) | 结束标<br>记 (1) |
|-----------------|----------------|------------|-----------------|-----------|-------------|-----------------|--------------|
| 0xAA            | 0x01           | 0xC1       | 0x00 00 00 01   | 0x00      | ~           | 0xC6            | 0x55         |

命令字为 0xC0，应答为 0x00，命令字自增从 0x01 开始累加，有效数据为：  
第一字节：历史记录分段号 0x00~0xFE，以 20 条历史记录为一段，0x01 表示请求第一  
段的历史记录；0x02 表示请求第三段的历史记录；0x03 表示请求第三段的历史记录, 类推，  
0xFF 表示请求全部的历史开锁记录

表 4-24-2 请求历史开锁记录回复包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异惑<br>校验<br>(1) | 结束标<br>记 (1) |
|-----------------|----------------|------------|-----------------|-----------|-------------|-----------------|--------------|
|-----------------|----------------|------------|-----------------|-----------|-------------|-----------------|--------------|

|      |      |      |               |      |   |      |      |
|------|------|------|---------------|------|---|------|------|
| 0xAA | 0x0C | 0xC1 | 0x00 00 00 01 | 0x01 | ~ | 0xCB | 0x55 |
|------|------|------|---------------|------|---|------|------|

命令字，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位：

第一字节：当前历史记录总个数，0x00~0xFF

第二字节：当前历史记录序列号，0x00~0xFF

第三字节：第一个密钥类型，0x01：指纹、0x02：密码、0x03：感应卡

第四字节：第一个密钥编号

第五字节：第二个密钥类型，安全模式下填 0x01，其他模式下开锁填 0xFF

第六字节：第二个密钥编号，安全模式下填 0x01，其他模式下开锁填 0xFF

第七~十二字节：时间年+月+日+时+分+秒，如 18 年 8 月 14 号 17 时 27 分 20 秒为 18 08 14 17 27 20，共六个字节

表 4-24-3 开锁记录接收成功，请发下一个包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异感<br>校验<br>(1) | 结束标<br>记 (1) |
|-----------------|----------------|------------|-----------------|-----------|-------------|-----------------|--------------|
| 0xAA            | 0x02           | 0xC1       | 0x00 00 00 01   | 0x01      | 0xFF FF     | -               | 0x55         |

命令字，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位：0xFF FF

表示成功接收到了上一个密钥信息包，可以继续发一下个历史开锁记录包（若历史开锁记录包尚未全部发送完毕）。

## 4.25 门锁常用模式设置（模块发送给门锁）

表 4-25-1 门锁常用模式设置数据包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异感<br>校验<br>(1) | 结束标<br>记 (1) |
|-----------------|----------------|------------|-----------------|-----------|-------------|-----------------|--------------|
| 0xAA            | 0x04           | 0xC2       | 0x00 00 00 01   | 0x00      | ~           | 0xC6            | 0x55         |

命令字为 0xC0，应答为 0x00，命令字自增从 0x01 开始累加，有效数据为：

第一字节：0x00 常规模式；0x01 安全模式；0xFF 不做修改

第二字节：0x00 静音模式；0x01 英文模式；0xFF 不做修改

第三字节：0x00 红外设备开启；0x01 红外设备关闭；0xFF 不做修改

第四字节：0x00 蓝牙设备开启；0x01 蓝牙设备关闭；0xFF 不做修改

第五字节：0x00 居家模式开启；0x01 离家模式开启；0xFF 不做修改



表 4-25-2 门锁常用模式设置回复包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异感<br>校验<br>(1) | 结束标<br>记 (1) |
|-----------------|----------------|------------|-----------------|-----------|-------------|-----------------|--------------|
| 0xAA            | 0x05           | 0xC2       | 0x00 00 00 01   | 0x01      | ~           | 0xCB            | 0x55         |

命令字，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位：

第一字节：0x00 设置成功，0x01~0xFF 设置失败

第二字节：当前开锁认证模式，0x00 常规模式；0x01 安全模式

第三字节：当前语音模式，0x00 静音模式；0x01 英文模式

第四字节：当前红外模式，0x00 红外设备开启；0x01 红外设备关闭

第五字节：当前蓝牙模块，0x00 蓝牙设备开启；0x01 蓝牙设备关闭

第六字节：当前门锁情景模式，0x00 居家模式开启；0x01 离家模式开启；

## 4.26 锁键盘报警（门锁发送给模块）

表 4-26-1 锁键盘数据包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)                        | 异感<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|----------------|------------|-----------------|-----------|------------------------------------|-----------------|-------------|
| 0xAA            | 0x0A           | 0xC3       | 0x00 00 00 01   | 0x00      | 0x00 00 00 00 00 00 00 00 00 00 00 | 0x91            | 0x55        |

命令字为 0x30，应答为 0x00，命令字自增从 0x01 开始累加，有效数据位全为 0。

表 4-26-2 锁键盘回复包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异感<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|----------------|------------|-----------------|-----------|-------------|-----------------|-------------|
| 0xAA            | 0x01           | 0xC3       | 0x00 00 00 01   | 0x01      | 0x00        | 0x9B            | 0x55        |

命令字，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位 0x00 表示数据接收成功，0x01~0xFE 表示接收数据失败。

## 4.27 其他用户非法操作报警（门锁发送给模块）

表 4-27-1 非法操作数据包

| 开始<br>标记<br>(1) | 有效数<br>据长度<br>(1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)                     | 异惑<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|-------------------|------------|-----------------|-----------|---------------------------------|-----------------|-------------|
| 0xAA            | 0x0A              | 0xC4       | 0x00 00 00 01   | 0x00      | 0x00 00 00 00 00 00 00 00 00 00 | 0x81            | 0x55        |

命令字为：0xC4，应答为 0x00，命令字自增从 0x01 开始累加

有效数：1 字节：0x01:指纹验证多次失败上报；

0x02:密码验证多次失败上报；

0x03:感应卡验证多次失败上报；

表 4-27-2 非法操作回复包

| 开始<br>标记<br>(1) | 有效数<br>据长度<br>(1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异惑<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|-------------------|------------|-----------------|-----------|-------------|-----------------|-------------|
| 0xAA            | 0x01              | 0xC4       | 0x00 00 00 01   | 0x01      | 0x00        | 0x8B            | 0x55        |

命令字，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位 0x00 表示数据接收成功，0x01~0xfe 表示接收数据失败。

## 4.28 用户信息变动上报（门锁发送给模块）

表 4-28-1 用户信息变动数据包

| 开始<br>标记<br>(1) | 有效数<br>据长度<br>(1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)   | 异惑<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|-------------------|------------|-----------------|-----------|---------------|-----------------|-------------|
| 0xAA            | 0x04              | 0xC5       | 0x00 00 00 01   | 0x00      | 0x00 00 00 00 | 0x81            | 0x55        |

命令字为：0xC4，应答为 0x00，命令字自增从 0x01 开始累加有效数：

第一字节：0x01：表示添加用户；

0x02：删除用户；

0x03：表示清空用户；

0x04：表示初始化用户；

第二字节：0x01：表示指纹；

0x02：表示密码；

0x03：表示卡片；

0x04：表示临时密码；

第三字节：0x01：表示管理用户；

0x02：表示普通用户；

0x03：表示报警用户；

第四字节：0x00~0xFF：表示编号；

表 4-28-2 用户信息变动回复包

| 开始<br>标记<br>(1) | 有效数<br>据长度<br>(1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异惑<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|-------------------|------------|-----------------|-----------|-------------|-----------------|-------------|
| 0xAA            | 0x01              | 0xC5       | 0x00 00 00 01   | 0x01      | 0x00        | 0x8B            | 0x55        |

命令字 0xC5，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位 0x00 表示数据接收成功，0x01~0xfe 表示接收数据失败。

#### 4.29 门锁本地常用模式设置上报（门锁发送给模块）

表 4-29-1 门锁常用模式上报数据包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异惑<br>校验<br>(1) | 结束标<br>记 (1) |
|-----------------|----------------|------------|-----------------|-----------|-------------|-----------------|--------------|
| 0xAA            | 0x05           | 0xC6       | 0x00 00 00 01   | 0x00      | ~           | 0xC6            | 0x55         |

命令字 0xC6，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位：

第一字节：填 00，保留位

第二字节：当前开锁认证模式，0x00 常规模式；0x01 安全模式；0xFF 无此功能

第三字节：当前语音模式，0x00 静音模式；0x01 英文模式；0xFF 无此功能

第四字节：当前红外模式，0x00 红外设备开启；0x01 红外设备关闭；0xFF 无此功能

第五字节：当前蓝牙模块，0x00 蓝牙设备开启；0x01 蓝牙设备关闭；0xFF 无此功能

第六字节：当前门锁情景模式，0x00 居家模式开启；0x01 离家模块开启；0xFF 无此功能

能

表 4-25-2 门锁常用模式上报回复包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异感<br>校验<br>(1) | 结束标<br>记 (1) |
|-----------------|----------------|------------|-----------------|-----------|-------------|-----------------|--------------|
| 0xAA            | 0x01           | 0xC6       | 0x00 00 00 01   | 0x01      | 00          | 0xCB            | 0x55         |

命令字 0xC6，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位 0x00 表示数据接收成功，0x01~0xFE 表示接收数据失败。

### 4.30 时间查询（模块发送给门锁）

表 4-30-1 非法操作数据包

| 开始<br>标记<br>(1) | 有效数<br>据长度<br>(1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)                     | 异感<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|-------------------|------------|-----------------|-----------|---------------------------------|-----------------|-------------|
| 0xAA            | 0x0A              | 0xC7       | 0x00 00 00 01   | 0x00      | 0x00 00 00 00 00 00 00 00 00 00 | 0x81            | 0x55        |

命令字为：0xC4，应答为 0x00，命令字自增从 0x01 开始累加

表 4-30-2 非法操作回复包

| 开始<br>标记<br>(1) | 有效数<br>据长度<br>(1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)                   | 异感<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|-------------------|------------|-----------------|-----------|-------------------------------|-----------------|-------------|
| 0xAA            | 0x0A              | 0xC7       | 0x00 00 00 01   | 0x01      | E0 07 07 1A 0B 1E 2D 00 00 00 | 0x8B            | 0x55        |

命令字，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位有效数据位包括年（2byte），月（1byte），日（1byte），时（1byte），分（1byte），秒（1byte）。

如：有效数据 2016 年（0x07E0），7 月（0x07），26 日（0x1A），12 时（0x0B），30 分（0x1E），45 秒（0x2D）。

### 4.31 指纹输入次数过多，禁止指纹输入报警（门锁发送给模块）

表 4-26-1 禁止指纹输入数据包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)                     | 异感<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|----------------|------------|-----------------|-----------|---------------------------------|-----------------|-------------|
| 0xAA            | 0x0A           | 0xC8       | 0x00 00 00 01   | 0x00      | 0x00 00 00 00 00 00 00 00 00 00 | 0x91            | 0x55        |

命令字为 0xC8，应答为 0x00，命令字自增从 0x01 开始累加，有效数据位全为 0。

表 4-26-2 禁止指纹输入回复包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异感<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|----------------|------------|-----------------|-----------|-------------|-----------------|-------------|
| 0xAA            | 0x01           | 0xC8       | 0x00 00 00 01   | 0x01      | 0x00        | 0x9B            | 0x55        |

命令字，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位 0x00 表示数据接收成功，0x01~0xfe 表示接收数据失败。

## 4.32 门内（内保险打开）反锁报警（门锁发送给模块）

表 4-32-1 开启反锁数据包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)                     | 异感<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|----------------|------------|-----------------|-----------|---------------------------------|-----------------|-------------|
| 0xAA            | 0x0A           | 0xC9       | 0x00 00 00 01   | 0x00      | 0x00 00 00 00 00 00 00 00 00 00 | 0x91            | 0x55        |

命令字为 0xC9，应答为 0x00，命令字自增从 0x01 开始累加，有效数据位全为 0。

表 4-32-2 开启反锁回复包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异感<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|----------------|------------|-----------------|-----------|-------------|-----------------|-------------|
| 0xAA            | 0x01           | 0xC9       | 0x00 00 00 01   | 0x01      | 0x00        | 0x9B            | 0x55        |

命令字，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位 0x00 表示数据接收成功，0x01~0xfe 表示接收数据失败。

#### 4.33 门内（内保险关闭）解除反锁报警（门锁发送给模块）

表 4-33-1 解除反锁数据包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)                        | 异惑<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|----------------|------------|-----------------|-----------|------------------------------------|-----------------|-------------|
| 0xAA            | 0x0A           | 0xCA       | 0x00 00 00 01   | 0x00      | 0x00 00 00 00 00 00 00 00 00 00 00 | 0x91            | 0x55        |

命令字为 0xCA，应答为 0x00，命令字自增从 0x01 开始累加，有效数据位全为 0。

表 4-33-2 解除反锁回复包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异惑<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|----------------|------------|-----------------|-----------|-------------|-----------------|-------------|
| 0xAA            | 0x01           | 0xCA       | 0x00 00 00 01   | 0x01      | 0x00        | 0x9B            | 0x55        |

命令字，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位 0x00 表示数据接收成功，0x01~0xfe 表示接收数据失败。

#### 4.34 卡片输入次数过多，禁止卡片输入报警（门锁发送给模块）

表 4-34-1 禁止卡片输入数据包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)                        | 异惑<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|----------------|------------|-----------------|-----------|------------------------------------|-----------------|-------------|
| 0xAA            | 0x0A           | 0xCB       | 0x00 00 00 01   | 0x00      | 0x00 00 00 00 00 00 00 00 00 00 00 | 0x91            | 0x55        |

命令字为 0xCB，应答为 0x00，命令字自增从 0x01 开始累加，有效数据位全为 0。

表 4-34-2 禁止卡片输入回复包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异惑<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|----------------|------------|-----------------|-----------|-------------|-----------------|-------------|
| 0xAA            | 0x01           | 0xCB       | 0x00 00 00 01   | 0x01      | 0x00        | 0x9B            | 0x55        |

命令字，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位 0x00 表示数据接收成功，0x01~0xfe 表示接收数据失败。

#### 4.35 机械钥匙开锁，无用户编号（门锁发送给模块）

表 4-35-1 机械钥匙开锁数据包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)                        | 异惑<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|----------------|------------|-----------------|-----------|------------------------------------|-----------------|-------------|
| 0xAA            | 0x0A           | 0xCC       | 0x00 00 00 01   | 0x00      | 0x00 00 00 00 00 00 00 00 00 00 00 | 0x91            | 0x55        |

命令字为 0xCC，应答为 0x00，命令字自增从 0x01 开始累加，有效数据位全为 0。

表 4-35-2 机械钥匙开锁回复包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异惑<br>校验<br>(1) | 结束标记<br>(1) |
|-----------------|----------------|------------|-----------------|-----------|-------------|-----------------|-------------|
| 0xAA            | 0x01           | 0xCC       | 0x00 00 00 01   | 0x01      | 0x00        | 0x9B            | 0x55        |

命令字，命令字自增与接收到的协议数据包一致，应答为 0x01，有效数据位 0x00 表示数据接收成功，0x01~0xFE 表示接收数据失败。

#### 4.36 门锁激活（模块发给门锁）

表 4-36-1 门锁激活

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)               | 异惑校<br>验 (1) | 结束标<br>记<br>(1) |
|-----------------|----------------|------------|-----------------|-----------|---------------------------|--------------|-----------------|
| 0xAA            | 0x08           | 0x74       | 0x00 00 00 01   | 0x00      | 0x01 02 03 04 05 06 07 08 | --           | 0x55            |

模块发给门锁命令字为 0x74，命令字自增字与接收到的数据包一致，应答 0x01，有效数据为 0x01 02 03 04 05 06 07 08 表示密码激活密码

表 4-36-2 门锁回复包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异惑校<br>验 (1) | 结束标<br>记<br>(1) |
|-----------------|----------------|------------|-----------------|-----------|-------------|--------------|-----------------|
| 0xAA            | 0x02           | 0x74       | 0x00 00 00 01   | 0x01      | 0x01 01     | --           | 0x55            |

门锁发给模块，命令字为 0x74，命令字自增字与接收到的数据包一致，应答 0x01，有效数据为

第一字节：0x00 表示门锁激动成功，其他数据表示门锁激活失败；

第二字节：0x00~0xFF 表示激活密码在锁上编号

4.36 用户种子下发（模块发给门锁）

表 4-37-1 用户种子下发

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)  | 异惑校<br>验 (1) | 结束标<br>记<br>(1) |
|-----------------|----------------|------------|-----------------|-----------|--|--------------|-----------------|
| 0xAA            | 0x1C           | 0xCD       | 0x00 00 00 01   | 0x00      | 0x01 02 03 04 05 06 07<br>0x02 02 03 04 05 06 07<br>0x03 02 03 04 05 06 07<br>0xFF FF FF FF FF FF FF | --           | 0x55            |

模块发给门锁命令字为 0xCD，命令字自增字与接收到的数据包一致，应答 0x01，用户种子的格式为“用户种子 ID（1 字节）”+“用户种子数据（6 字节）”，一次性包含 4 个用户种子，没有 4 个种子时剩余的字节用 0xFF 填充：例  
0x01 02 03 04 05 06 07：种子 ID 为 0x01，种子数据为 02 03 04 05 06 07；  
0x02 02 03 04 05 06 07：种子 ID 为 0x02，种子数据为 02 03 04 05 06 07；  
0x03 02 03 04 05 06 07：种子 ID 为 0x03，种子数据为 02 03 04 05 06 07；  
0xFF FF FF FF FF FF FF：全部填为 0xFF 表示无效数据  
注：用户种子 ID 取值为 1~10；用户种子数据中任何一字节都不能是 0xFF。

表 4-37-2 门锁回复包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)   | 异惑校<br>验 (1) | 结束标<br>记<br>(1) |
|-----------------|----------------|------------|-----------------|-----------|---------------|--------------|-----------------|
| 0xAA            | 0x04           | 0xCD       | 0x00 00 00 01   | 0x01      | 0x00 00 00 00 | --           | 0x55            |

门锁发给模块，命令字为 0xCD，命令字自增字与接收到的数据包一致，应答 0x01，有效数据为  
第一字节：0x00 表示第一个种子添加成功；0x01 种子已在归为添加成功；0x02 添加失败；  
第二字节：0x00 表示第二个种子添加成功；0x01 种子已在归为添加成功；0x02 添加失败；  
第三字节：0x00 表示第三个种子添加成功；0x01 种子已在归为添加成功；0x02 添加失败；  
第四字节：0xFF 表示无；

4.37 删除用户种子（模块发给门锁）

表 4-37-1 删除用户种子

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)   | 异惑校<br>验 (1) | 结束标<br>记<br>(1) |
|-----------------|----------------|------------|-----------------|-----------|---------------|--------------|-----------------|
| 0xAA            | 0x04           | 0xCE       | 0x00 00 00 01   | 0x00      | 0x01 02 03 FF | --           | 0x55            |

模块发给门锁命令字为 0xCE，命令字自增字与接收到的数据包一致，应答 0x01，一性最多删除 4 个种子，有效数据为  
第一字节：01 表示删除种子 ID 为 0x01 的用户种子；  
第二字节：02 表示删除种子 ID 为 0x02 的用户种子；  
第三字节：03 表示删除种子 ID 为 0x03 的用户种子；



第四字节：0xFF 表示无效数据；

表 4-37-2 门锁回复包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)   | 异惑校<br>验 (1) | 结束标<br>记<br>(1) |
|-----------------|----------------|------------|-----------------|-----------|---------------|--------------|-----------------|
| 0xAA            | 0x04           | 0xCE       | 0x00 00 00 01   | 0x01      | 0x00 00 00 00 | --           | 0x55            |

门锁发给模块, 命令字为 0xCE, 命令字自增字与接收到的数据包一致, 应答 0x01, 有效数据为

第一字节：0x00 表示第一个种子删除成功；0x01 种子 ID 不存在，删除失败；  
第二字节：0x00 表示第二个种子删除成功；0x01 种子 ID 不存在，删除失败；  
第三字节：0x00 表示第三个种子删除成功；0x01 种子 ID 不存在，删除失败；  
第四字节：0xFF 表示无效数据；

4.38 查询用户种子 ID 列表（模块发给门锁）

表 4-38-1 查询用户种子 ID 列表

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n) | 异惑校<br>验 (1) | 结束标<br>记<br>(1) |
|-----------------|----------------|------------|-----------------|-----------|-------------|--------------|-----------------|
| 0xAA            | 0x01           | 0xCF       | 0x00 00 00 01   | 0x00      | 0x00        | --           | 0x55            |

模块发给门锁命令字为 0xCF, 命令字自增字与接收到的数据包一致, 应答 0x01, 有效数据为 0x00;

表 4-38-2 门锁回复包

| 开始<br>标记<br>(1) | 有效数据<br>长度 (1) | 命令字<br>(1) | 命令字自增 ID<br>(4) | 应答<br>(1) | 有效数据<br>(n)            | 异惑校<br>验 (1) | 结束标<br>记<br>(1) |
|-----------------|----------------|------------|-----------------|-----------|------------------------|--------------|-----------------|
| 0xAA            | 0x01~0x14      | 0xCF       | 0x00 00 00 01   | 0x01      | 0x01 02 03 04 ..... 14 | --           | 0x55            |

门锁发给模块, 命令字为 0xCF, 命令字自增字与接收到的数据包一致, 应答 0x01, 有效数据为:

- 1、当没有用户种子时，有效数据长度填 0x01, 有效数据为 0xFF;
- 2、当有用户种子时，有效数据长度填为“用户种子总个数”，有效数据填上各个用户种子的 ID，最多包含 20 个用户种子 ID。

附录 1

UTC 时间转换公式:

```

#define YearLength(yr)  ((uint16)(IsLeapYear(yr) ? 366 : 365))

#define IsLeapYear(yr)  (!((yr) % 400) || (((yr) % 100) && !((yr) % 4)))

#define BEGYEAR (2000)

#define DAY          (60*60*24)


static uint8 monthLength( uint8 lpyr, uint8 mon )
{
    uint8 days = 31;

    if ( mon == 1 ) // feb
    {
        days = ( 28 + lpyr );
    }
    else
    {
        if ( mon > 6 ) // aug-dec
        {
            mon--;
        }

        if ( mon & 1 )
        {
            days = 30;
        }
    }

    return ( days );
}

typedef struct
{

```

```

uint8 seconds; // 0-59
uint8 minutes; // 0-59
uint8 hour;    // 0-23
uint8 day;     // 0-30
uint8 month;   // 0-11
uint16 year;   // 2000+
} UTCTimeStruct;

```

## 1. UTC 时间转换成本地时间

```

void osal_ConvertUTCTime( UTCTimeStruct *tm, UTCTime secTime )//转换公式
{
    // calculate the time less than a day - hours, minutes, seconds
    {
        uint32 day = secTime % DAY;
        tm->seconds = day % 60UL;
        tm->minutes = (day % 3600UL) / 60UL;
        tm->hour = day / 3600UL;
    }

    // Fill in the calendar - day, month, year
    {
        uint16 numDays = secTime / DAY;
        tm->year = BEGYEAR;
        while ( numDays >= YearLength( tm->year ) )
        {
            numDays -= YearLength( tm->year );
            tm->year++;
        }

        tm->month = 0;
    }
}

```

```

        while ( numDays >= monthLength( IsLeapYear( tm->year ), tm->month ) )
        {
            numDays -= monthLength( IsLeapYear( tm->year ), tm->month );
            tm->month++;
        }

        tm->day = numDays;
    }
}

```

## 2. 本地时间转换成 UTC 时间

```

UTCtime osal_ConvertUTCsecs( UTCtimeStruct *tm )
{
    uint32 seconds;

    /* Seconds for the partial day */
    seconds = (((tm->hour * 60UL) + tm->minutes) * 60UL) + tm->seconds;

    /* Account for previous complete days */
    {
        /* Start with complete days in current month */
        uint16 days = tm->day;

        /* Next, complete months in current year */
        {
            int8 month = tm->month;
            while ( --month >= 0 )
            {
                days += monthLength( IsLeapYear( tm->year ), month );
            }
        }
    }
}

```

```

/* Next, complete years before current year */
{
    uint16 year = tm->year;
    while ( --year >= BEGYEAR )
    {
        days += YearLength( year );
    }
}

/* Add total seconds before partial day */
seconds += (days * DAY);
}

return ( seconds );
}

```

## 附录 2

### 远程开锁密码加密解密方法

门锁将收到的 6 字节密码数据与 `key[6] = {0x46, 0x45, 0x49, 0x42, 0x49, 0x47,}` 进行异或运算，所得结果即为真正的密码。加密程序是相同的过程，也是同这 6 个字节做异或运算。APP 往门锁发送数据时的加密函数方法如下：

```

Byte key[] = {0x46, 0x45, 0x49, 0x42, 0x49, 0x47};//加密密钥
Byte npass[6];

for (int i = 0; i < 6; i ++)
{

```

```
    npass[i] = (Byte) (key[i] ^ password[i]);  
}
```

-----The

End-----