
HTTP 通讯协议文档_V1.0.0

修订记录

| 版本 | 修改日期 | 修改人 | 修改内容 |
|---------------|------------|-----|------|
| V1.0.0 | 2021/08/18 | 常永彬 | 新建文档 |

目录

| | |
|---------------------------------|----|
| HTTP 通讯协议文档_V1.0.0..... | 1 |
| 一、 协议说明..... | 3 |
| 二、 普通透传模式..... | 4 |
| 三、 协议透传模式..... | 4 |
| 3.1 协议通道为 HTTP 时数据格式..... | 4 |
| 3.2 协议通道为 485/TCP 时数据格式..... | 5 |
| 四、 二次开发模式..... | 7 |
| 4.1 指令 0x53 卡号上报开关..... | 8 |
| 4.2 M1 卡快操作..... | 9 |
| 4.2.1 指令 0x51 读取 M1 卡一块数据..... | 9 |
| 4.2.2 指令 0x52 写 M1 卡一块数据..... | 10 |
| 4.2.3 指令 0xA0 读 M1 卡扇区内多个块..... | 11 |
| 4.2.4 指令 0xA1 写 M1 卡扇区内多个块..... | 12 |
| 4.3 ISO14443-4..... | 14 |
| 4.3.1 指令 0xA6 发送 APDU 指令..... | 14 |

一、协议说明

本协议旨在说明 VF102 的三种不同工作模式下设备与上位机交互的方式。

三种不同工作模式

- 1.普通透传模式，支持 TCP、485、HTTP、韦根，详见《二、普通透传模式》
- 2.协议透传模式，支持 TCP、485、HTTP，详见《三、协议透传模式》
- 3.二次开发模式，支持 TCP、485，详见《四、二次开发模式》

二、普通透传模式

刷脸成功透传人员 ID，扫码成功透传二维码原数据，刷卡成功透传卡号

三、协议透传模式

3.1 协议通道为 HTTP 时数据格式

编码格式：UTF-8

请求地址：http://ip:port/xxx

请求方式：POST

提交数据格式：application/json

请求参数

| 参数名 | 必填 | 类型 | 说明 |
|----------|----|--------|----------------------------|
| type | 是 | Int | 数据类型 1：刷脸 2：扫码 3：刷卡 |
| data | 是 | String | 扫描二维码获取到的文本信息、刷卡卡号和刷脸人员 ID |
| time | 是 | Long | 扫码或刷卡时间（时间戳） |
| deviceSn | 是 | String | 设备序列号（设备地址） |

响应

| 参数名 | 必填 | 类型 | 说明 |
|---------|----|--------|-----------------------------|
| code | 是 | Int | code < 0：失败 code >= 0：成功 |
| cmd | 否 | Int | 1：输出继电器 |
| message | 否 | String | 响应描述 |

示例：

请求参数：

```
{  
  "type":1,  
  "data":"test data",  
  "time":1625465218,  
  "deviceSn":"aabbccddeeff"  
}
```

响应：

```
{  
  "code":0,  
  "message":"success"  
}
```

3.2 协议通道为 485/TCP 时数据格式

发出数据包格式

| 项目 | 字节 | 值 | 说明 |
|-------|----|-----------|---|
| 数据帧头 | 2 | 0x55 0xAA | 数据包包头固定为 0x55 0xAA |
| 命令字 | 1 | 0x30 | 数据包命令字固定为 0x30 |
| 数据域长度 | 2 | | 传输用户数据得长度（指明本条命令从长度字后面开始到校验字的字节数（不含校验字），低位在前） |
| 数据域 | n | | 用户数据，数据格式为 JSON 字符串，详见下表《用户数据格式》 |
| 校验和 | | | 校验为 BCC 异或校验（从命令头到数据域最后一字节的逐字节异或值） |

应答数据包格式

| 项目 | 字节 | 值 | 说明 |
|------|----|-----------|---|
| 数据帧头 | 2 | 0x55 0xAA | 数据包包头固定为 0x550xAA |
| 命令字 | 1 | 0x30 | 数据包命令字 |
| 结果字 | 1 | 0x00/0xff | 0x00: 成功 0xff: 失败 |
| 数据长度 | 2 | | 传输用户应答数据得长度（指明本条命令从长度字后面开始到校验字的字节数（不含校验字），低位在前） |
| 数据域 | n | | 用户数据，数据格式为 JSON 字符串，详见下表《用户应答数据格式》 |
| 校验字 | 1 | | 校验为 BCC 异或校验（从命令头到数据域最后一字节的逐字节异或值） |

用户数据格式

| 参数名 | 必填 | 类型 | 说明 |
|----------|----|--------|----------------------------|
| type | 是 | Int | 数据类型 1: 刷脸 2: 扫码 3: 刷卡 |
| data | 是 | String | 扫描二维码获取到的文本信息、刷卡卡号和刷脸人员 ID |
| time | 是 | Long | 扫码或刷卡时间（时间戳） |
| deviceSn | 是 | String | 设备序列号（设备地址） |

用户应答数据格式

| 参数名 | 必填 | 类型 | 说明 |
|---------|----|--------|--------------------------------|
| code | 是 | Int | code < 0 : 失败 code >= 0: 成功 |
| cmd | 否 | Int | 1: 输出继电器 |
| message | 否 | String | 响应描述 |

示例 1
刷卡上报

55 aa 30 45 00 7b 22 74 79 70 65 22 3a 33 2c 22 64 61 74 61 22 3a 22 39 33 61 37 39
32 66 64 22 2c 22 74 69 6d 65 22 3a 31 36 32 39 34 32 31 39 36 33 2c 22 64 65 76 69
63 65 53 6e 22 3a 22 61 73 64 66 67 68 6a 31 31 22 7d 8c

应答:

55 AA 30 00 1E 00 7b 22 63 6f 64 65 22 3a 30 2c 22 6d 65 73 73 61 67 65 22 3a 22
73 75 63 63 65 73 73 22 7d CE

示例 2: 扫码上报

55 aa 30 68 00 7b 22 74 79 70 65 22 3a 32 2c 22 64 61 74 61 22 3a 22 68 74 74 70 3a
2f 2f 77 65 69 78 69 6e 2e 71 71 2e 63 6f 6d 2f 72 2f 56 7a 6b 48 48 37 54 45 31 5f
6d 6f 72 66 42 38 39 32 78 46 22 2c 22 74 69 6d 65 22 3a 31 36 32 39 34 32 32 31 34
34 2c 22 64 65 76 69 63 65 53 6e 22 3a 22 61 73 64 66 67 68 6a 31 31 22 7d 98

应答:

55 AA 30 00 1E 00 7b 22 63 6f 64 65 22 3a 30 2c 22 6d 65 73 73 61 67 65 22 3a 22
73 75 63 63 65 73 73 22 7d CE

四、二次开发模式

二次开发模式当前只包函 NFC 模块相关操作，NFC 模块可支持 Mifare One 卡块读写、CPU 卡发送 APDU 指令，详见具体指令。

名词解释：

任务启动标志位--该标志位用于告知扫码器何时开始卡的操作，何时结束卡的操作，或告知扫码器操作卡的指令是独立的，无指令间依赖关系。

该标志位用来设置卡片的操作环境，标志位的值有以下三种：

0x00->AUTO 告知扫码器该指令可单独执行，无指令间的依赖关系。

0x01->START 告知扫码器开始对卡操作或对卡操作尚未结束，且指令间可能存在依赖关系。

0x02->FINISH 告知扫码器本条指令是操作卡的最后一条指令，将卡片操作环境恢复到默态。

若操作卡的指令是独立的，如读写 M1 卡的某块数据，则该标志位即可设置为 AUTO 也可设置为 FINISH.

注：

1、若使用 **START** 标识开始操作卡片，则必须使用 **FINISH** 标识结束操作，否则会导致 NFC 模块工作异常，需重启后才可再次使用。

2、若卡片操作过程中涉及多条卡片操作指令，则过程中所发指令的任务启动标志位均为 **START**，最后一条指令标记为 **FINISH**。

4.1 指令 0x53 数据上报开关

| 指令: 0x53 | | | | | |
|---|--------|--|---------------------|--------|---------------------|
| 说明: 数据域的值设置为 0x01 或 0x00 时 (即进入或退出命令模式), 均为空操作且扫描器回复成功。 | | | | | |
| 注: 数据上报格式参见《3.2》, 若关闭数据上报功能, 将收不到扫码、刷脸、刷卡数据 | | | | | |
| PC->Reader(Send) | | | Reader->PC(Receive) | | |
| 项目 | 字节 | 说明 | 项目 | 字节 | 说明 |
| 包头 | 2Byte | Default: 0x55 0xAA | 包头 | 1Byte | Default: 0x55 0xAA |
| 命令字 | 1Byte | 0x53 | 命令字 | 1Byte | 0x53 |
| 数据域长度 | 2 Byte | 0x01 0x00 | 标识字 | 1Byte | 0x00: 成功 非 0: 失败 |
| 数据域 | 1 Byte | 0x01: 模块进入命令模式 | 数据域长度 | 2 Byte | N |
| | | 0x00: 模块退出命令模式 0x02: 刷卡上报 0x03: 关闭上报 | 数据域 | N Byte | 数据 N = 0 时没有此项 |
| 校验字 | 1 Byte | | 校验字 | 1 Byte | |

例:

PC-->Reader :55 AA 53 01 00 02 AF 使能卡号上报

PC-->Reader :55 AA 53 01 00 03 AE 关闭卡号上报

Reader-->PC :55 AA 53 00 00 00 AC

4.2 M1 卡块操作

4.2.1 指令 0x51 读取 M1 卡一块数据

| | | | | | | | |
|---|--------|---------------|---------------------|---|---------------------|--------|-----------------------|
| 指令: 0x51 | | | 读取 M1 卡某块 | | | | |
| 说明: 任务启动标志字段可选, 当指令中不包含该标志位时, 默认按 AUTO 标识执行 | | | | | | | |
| PC->Reader(Send) | | | Reader->PC(Receive) | | | | |
| 项目 | 字节 | 说明 | 项目 | 字节 | 说明 | | |
| 包头 | 2Byte | 固定: 0x55 0xAA | 包头 | 1Byte | 0x55 0xAA | | |
| 命令字 | 1Byte | 0x51 | 命令字 | 1Byte | 0x51 | | |
| 数据域长度 | 2 Byte | N | 标识字 | 1Byte | 0x00: 成功 非 0: 失败 | | |
| 数据域 | N Byte | 密钥类型 | 1Byte | 0x60 -> KEY A 0x61 -> KEY B | 数据域长度 | 2 Byte | N |
| | | 块号 | 1 Byte | 0 ~ 0xFF | | | |
| | | 密钥 | 6 Byte | | 数据域 | N Byte | 数据 N = 0 时没有 此项 |
| | | 任务启动标志位(可选) | 1 Byte | 0x00 -> AUTO 0x01 -> START 0x02 -> FINISH | | | |
| 校验字 | 1 Byte | | 校验字 | 1Byte | | | |

例:

用 A (0x60) 密钥做认证, 读取 6 扇区第二块(即绝对块号为 0x19) 数据。

认证密钥为 FF FF FF FF FF FF, 标志位选配。

PC-->Reader :55AA 51 0900 60 19 FF FF FF FF FF FF 00 DE 包含标志位

PC-->Reader :55AA 51 0800 60 19 FF FF FF FF FF FF DF 无标志位

Reader-->PC :55AA 51 00 1000 12 34 56 78 90 12 34 56 78 90 12 34 56 78 90 12 34 读卡成功

Reader-->PC :55AA 51 FF 0000 51 失败或无卡

4.2.2 指令 0x52 写 M1 卡一块数据

| 指令: 0x52 | | | 向 M1 卡某块写数据 | | | | |
|---|--------|---|---------------------|--------------------------------|-------|--------------------------|-----------------------|
| 说明: 任务启动标志字段可选, 当指令中不包含该标志位时, 默认按 AUTO 标识执行 | | | | | | | |
| PC->Reader(Send) | | | Reader->PC(Receive) | | | | |
| 项目 | 字节 | 说明 | | 项目 | 字节 | 说明 | |
| 包头 | 2 Byte | Default: 0x55 0xAA | | 包头 | 1Byte | Default: 0x55 0xAA | |
| 命令字 | 1 Byte | 0x52 | | 命令字 | 1Byte | 0x52 | |
| 数据域长度 | 2 Byte | N | | 标识字 | 1Byte | 0x00: 成功 非 0: 失败 | |
| 数据域 | N Byte | 密钥类型 | 1 Byte | 0x60 -> KEY A 0x61 -> KEY B | 数据域长度 | 2Byte | N |
| | | 块号 | 1 Byte | 0 ~ 0xFF | | | |
| | | 密钥 | 6 Byte | | 数据域 | NByte | 数据 N = 0 时没有 此项 |
| | | 数据 | 16 Byte | | | | |
| 任务标志位(可选) | 1 Byte | 0x00 -> AUTO 0x01 -> START 0x02 -> FINISH | | | | | |
| 校验字 | 1 Byte | | | 校验字 | 1Byte | | |

例:

用 B (0x61) 密钥做认证, 向 6 扇区第二块(即绝对块号为 0x19) 写数据。

认证密钥为 FF FF FF FF FF FF, 标志位选配。

PC-->Reader :55AA 52 1900 61 19 FF FF FF FF FF FF 11 11 11 11 11 11 11 22 22 22 22
22 22 22 22 00 CC 包含标志位

PC-->Reader :55AA 52 1800 61 19 FF FF FF FF FF FF 12 34 56 78 90 12 34 56 12 34 56 78
90 12 34 56 CD 无标志位

Reader-->PC :55AA 52 00 0000 AD 写数据成功

Reader-->PC :55AA 52 FF 0000 52 失败或无卡

4.2.4 指令 0xA1 写 M1 卡扇区内多个块

| | | | | | | | | | | | |
|--|--------|--------------------|---|---|-----|---------------------|----------------|---------------------|--------|-----|--|
| 指令: 0xA1 | 写多块数据 | | | | | | | | | | |
| <p>说明: 可读 S50/S70 卡, 扇区号、偏移、块数的取值根据卡片类型而定</p> <p>偏移--以选取的扇区 0 块为起始地址计算待写块的基地址。</p> <p>块数--以选定的基地址块为写卡开始块, 连续写数据到选中的块数。</p> <p>命令解析:</p> <p>写数据到一张卡的 2 扇区 1 块和 2 块 (指令详见示例)</p> <p>55 AA A1 2B 00 00 60 02 01 02 FF FF FF FF FF FF 36</p> | | | | | | | | | | | |
| 55 AA | A1 | 2B 00 | 00 | 60 | 02 | 01 | 02 | FF ~FF | | 36 | |
| 命令头 | 指令 | 数据长度 | AUTO | 密钥类型 | 扇区号 | 待写块基地址 | 从基地址开始连续写几块 | 密钥 | 待写数据 | 校验字 | |
| <p>注: 待写入的块数不可为 0, 若为 0 视为无效指令; 一条指令中不可跨扇区写入数据</p> | | | | | | | | | | | |
| PC->Reader(Send) | | | | | | Reader->PC(Receive) | | | | | |
| 项目 | 字节 | 说明 | | | | 项目 | 字节 | 说明 | | | |
| 包头 | 2 Byte | Default: 0x55 0xAA | | | | 包头 | 1 Byte | Default: 0x55 0xAA | | | |
| 命令字 | 1 Byte | 0xA1 | | | | 命令字 | 1 Byte | 0xA1 | | | |
| 数据域长度 | 2 Byte | N | | | | 标识字 | 1 Byte | 0x00: 成功 非 0: 失败 | | | |
| 数据域 | N Byte | 任务标志位 | 1 Byte | 0x00 -> AUTO 0x01 -> START 0x02 -> FINISH | | 数据域长度 | 2 Byte | N | | | |
| | | 密钥类型 | 1 Byte | 0x60 -> KEY A 0x61 -> KEY B | | | | | | | |
| | | 扇区号 | 1 Byte | S50 -> 0x00~0x0F S70 -> 0x00~0x27 | | | | | | | |
| | 偏移 | 1 Byte | S50 -> 0x00~0x03 S70 -> 0x00~0x03 或 0x00~0x0F | | 数据域 | 0 Byte | 数据 N = 0 时没有此项 | | | | |
| | 块数 | 1 Byte | S50 -> 0x01~0x04 S70 -> 0x01~0x04 或 0x01~0x10 | | | | | | | | |
| | 密钥 | 6 Byte | | | | | | | | | |
| 数据 | N Byte | N = 16 * 块数 | | | | | | | | | |
| 校验字 | 1 Byte | | | | | 校验字 | 1 Byte | | | | |

例:

用 A (0x60) 密钥做认证, 向 2 扇区 1 块、2 块写入数据, 即以 1 块为基地址连续写 2 块。

认证密钥为 FF FF FF FF FF FF,标志位设置为 AUTO。

PC-->Reader :55 AA A1 2B 00 00 60 02 01 02 FF FF FF FF FF FF 11 11 11 11 11 11 11 11
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 33 33 33 33 33 33 33 33 36

Reader-->PC : 55 AA A1 00 00 00 5E 写数据成功

Reader-->PC :55 AA A1 FF 00 00 A1 失败或无卡

4.3 ISO14443-4

4.3.1 指令 0xA6 发送 APDU 指令

| | | | | | | |
|--|-----------|--------------------|-----------|---------------------------------|-----------|-----------|
| 指令: 0xA6 | | | | | | |
| 说明: 用于与 CPU 卡之间进行通信, APDU 指令可参见《FMCOS2.0 用户手册》 | | | | | | |
| PC->Reader(Send) | | | | Reader->PC(Receive) | | |
| 项目 | 字节 | 说明 | | 项目 | 字节 | |
| 包头 | 2 Byte | Default: 0x55 0xAA | | 包头 | 1 Byte | |
| 命令字 | 1 Byte | 0xA6 | | 命令字 | 1 Byte | |
| 数据域长度 | 2 Byte | N | | 标识字 | 1 Byte | |
| 数据域 | N Byte | 任务标志位 | 1 Byte | 0x01 -> START 0x02 -> FINISH | 数据域长度 | 2 Byte |
| | | APDU DATA | N Byte | 符合 ISO7816-4 的数据结构 | 数据域 | N Byte |
| 校验字 | 1 Byte | | | 校验字 | 1 Byte | |

例: 红色字体部分为 APDU 指令

选择应用目录:

PC-->Reader : 55 AA A6 08 00 01 00 A4 00 00 02 3F 01 C8

Reader-->PC : 55 AA A6 00 11 00 6F 0D 84 05 41 44 46 30 31 A5 04 9F 08 01 02 90 00 4C

获取 4 位随机数:

Reader-->PC : 55 AA A6 06 00 01 00 84 00 00 04 DE

Reader-->PC : 55 AA A6 00 06 00 7C C9 56 38 90 00 14

外部认证: 四位随机数用于外部认证, 认证方式为 DES 单倍长, 默认密钥(1122334455667788)

PC-->Reader : 55 AA A6 0E 00 01 00 82 00 00 08 71 7E B1 7D 4C F6 81 17 33

Reader-->PC : 55 AA A6 00 02 00 90 00 CB

选择二进制文件:

PC-->Reader : 55 AA A6 06 00 02 00 B0 83 00 00 6E

Reader-->PC : 55 AA A6 00 12 00 11 22 33 44 55 66 77 88 00 00 00 00 00 00 00 90 00 53

五、人员注册

人员注册需要在协议透传模式或开发模式下才会生效。

5.1 指令 0x90 拍照注册

| | | | | | |
|-------------------------|--------|-----------------------------------|---------------------|--------|---------------------|
| 指令: 0x90 | | | | | |
| 说明: 用于协议透传模式或开发模式下的拍照注册 | | | | | |
| PC->Reader(Send) | | | Reader->PC(Receive) | | |
| 项目 | 字节 | 说明 | 项目 | 字节 | 说明 |
| 包头 | 2 Byte | Default: 0x55 0xAA | 包头 | 1 Byte | Default: 0x55 0xAA |
| 命令字 | 1 Byte | 0x90 | 命令字 | 1 Byte | 0x90 |
| 数据域长度 | 2 Byte | N | 标识字 | 1 Byte | 0x00: 成功 非 0: 失败 |
| 数据域 | N Byte | 数据为 json 格式的字符串详见表《拍照注册 json 数据表》 | 数据域长度 | 2 Byte | N |
| | | | 数据域 | N Byte | 数据 N = 0 时没有此项 |
| 校验字 | 1 Byte | | 校验字 | 1 Byte | |

拍照注册 json 数据表

| 参数名 | 描述 | 类型 | 必传 | 附加说明 |
|-----------------------|-------|--------|----|--|
| option | 类型 | Int | Y | 1: 注册, 2: 删除 |
| employeeNumber | 人员 Id | String | Y | 员工号, E.g SW200 为删除时: 若删除多个人员, id 用英文逗号拼接传入-1 则删除所有人员 • 以下数据会被删除, 本地将不再做存储: 该人员 ID、人员信息 该人员对应的识别记录、现场抓拍照片 该人员对应的注册照片 • 人员 ID(id)只允许数字-1, 0~9 和英文字母, 且最大长度为 255 |
| name | 人员名称 | String | Y | name 参数必传且内容不可为空。长度、内容无限制。 |
| gender | 性别 | String | N | Gender(E.g "Man/Women/...") |
| nationa | 国家 | String | N | |
| departmentName | 部门名称 | String | N | |
| idCardNumber | 人员卡号 | String | N | 注册时可以不填, 长度、内容无限制。若注册人员时填写了卡号, 可直接刷对应卡号的卡进行识别, 屏幕也会显示与该卡号对应的人员的名字。 |
| mobile | 手机号 | String | N | 人员手机号 |

| | | | | |
|---------------------------------|-------|--------|---|---|
| accessRight | 权限模式 | Int | N | 0: 自由通行 1: 临时通行, 模式 1 2: 临时通行模式 2 |
| temporaryAccessStartTime | 开始时间戳 | Int | N | 对应临时通行模式 1 |
| temporaryAccessEndTime | 结束时间戳 | Int | N | 对应临时通行模式 1 |
| temporaryAccessTimes | 通行次数 | Int | N | 对应临时通行模式 2 |
| accessCardNumber | 门禁卡号 | String | N | 门禁卡号, 识别模式为刷卡时有效 |
| remarks | 备注 | String | N | 用于填写说明信息 |

例: 注册人员信息如下

姓名: 彬

ID: 10001

① 添加人员

```
{
  "option": 1,
  "employeeNumber": "10001",
  "name": "彬"
}
```

开启拍照注册:

```
PC-->Reader :55 AA 90 3200 7b 22 6f 70 74 69 6f 6e 22 3a 31 2c 22 65 6d 70
6c 6f 79 65 65 4e 75 6d 62 65 72 22 3a 22 31 30 30 30 31 22 2c 22 6e 61 6d 65 22 3a
22 e5 bd ac 22 7d B1
```

```
Reader-->PC : 55AA 90 00 0000 6F
```

② 删除人员

```
{
  "option": 2,
  "employeeNumber": "10001"
}
```

```
PC-->Reader :55 AA 90 2500 7b 22 6f 70 74 69 6f 6e 22 3a 32 2c 22 65 6d 70 6c
6f 79 65 65 4e 75 6d 62 65 72 22 3a 22 31 30 30 30 31 22 7d 40
```

```
Reader-->PC : 55AA 90 00 0000 6F
```

③ 删除所有人员

```
{
  "option": 2,
  "employeeNumber": "-1"
}
```

```
PC-->Reader :55 AA 90 2200 7b 22 6f 70 74 69 6f 6e 22 3a 32 2c 22 65 6d 70 6c
6f 79 65 65 4e 75 6d 62 65 72 22 3a 22 2d 31 22 7d 6B
```

```
Reader-->PC : 55AA 90 00 0000 6F
```

附录 1:

| 标识号 | 说明 |
|------|-------------|
| 0x00 | 成功 |
| 0x10 | 成功, 结果不为空 |
| 0x90 | 失败 |
| 0x01 | 校验失败 |
| 0x02 | 数据长度越界 |
| 0x03 | 本指令不支持 |
| 0x04 | Json 解析失败 |
| 0x05 | 内存不足 |
| 0x06 | 密码长度错误 |
| 0x07 | 密码校验错误 |
| 0x08 | 功能未开启 |
| 0x09 | 卡号长度越界 |
| 0x0A | 升级超时失败 |
| 0x0B | 写入 flash 失败 |
| 0x0C | 包序号错误 |
| 0x0D | 不支持的压缩方式 |
| 0x0E | 参数错误 |