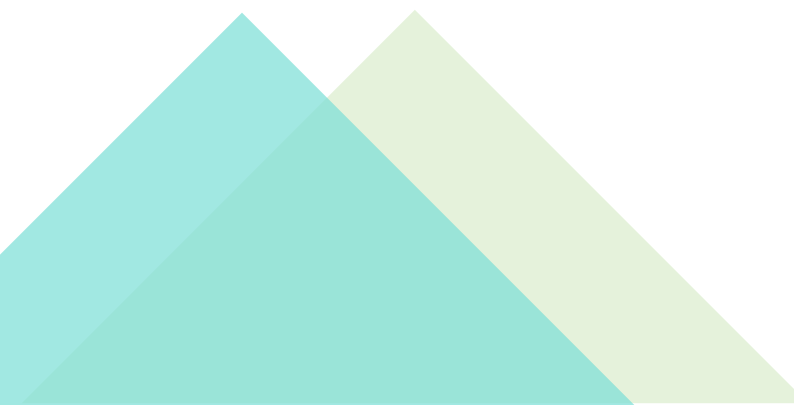




Unit-Fingerprint

控制协议



目录

1、通信协议结构	4
1.1 通信协议参数	4
1.1.1 内部 MCU 与指纹模块的通信参数	4
1.1.2 外部通信接口的通信参数	4
1.2 数据包格式	4
1.2.1 命令包	4
1.2.2 数据包	5
1.2.3 应答包（返回包）	5
2、软件开发指南	5
2.1 参数表（内部指纹模组）	5
2.2 ROM 及传感器驱动	10
2.3 设备地址	10
2.4 Unit-Fingerprint 工作状态	10
2.4.1 定时休眠模式	10
2.4.2 开启模式	10
2.5 Unit-Fingerprint 内部指纹模组关闭状态	11
3、指令集定义	11
3.1 通用指令集	11
3.1.1 验证用获取图像 PS_GetImage	12
3.1.2 注册用获取图像 PS_GetEnrollImage	12
3.1.3 生成特征 PS_GenChar	13
3.1.4 精确比对两枚指纹特征 PS_Match	13
3.1.5 搜索指纹 PS_Search	14
3.1.6 合并特征（生成模板）PS_RegModel	14
3.1.7 储存模板 PS_StoreChar	15
3.1.8 读出模板 PS_LoadChar	15
3.1.9 上传图像 PS_UpImage	16
3.1.10 删除模板 PS_DeletChar	17
3.1.11 清空指纹库 PS_Empty	17
3.1.12 写系统寄存器 PS_WriteReg	18
3.1.13 读系统基本参数 PS_ReadSysPara	18
3.1.14 采样随机数 PS_GetRandomCode	19
3.1.15 读 flash 信息页 PS_ReadINFpage	20
3.1.16 写记事本 PS_WriteNotepad	20
3.1.17 读记事本 PS_ReadNotepad	21
3.1.18 读有效模板个数 PS_ValidTemplateNum	21
3.1.19 读索引表 PS_ReadIndexTable	22
3.1.20 获取芯片唯一序列号 PS_GetChipSN	22
3.1.21 握手指令 PS_HandShake	23
3.1.22 校验传感器 PS_CheckSensor	23
3.1.23 LED 控制灯指令 PS_ControlBLN	24
3.1.24 获取图像信息指令 PS_GetImageInfo	25
3.1.25 搜索当前指纹指令 PS_SearchNow	25
3.1.26 特殊上传模板 PS_UpTemplet	26
3.1.27 特殊下载模板 PS_DownTemplet	27
3.1.28 设置休眠时间 PS_SetSleepTime	27
3.1.29 获取休眠时间 PS_GeSleepTime	28
3.1.30 设置工作模式 PS_SetWorkMode	28
3.1.31 获取工作模式 PS_GetWorkMode	29
3.1.32 激活 Unit-Fingerprint 内部指纹模组 PS_ActivateFingerprintModule	29
3.1.33 获取指纹模组工作状态 PS_GetFingerprintModuleStatus	30

3.1.34 保存配置信息到 flash PS_SaveConfigurationToFlash	30
3.1.35 获取 STM32 固件版本 PS_GetFirmwareVersion	31
3.2 模块化指令集	31
3.2.1 取消指令 PS_Cancel	31
3.2.2 自动注册模板 PS_AutoEnroll	32
3.2.3 自动验证指纹 PS_AutoIdentify	34
4、功能演示示例	36
4.1 UART 命令包的处理过程	36
4.2 UART 数据包的发送过程	37
4.3 UART 数据包的接收过程	38
4.4 注册指纹流程	39
4.5 搜索指纹流程	40
4.6 主控加载一个指纹特征或者模板进行精确比对	41
4.7 特殊上传模板流程	42
4.8 特殊下载模板流程	43
5、返回类型表	44
6. CRC 计算方法示例（C 语言）	45
6.1 校验	45
6.2 计算	46

1、通信协议结构

1.1 通信协议参数

1.1.1 内部 MCU 与指纹模块的通信参数

采用半双工异步串行通讯。

默认波特率为 57600bps。（禁止修改）

数据格式：8 位数据位（低位在前）和 2 位停止位，无校验位。

内部指纹模组与 STM32 芯片通信接口。

1.1.2 外部通信接口的通信参数

采用半双工异步串行通讯。

波特率为 115200bps。

数据格式：8 位数据位（低位在前）和 1 位停止位，无校验位。

ST 芯片外部通信接口，既用户控制接口。

1.2 数据包格式

1.2.1 命令包

包头	芯片地址	包标识	包长度	指令	参数 1	...	参数 N	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	N bytes		N bytes	2 bytes
参数说明		包头	数据包头，每包数据均以 0xEF01 开始。					
		芯片地址	设备地址 0xFFFFFFFF，所有数据包必须包含该地址。					
		包标识	代表不同的数据包： 0x01：表示指令包。 0x02：表示数据包，还有后续的数据包。 0x08：表示数据包，没有后续包。 0x07：表示返回包。					
		包长度	包长度 = 包长度至校验和（指令、参数或数据）的总字节数，包含校验和，但不包含包长度本身的字节数。					
		指令	命令参数。					
		参数	不同指令参数不同					
		校验和	校验和是从包标识至校验和之间所有字节之和，包含包标识，但不包含校验和本身的字节数，超出 2 字节的进位忽略。					

1.2.2 数据包

包头	芯片地址	包标识	包长度	数据	校验和
2 bytes	4 bytes	1 byte	2 bytes	N bytes	2 bytes
参数说明	包头	数据包头，每包数据均以 0xEF01 开始。			
	芯片地址	设备地址 0xFFFFFFFF，所有数据包必须包含该地址。			
	包标识	代表不同的数据包： 0x01：表示指令包。 0x02：表示数据包，还有后续的数据包。 0x08：表示数据包，没有后续包。 0x07：表示返回包。			
	包长度	包长度 = 包长度至校验和（指令、参数或数据）的总字节数，包含校验和，但不包含包长度本身的 字节数。			
	数据	发送的数据本身			
	校验和	校验和是从包标识至校验和之间所有字节之和，包含包标识，但不包含校验和本身的字节数，超出 2 字节的进位忽略。			

1.2.3 应答包（返回包）

包头	芯片地址	包标识	包长度	确认码	返回参数	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 bytes	N bytes	2 bytes
参数说明	包头	数据包头，每包数据均以 0xEF01 开始。				
	芯片地址	设备地址 0xFFFFFFFF，所有数据包必须包含该地址。				
	包标识	代表不同的数据包： 0x01：表示指令包。 0x02：表示数据包，还有后续的数据包。 0x08：表示数据包，没有后续包。 0x07：表示返回包。				
	包长度	包长度 = 包长度至校验和（指令、参数或数据）的总字节数，包含校验和，但不包含包长度本身的 字节数。				
	确认码	反馈指令执行的情况				
	返回参数	反馈指令执行的参数				
	校验和	校验和是从包标识至校验和之间所有字节之和，包含包标识，但不包含校验和本身的字节数，超出 2 字节的进位忽略。				

- 注：
- 1、数据包不能单独进入执行流程，必须跟在指令包或应答包后面。
 - 2、下传或上传的数据包格式相同。
 - 3、对于多字节的高字节在前，低字节在后（如 2bytes 的 00 06 表示 0006，而不是 0600）。
 - 4、指令只能由上位机下给模块，模块向上位机应答。

2、软件开发指南

2.1 参数表（内部指纹模组）

- 参数表的内容是协议、算法运行的基本参数。整个软件系统都会用到参数表的内容，所以理解并妥善设置参数表对于如何正确使用芯片至关重要。
- 参数表由 DSP 初始化程序在初次上电时设置，并存于 FLASH 的系统参数存储区，以后每次上电 SOC 初始化程序都要首先将参数表装载到 RAM 中，并根据参数表内容初始化系统寄存器；参数表长度为 64 字（128 字节）。
- 参数表结构如下表所示：

参数表的初始内容由 ROM 驻留程序或用户程序在系统第一次上电时设置。

类型	序号	中文名称	英文名称	长度（字）	内容与默认值	注释
PART1	1	状态寄存器	SSR	1	0	
	2	传感器类型	SensorType	1		
	3	指纹库大小	DataBaseSize	1	根据 FLASH 类型自动判别	
PART2	4	安全等级	SecurLevel	1	3	分 5 个等级
	5	设备地址	DeviceAddress	2	0xffffffff	
	6	数据包大小	CFG_PktSize	1	1	此 8 个寄存器为系统配置表
	7	波特率系数	CFG_BaudRate	1	6	
	8		CFG_VID	1		
	9		CFG_PID	1		
	10	保留		1		
	11	保留		1		
	12	保留		1		
	13	保留		1		
	14	产品型号	ProductSN	4	ASCII 码	设备描述符
	15	软件版本号	SoftwareVersion	4	ASCII 码	
	16	厂家名称	Manufacturer	4	ASCII 码	
	17	传感器名称	SensorName	4	ASCII 码	
	18	密码	PassWord	2	00000000H	未启用
	19	Jtag 锁定标志	JtagLockFlag	2	00000000H	
	20	传感器初始化程序入口	SensorInitEntry	1	入口地址	
	21	录入图像程序入口	SensorGetImageEntry	1	入口地址	
	22	保留		27		
PART3	23	参数表有效标志	ParaTableFlag	1	0x1234	

- 参数表位于系统参数存储区第 1 页；
- 参数表在芯片上电时从 flash 装载到 RAM 中，结构与顺序不作任何改变；
- 参数表详解：

1) 状态寄存器

SSR

Reset Value: 0x0000

长度: 1 word

属性: 只读

用途: 系统状态指示

读取指令: PS_ReadSysPara, 详见指令说明

格式: 详见下表

状态寄存器格式

第 15~4 位	第 3 位	第 2 位	第 1 位	第 0 位
Reserved	ImgBufStat	PWD	Pass	Busy

注:

- Busy: 占 1 位，置“1”表示系统正在执行命令，“0”表示系统空闲；
- Pass: 占 1 位，置“1”表示指纹验证通过；
- PWD: 占一位，置“1”表示设备握手口令通过验证；
- ImgBufStat: 占一位，置“1”表示指纹图像缓冲区存在有效指纹图像。

2) 传感器类型	SensorType
Reset Value:	0x0000
长度:	1 word
属性:	只读
用途:	表示传感器驱动类型
读取指令:	PS_ReadSysPara, 详见指令说明
3) 指纹库大小	DataBaseSize
Reset Value:	According to FLASH
长度:	1 word
属性:	只读
用途:	指纹库容量指示
读取指令:	PS_ReadSysPara, 详见指令说明
4) 安全等级	SecurLevel
Reset Value:	0x0003
长度:	1 word
属性:	只读
用途:	安全等级指示; 系统根据该值设定比对阈值
读取指令:	PS_ReadSysPara 详见指令说明
设置指令:	PS_WriteReg 详见指令说明
五个等级:	
1: Level 1	Lowest
2: Level 2	
3: Level 3	
4: Level 4	
5: Level 5	Highest
5) 设备地址	DeviceAddress (STM32 已禁止修改)
Reset Value:	0xffffffff
长度:	2 word
属性:	只读
用途:	系统只接收地址相配的指令包/数据包
读取指令:	PS_ReadSysPara, 详见指令说明
设置指令:	PS_SetChipAddr, 详见指令说明
6) 数据包大小	CFG_PktSize
Reset Value:	0x0001
长度:	1 word
属性:	读/写
用途:	发送数据时, 系统根据该值设定单个数据包的长度

读取指令:	PS_ReadSysPara, 详见指令说明
设置指令:	PS_WriteReg, 详见指令说明
7) 波特率系数	CFG_BaudRate (STM32 已禁止修改)
Reset Value:	0x0006
长度:	1 word
属性:	只读
用途:	确定 uart 波特率=该值*9600
读取指令:	PS_ReadSysPara, 详见指令说明
设置指令:	PS_WriteReg, 详见指令说明
8) USB ID	CFG_VID
Reset Value:	0x0453
长度:	1 word
属性:	只读
用途:	USB 内嵌协议 VID
读取指令:	PS_ReadINPage, 详见指令说明
9) USB ID	CFG_PID
Reset Value:	0x9005
长度:	1 word
属性:	只读
用途:	USB 内嵌协议 PID
读取指令:	PS_ReadINPage, 详见指令说明
10) 产品型号	ProductSN
Reset Value:	第一次上电初始化值
长度:	4 words
属性:	只读
用途:	指示产品型号
读取指令:	PS_ReadINPage, 详见指令说明
11) 软件版本号	SoftwareVersion
Reset Value:	第一次上电初始化值
长度:	4 words
属性:	只读
用途:	指示软件版本号
读取指令:	PS_ReadINPage, 详见指令说明
12) 厂家名称	Manufacturer
Reset Value:	第一次上电初始化值
长度:	4 words

属性:	只读
用途:	指示厂家名称
读取指令:	PS_ReadINFpage, 详见指令说明
13) 传感器名称	SensorName
Reset Value:	第一次上电初始化值
长度:	4 words
属性:	只读
用途:	指示传感器名称
读取指令:	PS_ReadINFpage, 详见指令说明
14) 密码	PassWord
Reset Value:	0x00000000
长度:	2 words
属性:	读/写
用途:	握手口令, 口令通过系统才能响应
读取指令:	PS_ReadINFpage, 详见指令说明
设置指令:	PS_SetPwd, 详见指令说明
15) JTAG 锁止标志	JtagLockFlag
Reset Value:	0x00000000
长度:	2 words
属性:	只读
用途:	第一次上电时写入特定的值将关闭 JTAG 端口
读取指令:	PS_ReadINFpage, 详见指令说明
16) 传感器初始化入口	SensorInitEntry
Reset Value:	保留
长度:	1 word
属性:	只读
用途:	系统根据该值调用传感器初始化程序, 保留
读取指令:	PS_ReadINFpage, 详见指令说明
17) 传感器图像获取入口	SensorGetImageEntry
Reset Value:	保留
长度:	1 word
属性:	只读
用途:	系统根据该值调用传感器采集图像程序, 保留
读取指令:	PS_ReadINFpage, 详见指令说明
18) 参数表有效标志	ParaTableFlag
Reset Value:	0x1234
长度:	1 word

属性:	只读
用途:	若该域的值是 0x1234，则表示参数表已经初始化；若该域的值是 0x0204，则表示系统只对参数表的 PART1 部分进行初始化；该域若为其他值，系统将初始化参数表。
读取指令:	PS_ReadINPage，详见指令说明

2.2 ROM 及传感器驱动

ROM 内嵌了完整的指纹识别算法。

2.3 设备地址

芯片的默认地址为 0xFFFFFFFF，不允许修改，数据包的地址域必须与该地址相配，命令包/数据包才被系统接收。

2.4 Unit-Fingerprint 工作状态

2.4.1 定时休眠模式

(1) 唤醒

- 1、休眠状态，通过手指按压唤醒。
- 2、通过指令唤醒（具体可查看 3.1.32 指令）

(2) 休眠

1. 定时器功能与配置

STM32 内部通过定时器进行周期性计时，定时时间可通过指令 3.1.28 进行配置。当串口接收到数据时，定时器将被复位并重新开始计时。

2. 到时的自动查询机制

若定时时间到达且期间未接收到串口数据，STM32 会主动向指纹模组查询手指按压状态，最多进行三次：

如果三次均返回“无手指按压”，则关闭指纹模组；

如果任一查询返回“检测到手指按压”，则立即终止查询，并复位定时器重新计时。

3. 指令处理限制

在 STM32 进行手指状态查询期间，仅允许处理指令码在 0xD0 ~ 0xD7 范围内的指令。其余指令将被拒绝，并返回错误码 0xFB（详见“返回类型表”）。

4. 异常指令的处理与响应

在 STM32 进行手指状态自动查询期间，若接收到不在允许范围（0xD0 ~ 0xD7）内的指令，STM32 会先返回错误码 0xFB（详见“返回类型表”），然后退出当前查询流程，并复位定时器重新开始计时。

2.4.2 开启模式

该模式下，设备一直是开启状态。

注：

定时休眠模式下，激活成功后发送返回包，格式对应 1.2.3 应答包，确认码为 0xFF（返回数据包则为 0xEF 0x01 0xFF 0xFF 0xFF 0xFF 0x07 0x00 0x03 0xFF 0x01 0x09）

2.5 Unit-Fingerprint 内部指纹模组关闭状态

在该状态下，指令码为 0xD0~0xD7 的指令仍可正常处理。若需让内部指纹模组进入工作状态，必须先通过被动或主动方式激活其内部指纹模组。

3、指令集定义

指纹模块 SOC 挂接必要的外围电路后即可构成完整的指纹识别模块，模块始终处于从属地位（Slave mode），主控（Host）需要通过不同的指令让模块完成各种功能。主控的指令、模块的应答以及数据交换都是按照规定格式的数据包来进行的。主控必须按照下述格式封装要发送的指令或数据，也必须按下述格式解析收到的数据包。

注：若指令返回的确认码为 0xFE 则表示内部指纹模块未激活，需要先激活指纹模组，激活的具体步骤可参考 2.4。

3.1 通用指令集

序号	命令码	名称	描述
1	0x01	PS_GetImage	验证用获取图像
2	0x29	PS_GetEnrollImage	注册用获取图像
3	0x02	PS_GenChar	根据原始图像生成指纹特征文件存于模板缓冲区
4	0x03	PS_Match	精确比对模板缓冲区中的特征文件或者模板
5	0x04	PS_Search	以模板缓冲区中的特征文件搜索整个或部分指纹库
6	0x05	PS_RegModel	将特征文件融合后生成一个模板
7	0x06	PS_StoreChar	将模板缓冲区中的模板文件储存在 flash 指纹库中
8	0x07	PS_LoadChar	从 flash 指纹库中读取一个模板到模板缓冲区
9	0x0A	PS_UpImage	上传原始图像
10	0x0C	PS_DeletChar	删除 flash 指纹库中的一个模板文件
11	0x0D	PS_Empty	清空 flash 指纹库
12	0x0E	PS_WriteReg	写 SOC 系统寄存器
13	0x0F	PS_ReadSysPara	读系统基本参数
14	0x14	PS_GetRandomCode	采样随机数
15	0x16	PS_ReadINFpage	读取 FLASH Information Page 内容
16	0x18	PS_WriteNotepad	写记事本
17	0x19	PS_ReadNotepad	读记事本
18	0x1D	PS_ValidTemplateNum	读有效模板个数
19	0x1F	PS_ReadIndexTable	读索引表
20	0x34	PS_GetChipSN	获取芯片唯一序列号
21	0x35	PS_HandShake	握手指令
22	0x36	PS_CheckSensor	校验传感器
23	0x3C	PS_ControlBLN	呼吸灯指令
24	0x3D	PS_GetImageInfo	获取图像信息
25	0x3E	PS_SearchNow	搜索当前指纹
26	0x7A	PS_UpTemplet	特殊上传模板
27	0x7B	PS_DownTemplet	特殊下载模板
28	0xD0	PS_SetSleepTime	设置休眠时间
29	0xD1	PS_GetSleepTime	获取休眠时间
30	0xD2	PS_SetWorkMode	设置工作模式
31	0xD3	PS_GetWorkMode	获取工作模式
32	0xD4	PS_ActivateModule	激活指纹模组

33	0xD5	PS_GetFigurationModuleStatus	获取指纹模组工作状态
34	0xD6	PS_SaveConfigurationToFlash	保存配置信息到 flash
35	0xD7	PS_GetFirmwareVersion	获取 STM32 固件版本

3.1.1 验证用获取图像 PS_GetImage

- 功能说明： 验证指纹时，探测手指，探测到后录入指纹图像存于图像缓冲区。返回确认码表示：录入成功、无手指等。
- 输入参数： none
- 返回参数： 确认字
- 指令代码： 01H
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	01H	0005H

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注：确认码=00H 表示获取图像成功；

确认码=01H 表示收包有错；

确认码=02H 表示传感器上无手指；

确认码=03H 表示获取图像不成功；

sum 指校验和。

3.1.2 注册用获取图像 PS_GetEnrollImage

- 功能说明： 注册指纹时，探测手指，探测到后录入指纹图像存于图像缓冲区。返回确认码表示：录入成功、无手指等。
- 输入参数： none
- 返回参数： 确认字
- 指令代码： 29H
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	29H	002DH

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注：确认码=00H 表示获取图像成功；
 确认码=01H 表示收包有错；
 确认码=02H 表示传感器上无手指；
 确认码=03H 表示获取图像不成功；
 sum 指校验和。

3.1.3 生成特征 PS_GenChar

- 功能说明： 将图像缓冲区中的原始图像生成指纹特征文件存于模板缓冲区。
- 输入参数： BufferID（正整数，即 1、2……）
- 返回参数： 确认字
- 指令代码： 02H
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	缓冲区号	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes
0xEF01	xxxx	01H	0004H	02H	BufferID	sum

注：在注册过程中，BufferID 表示按第几次手指；其他情况中，BufferID 有相应的默认值。

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注：确认码=00H 表示生成特征成功；
 确认码=01H 表示收包有错；
 确认码=06H 表示图像太乱而生不成特征；
 确认码=07H 表示指纹图像正常，但特征点太少而生不成特征；
 确认码=08H 表示在注册过程中要求每次录入相似的手指区域，当次特征与前次特征不相似；(默认关闭此功能)
 确认码=0aH 表示特征合并失败；
 确认码=15H 表示图像缓冲区内没有有效原始图而生不成图像；
 确认码=28H 表示在注册过程中要求每次录入不同的手指区域，当次特征与前次特征重合面积过多；(默认关闭此功能)
 sum 指校验和。

3.1.4 精确比对两枚指纹特征 PS_Match

- 功能说明： 精确比对模板缓冲区中的特征文件。
- 输入参数： none
- 返回参数： 确认字，比对得分
- 指令代码： 03H

- 指令包格式:

包头	设备地址	包标识	包长度	指令码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	03H	0007H

- 应答包格式:

包头	设备地址	包标识	包长度	确认码	得分	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes	2 bytes
0xEF01	xxxx	07H	0005H	xxH	xxxxH	sum

注: 确认码=00H 表示指纹匹配;
 确认码=01H 表示收包有错;
 确认码=08H 表示指纹不匹配;
 sum 指校验和

3.1.5 搜索指纹 PS_Search

- 功能说明: 以模板缓冲区中的特征文件搜索整个或部分指纹库。若搜索到, 则返回页码。加密等级设置为 0 或 1 情况下支持此功能。
- 输入参数: BufferID (默认为 1), StartPage (起始页), PageNum (页数)
- 返回参数: 确认字, 页码 (相配指纹模板), 得分 (MatchScore)
- 指令代码: 04H
- 指令包格式:

包头	设备地址	包标识	包长度	指令码	缓冲区号	参数	参数	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes	2 bytes	2 bytes
0xEF01	xxxx	01H	0008H	04H	BufferID	StartPage	PageNum	sum

注: BufferID 默认为 1, 以模板缓冲区中指纹模板搜索整个或部分指纹库。

- 应答包格式:

包头	设备地址	包标识	包长度	确认码	页码	得分	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes	2 bytes	2 bytes
0xEF01	xxxx	07H	07H	xxH	PageID	MatchScore	sum

注: 确认码=00H 表示搜索到;
 确认码=01H 表示收包有错;
 确认码=09H 表示没搜索到; 此时页码与得分为 0;
 确认码=17H 表示残留指纹或两次采集之间手指没有移动;
 sum 指校验和。

3.1.6 合并特征 (生成模板) PS_RegModel

- 功能说明: 将特征文件融合后生成一个模板。
- 输入参数: none
- 返回参数: 确认字

- 指令代码： 05H

- 指令包格式：

包头	设备地址	包标识	包长度	指令码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	05H	0009H

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注：确认码=00H 表示合并成功；
 确认码=01H 表示收包有错；
 确认码=0aH 表示合并失败；
 确认码=3bH 表示模板质量差，合并失败；
 sum 指校验和。

3.1.7 储存模板 PS_StoreChar

- 功能说明： 将模板缓冲区中的模板文件存到 PageID 号 flash 数据库位置。加密等级设置为 0 或 1 情况下支持此功能。
- 输入参数： BufferID（默认为 1），PageID（指纹库位置号）
- 返回参数： 确认字
- 指令代码： 06H
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	缓冲区号	位置号	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes	2 bytes
0xEF01	xxxx	01H	0006H	06H	BufferID	PageID	sum

注：BufferID 默认为 1。

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注：确认码=00H 表示储存成功；
 确认码=01H 表示收包有错；
 确认码=0bH 表示 PageID 超出指纹库范围；
 确认码=18H 表示写 FLASH 出错；
 确认码=35H 表示非法数据；
 sum 指校验和。

3.1.8 读出模板 PS_LoadChar

- 功能说明： 将flash 数据库中指定 ID 号的指纹模板读入到模板缓冲区中。

- 输入参数： BufferID（默认为2）， PageID（指纹库模板号）
- 返回参数： 确认字
- 指令代码： 07H
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	缓冲区号	页码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes	2 bytes
0xEF01	xxxx	01H	0006H	07H	BufferID	PageID	sum

注：BufferID 默认为 2。

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注：确认码=00H 表示读出成功；

确认码=01H 表示收包有错；

确认码=0bH 表示 PageID 超出指纹库范围；

确认码=01H 表示读出有错或模板无效；

sum 指校验和。

3.1.9 上传图像 PS_UplImage

- 功能说明： 将图像缓冲区中的数据上传给主控。
- 输入参数： none
- 返回参数： 确认字
- 指令代码： 0AH
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	0AH	000EH

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注：确认码=00H 表示接着发送后续数据包；

确认码=01H 表示收包有错；

确认码=0fH 表示不能发送后续数据包；

sum 指校验和。

- 应答之后发送后续数据包。

包头	设备地址	包标识	包长度	数据	校验和
2 bytes	4 bytes	1 byte	2 bytes	N bytes	2 bytes
0xEF01	xxxx	xxH	xxxxH	xx	sum

注：包标识=02：数据包，且有后续包。

包标识=08：最后一个数据包，即结束包。

UART 上传图像数据包时，按照预先设置的长度分包发送。

- 一个字节含两个像素，每个像素占4bits。

3.1.10 删除模板 PS_DeletChar

- 功能说明： 删除flash 数据库中指定 ID 号开始的 N 个指纹模板。
- 输入参数： 指纹库模板号（PageID），删除的模板个数（N）。
- 返回参数： 确认字
- 指令代码： 0CH
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	页码	删除个数	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes	2 bytes	2 bytes
0xEF01	xxxx	01H	0007H	0CH	PageID	N	sum

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注：确认码=00H 表示删除模板成功；

确认码=01H 表示收包有错；

确认码= 10H 表示删除模板失败；

sum 指校验和。

3.1.11 清空指纹库 PS_Empty

- 功能说明： 删除flash 数据库中所有指纹模板。
- 输入参数： none
- 返回参数： 确认字
- 指令代码： 0DH
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	0DH	0011H

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注：确认码=00H 表示清空成功；

确认码=01H 表示收包有错；
 确认码=11H 表示清空失败；
 sum 指校验和。

3.1.12 写系统寄存器 PS_WriteReg

- 功能说明： 写模块寄存器。
- 输入参数： 寄存器序号、内容
- 返回参数： 确认字
- 指令代码： 0EH
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	寄存器序号	内容	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	1 byte	2 bytes
0xEF01	xxxx	01H	0005H	0EH	6	xx	sum

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注 1：确认码=00H 表示 OK；

确认码=01H 表示收包有错；

确认码=18H 表示读写 FLASH 出错；

确认码=1aH 表示寄存器序号有误；

确认码=1bH 表示寄存器设定内容错误号；

sum 指校验和。

注 2：写系统寄存器（PS_WriteReg）指令执行时，先按照原配置进行应答，应答之后修改系统设置，并将配置记录于 FLASH，同时按照新的配置工作。

寄存器号	寄存器名称	内容说明
6	包大小寄存器	0 : 32bytes 1 : 64bytes 2 : 128bytes 3 : 256bytes

3.1.13 读系统基本参数 PS_ReadSysPara

- 功能说明： 读取模块的基本参数（波特率，包大小等）。参数表前 16 个字节存放了模块的基本通讯和配置信息，称为模块的基本参数。
- 输入参数： none
- 返回参数： 确认字，基本参数（16bytes）
- 指令代码： 0FH
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	0FH	0013H

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	基本参数列表	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	16 bytes	2 bytes
0xEF01	xxxx	07H	0013H	xxH	结构见表	sum

注：确认码=00H 表示 OK；

确认码=01H 表示收包有错；

sum 指校验和。

名称	内容说明	偏移量（字节）	大小（字节）
状态寄存器	系统的状态寄存器内容	0	2
传感器类型	传感器类型代码。	2	2
指纹库大小	指纹库容量	4	2
安全等级	分数等级代码（1/2/3/4/5）	6	2
设备地址	32 位设备地址	8	4
数据包大小	数据包大小代码： 0：32bytes 1：62bytes 2：128bytes 3：256bytes	12	2
波特率设置	N（波特率为 9600*N bps）	14	2

3.1.14 采样随机数 PS_GetRandomCode

- 功能说明： 令芯片生成一个随机数并返回给主控。

- 输入参数： none

- 返回参数： 确认字，随机数

- 指令代码： 14H

- 指令包格式：

包头	设备地址	包标识	包长度	指令码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	14H	0018H

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	随机数	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	4 bytes	2 bytes
0xEF01	xxxx	07H	0007H	xxH	xxxx	sum

注：确认码=00H 表示生成成功；

确认码=01H 表示收包有错；

确认码=19H 表示随机数生成失败；

sum 指校验和。

3.1.15 读 flash 信息页 PS_ReadINFpage

- 功能说明： 读取 FLASH Information Page 所在的信息页（512bytes）。
- 输入参数： none
- 返回参数： 确认字
- 指令代码： 16H
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	16H	001AH

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注：确认码=00H 表示随后发数据包；

确认码=01H 表示收包有错；

sum 指校验和。

- 应答之后发送后续数据包。

包头	芯片地址	包标识	包长度	数据	校验和
2 bytes	4 bytes	1 byte	2 bytes	N bytes	2 bytes
0xEF01	xxxx	xxH	xxxxH	xx	sum

注：包标识=02：数据包，且有后续包。

包标识=08：最后一个数据包，即结束包。

UART 读 flash 信息页数据包时，按照预先设置的长度分包发送。

3.1.16 写记事本 PS_WriteNotepad

- 功能说明： 模块内部为用户开辟了256bytes的FLASH 空间用于存放用户数据，该存储空间称为用户记事本，该记事本逻辑上被分成8个页，写记事本命令用于写入用户的32bytes 数据到指定的记事本页。
- 输入参数： 页码（0~7），用户信息（User content）
- 返回参数： 确认字
- 指令代码： 18H
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	页码	用户信息	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	32 bytes	2 bytes
0xEF01	xxxx	01H	0024H	18H	0~7	User content	sum

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注：确认码=00H 表示 OK；

确认码=01H 表示收包有错；

确认码= 1cH 表示记事本页码指定错误；

确认码= 18H 表示读写 FLASH 出错

sum 指校验和。

3.1.17 读记事本 PS_ReadNotepad

- 功能说明： 读取 FLASH 用户区的 256byte 数据。

- 输入参数： 页码（0~7）

- 返回参数： 确认字，用户信息（User content）

- 指令代码： 19H

- 指令包格式：

包头	设备地址	包标识	包长度	指令码	页码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes
0xEF01	xxxx	01H	0004H	19H	0~7	sum

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	用户信息	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	32 bytes	2 bytes
0xEF01	xxxx	07H	0023H	xxH	User content	sum

注：确认码=00H 表示 OK；

确认码=01H 表示收包有错；

确认码= 1cH 表示记事本页码指定错误；

sum 指校验和。

3.1.18 读有效模板个数 PS_ValidTemplateNum

- 功能说明： 读有效模板个数。

- 输入参数： none

- 返回参数： 确认字，有效模板个数（ValidN）

- 指令代码： 1DH

- 指令包格式：

包头	设备地址	包标识	包长度	指令码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	1DH	0021H

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	有效模板个数	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes	2 bytes
0xEF01	xxxx	07H	0005H	xxH	ValidN	sum

注：确认码=00H 表示读取成功；

确认码=01H 表示收包有错；

sum 指校验和。

3.1.19 读索引表 PS_ReadIndexTable

- 功能说明： 读取录入模版的索引表。
- 输入参数： 索引表页码，页码0，1 分别对应模版从0-256，256-512 的索引，每1 位代表一个模 版，1 表示对应存储区域的模版已经录入，0 表示没录入。

- 返回参数： 确认字，索引表信息（Index）

- 指令代码： 1FH

- 指令包格式：

包头	设备地址	包标识	包长度	指令码	页码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes
0xEF01	xxxx	01H	0004H	1FH	0~1	sum

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	索引信息	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	32 bytes	2 bytes
0xEF01	xxxx	07H	0023H	xxH	Index	sum

注：确认码=00H 表示 OK；

确认码=01H 表示收包有错；

确认码=0bH 表示访问指纹库时地址序号超出指纹库范围；

sum 指校验和。

3.1.20 获取芯片唯一序列号 PS_GetChipSN

- 功能说明： 获取芯片唯一序列号。
- 输入参数： 预留
- 返回参数： 确认字，唯一序列号（SN）

- 指令代码： 34H

- 指令包格式：

包头	设备地址	包标识	包长度	指令码	参数	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes
0xEF01	xxxx	01H	0004H	34H	0	0039H

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	唯一序列号	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	32 bytes	2 bytes

0xEF01	xxxx	07H	0023H	xxH	SN	sum
--------	------	-----	-------	-----	----	-----

注：确认码=00H 表示 OK；

确认码=01H 表示收包有错；

sum 指校验和。

3.1.21 握手指令 PS_HandShake

- 功能说明： 检查模组是否正常工作。
- 输入参数： none
- 返回参数： 确认字
- 指令代码： 35H
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	35H	0039H

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注：确认码=00H 表示 OK；

确认码=01H 表示收包有错；

sum 指校验和。

3.1.22 校验传感器 PS_CheckSensor

- 功能说明： 校验传感器是否正常工作。
- 输入参数： none
- 返回参数： 确认字
- 指令代码： 36H
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	36H	003AH

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注：确认码=00H 表示 OK；

确认码=01H 表示收包有错；

确认码=29H 表示校验传感器出错；

sum 指校验和。

3.1.23 LED 控制灯指令 PS_ControlBLN

- 功能说明： 控制灯指令主要分为两类：一般指示灯和七彩编程呼吸灯。
- 输入参数： 功能码，起始颜色，结束颜色，循环次数
- 返回参数： 确认字
- 指令代码： 3CH
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	功能码	起始颜色	结束颜色	循环次数	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	1 byte	1 byte	1 byte	2 bytes
0xEF01	xxxx	01H	0007H	3CH	xxH	xxH	xxH	xxH	sum

- 辅助说明：

功能码：LED 灯模式控制位，1-普通呼吸灯，2-闪烁灯，3-常开灯，4-常闭灯，5-渐开灯，6 渐闭灯， 其他功能码不适用于此指令包格式：

起始颜色：设置为普通呼吸灯时，由灭到亮的颜色，只限于普通呼吸灯（功能码 01）功能，其他功能时，与结束颜色保持一致，其中，bit0 是蓝灯控制位，bit1 是绿灯控制位，bit2 是红灯控制位，置 1 灯亮，置 0 灯灭。例如 0x01 蓝灯亮，0x02 绿灯亮，0x03 青色灯亮，0x04 红灯亮，0x05 紫色灯亮，0x06 黄灯亮，0x07 白色灯亮，0x00 全灭。

结束颜色：设置为普通呼吸灯时，由亮到灭的颜色，只限于普通呼吸灯（功能码 0x01），其他功能时，与起始颜色保持一致，设置方式与起始颜色一样。

循环次数：表示呼吸或者闪烁灯的次数，当设为 0 时，表示无限循环，当设为其他值时，表示呼吸有限次数，循环次数适用于呼吸，闪烁功能，其他功能中无效，例如在常开，常闭，渐开和渐闭中是无效的：

包头	芯片地址	包标识	包长度	指令码	功能码	时间位	颜色 1		颜色5	循环次数	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	1 byte	1 byte		1 byte	1 byte	2 bytes
0xEF01	xxxx	01H	000BH	3CH	xxH	xxH	xxH		xxH	xxH	sum

- 辅助说明：

功能码：7-七彩编程呼吸灯，其他功能码不适用于此指令包格式：

时间位：用来控制灯呼吸一次的时间，即从灭到亮，再到灭的时间，单次呼吸的时间范围是 0.1s~10.0s 左右，用 1-100 之间的数字表示，超出此范围的数字无效。即时间位设为 1 对应 0.1s。时间位设为 100 对应 10.0s。推荐时间位设置为 36，呼吸时间与普通呼吸灯（功能码 0x01）相同，大概是 3.6s 左右。

颜色码：共 5 个字节组成，如下表所示，每个字节的颜色码分成 2 个单元，每个单元有 4 位，从高位开始分为一个有效位，和 3 个颜色控制位，每个单元控制某种颜色的灯从灭到亮，再到灭的过程。另外，编程后的呼吸灯 1 次循环点亮顺序是从颜色 1 的单元 1 开始，然后是颜色 1 的单元 2，接着是颜色 2 的单元 1，以此类推。

颜色（1 byte）							
单元 1				单元 2			
有效位	红灯位	绿灯位	蓝灯位	有效位	红灯位	绿灯位	蓝灯位

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
------	------	------	------	------	------	------	------

有效位：0-此单元及此单元后的所有单元均无效，1-此单元有效；

红灯位：0-红灯灭，1-红灯亮；

绿灯位：0-绿灯灭，1-绿灯亮；

蓝灯位：0-蓝灯灭，1-蓝灯亮；

循环次数：表示呼吸灯的次数，当设为 0 时，表示无限循环，当设为其他值时，表示呼吸有限次数。

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注：确认码=00H 表示生成地址成功；

确认码=01H 表示收包有错；

确认码=35H 表示非法数据；

sum 指校验和。

3.1.24 获取图像信息指令 PS_GetImageInfo

- 功能说明：探测到后录入指纹图像存于图像缓冲区，并返回图像信息。
- 输入参数：none
- 返回参数：确认字、图像面积（百分比）、图像质量（0：合格；其他：不合格）
- 指令代码：3DH
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	3DH	0041H

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	图像面积	图像质量	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	1 byte	2 bytes
0xEF01	xxxx	07H	0005H	xxH	xxH	xxH	sum

注：确认码=00H 表示获取图像成功；

确认码=01H 表示收包有错；

确认码=02H 表示传感器上无手指；

确认码=03H 表示录入指纹图像失败；

sum 指校验和。

3.1.25 搜索当前指纹指令 PS_SearchNow

- 功能说明：以模板缓冲区中最近一次提取的特征文件搜索整个或部分指纹库。若搜索到，则返回页码。
如表 3-1 中加密等级设置为 0 和 1 情况支持此功能。
- 输入参数：起始页（StartPage），页数（PageNum）

- 返回参数： 确认字，页码（相配指纹模板），得分（MatchScore）
- 指令代码： 3EH
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	起始页	页数	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes	2 bytes	2 bytes
0xEF01	xxxx	01H	0007H	3EH	StartPage	PageNum	sum

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	图像面积	图像质量	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes	2 bytes	2 bytes
0xEF01	xxxx	07H	0007H	xxH	PageID	MatchScore	sum

注：确认码=00H 表示搜索到；

确认码=01H 表示收包有错；

确认码=09H 表示没搜索到；此时页码与得分为0； 确

认码=31H 表示功能与加密等级不匹配；

sum 指校验和。

3.1.26 特殊上传模板 PS_UpTemplet

- 功能说明： 上传模板。
- 输入参数： 模板偏移地址，上传模板大小
- 返回参数： 确认字，上传模板大小，模板数据
- 指令代码： 7AH
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	模板偏移地址	上传模板大小	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes	2 bytes	2 bytes
0xEF01	xxxx	01H	0007H	7AH	xxxx	xxxx	sum

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	上传模板大小	模板数据	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes	N bytes	2 bytes
0xEF01	xxxx	07H	xxxx	xxH	xxxx	xxxx	sum

注 1：确认码=00H 表示 OK；

确认码=01H 表示收包有错；

sum 指校验和。

注 2：指定上传模板数据大于实际剩余模板数据时，按实际剩余模板数据上传；

指定模板偏移地址大于模板长度时，返回上传模板大小为 0。

3.1.27 特殊下载模板 PS_DownTemplet

- 功能说明： 下载模板。
- 输入参数： 模板偏移地址，下载模板大小，模板数据
- 返回参数： 确认字
- 指令代码： 7BH
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	模板偏移地址	下载模板大小	模板数据	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes	2 bytes	N bytes	2 bytes
0xEF01	xxxx	01H	0007H	7BH	xxxx	xxxx	xxxx	sum

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注 1：确认码=00H 表示 OK；

确认码=01H 表示收包有错；

sum 指校验和。

3.1.28 设置休眠时间 PS_SetSleepTime

- 功能说明： 设置指纹模组的休眠时间（系统默认是 10 秒）
- 输入参数： N（时间刻度是秒，范围 10~254）
- 返回参数： 确认字
- 指令代码： D0H
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	休眠时间	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 bytes	2 bytes
0xEF01	xxxx	01H	0004H	D0H	xxxx	sum

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxxx	sum

注 1：确认码=00H 表示 OK；

确认码=01H 表示收包有错；

确认码=FDH 表示参数错误；

sum 指校验和。

注 2：该时间参数仅在“定时休眠模式”下有效，用于判断指纹模组何时进入休眠状态。如果在设定的时间周期内未向指纹模组发送任何指令，模组将主动检测是否有指纹按压。检测过程共进行三次，若三次检测结果均为无指纹按压，指纹模组将自动关闭，并进入待机状态，等待指纹按压或外部指令唤醒。

如果在三次检测过程中发现有指纹按压，计时器将复位，重新开始定时周期。此外，在定时期内，若向指纹模组发送控制指令或修改定时时间参数，也会导致定时器复位并重新开始计时。

注 3：可通过 PS_SaveConfigurationToFlash 保存到内部 flash，掉电不丢失。

3.1.29 获取休眠时间 PS_GeSleepTime

- 功能说明： 获取指纹模组的休眠时间
- 输入参数： 无
- 返回参数： 确认字、休眠时间（时间刻度是秒）
- 指令代码： D1H
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	D1H	00D5H

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	休眠时间	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 bytes	2 bytes
0xEF01	xxxx	07H	0004H	xxH	xxH	sum

注：确认码=00H 表示 OK；

确认码=01H 表示收包有错；

sum 指校验和。

3.1.30 设置工作模式 PS_SetWorkMode

- 功能说明： 设置工作模式
- 输入参数： 工作模式（0:定时休眠模式，1:开启模式）
- 返回参数： 确认字
- 指令代码： D2H
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	工作模式	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 bytes	2 bytes
0xEF01	xxxx	01H	0004H	D2H	xxH	sum

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注 1：确认码=00H 表示 OK；

确认码=01H 表示收包有错；

确认码=FDH 表示参数错误；

sum 指校验和。

注 2：“定时休眠模式”是指指纹模组根据其当前工作状态，定时判断是否应进入休眠。在该模式下，系统会定期查询模组的状态，以决定是否关闭指纹识别功能。（何时进入休眠可查看设置休眠时间指令介绍）而“开启模式”则表示指纹模组始终处于工作状态，不会自动进入休眠。

注 3：可通过 PS_SaveConfigurationToFlash 保存到内部 flash，掉电不丢失。

3.1.31 获取工作模式 PS_GetWorkMode

- 功能说明： 获得工作模式
- 输入参数： 无
- 返回参数： 工作模式
- 指令代码： D3H
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	D3H	00D7H

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	工作模式	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 bytes	2 bytes
0xEF01	xxxx	07H	0004H	xxH	xxxx	sum

注：确认码=00H 表示 OK；

确认码=01H 表示收包有错；

sum 指校验和。

3.1.32 激活 Unit-Fingerprint 内部指纹模组 PS_ActivateFingerprintModule

- 功能说明： 主动激活 Unit-Fingerprint 内部指纹模组（如果指纹模组未开启将开启指纹模组）
- 输入参数： 无
- 返回参数： 确认字
- 指令代码： D4H
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	D4H	00D8H

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxxx	sum

注：确认码=00H 表示 OK；

确认码=01H 表示收包有错；

sum 指校验和

3.1.33 获取指纹模组工作状态 PS_GetFingerprintModuleStatus

- 功能说明： 获取指纹模组工作状态
- 输入参数： 无
- 返回参数： 确认字、指纹模组的工作状态（0 表示未开启，1 表示以开启）
- 指令代码： D5H
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	D5H	00D9H

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	工作状态	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 bytes	2 bytes
0xEF01	xxxx	07H	0004H	xxH	xxxx	sum

注：确认码=00H 表示 OK；

确认码=01H 表示收包有错；

确认码=FDH 表示参数错误；

sum 指校验和。

3.1.34 保存配置信息到 flash PS_SaveConfigurationToFlash

- 功能说明： 保存休眠时间、工作模式到内部 flash，掉电不丢失，**该指令会影响设备使用寿命请勿频繁使用**。
- 输入参数： 保存配置（0 保存休眠时间、1 保存工作模式）
- 返回参数： 确认字、工作状态（0 表示保存成功，1 表示保存失败）
- 指令代码： D6H
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	保存配置	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes
0xEF01	xxxx	01H	0004H	D6H	xxxx	xxxx

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注 1：确认码=00H 表示 OK；

确认码=01H 表示 Error；

sum 指校验和。

注 2：在对 ST 芯片的 Flash 进行读写操作时，为了延长其使用寿命，请避免频繁写入。写入 Flash 之前需

要先擦除页面，该过程相对耗时，大约需要 30ms。如果内存中的值与即将保存的值相同，则无需执行擦除和写入操作，从而减少不必要的 Flash 损耗。

3.1.35 获取 STM32 固件版本 PS_GetFirmwareVersion

- 功能说明： 获得 STM32 的固件版本
- 输入参数： 无
- 返回参数： 确认字、固件版本号
- 指令代码： D7H
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	D7H	00DBH

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	固件版本	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 bytes	2 bytes
0xEF01	xxxx	07H	0004H	xxH	xxxx	sum

注：确认码=00H 表示 OK；
确认码=01H 表示收包有错；
sum 指校验和。

3.2 模块化指令集

序号	命令码	名称	描述
1	0x30	PS_Cancel	取消指令
2	0x31	PS_AutoEnroll	自动注册模块指令
3	0x32	PS_AutoIdentify	自动验证指纹指令

对于自动注册(0x31)和自动验证(0x32)，若在等待期间无手指按压，且模组仍未返回超时的前提下，不可直接切断电源 VDD，请先发送“取消指令”PS cancel(0x30)。

3.2.1 取消指令 PS_Cancel

- 功能说明： 取消自动注册模板和自动验证指纹。加密等级设置为0 或 1 情况下支持此功能。
- 输入参数： none
- 返回参数： 确认字
- 指令代码： 30H
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	30H	xxxxH

- 应答包格式：

包头	设备地址	包标识	包长度	确认码	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注：确认码=00H 表示取消设置成功；

确认码=01H 表示取消设置失败；

确认码=31H 表示功能与加密等级不匹配；

sum 指校验和。

3.2.2 自动注册模板 PS_AutoEnroll

- 功能说明： 一站式注册指纹，包含采集指纹、生成特征、组合模板、存储模板等功能。如加密等级设置为 0 或 1 情况下支持此功能。
- 输入参数： ID 号、录入次数、参数
- 返回参数： 确认字、参数
- 指令代码： 31H
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	ID 号	录入次数	参数	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes	1 byte	2 bytes	2 bytes
0xEF01	xxxx	01H	0008H	31H	xxxxH	xxH	xxH	sum

- 辅助说明：

ID 号：高字节在前，低字节在后。例如录入 1 号指纹，则是 0001H。

录入次数：1byte，录入 2 次，则为 02H，录入 4 次则为 04H。参数：

最低位为 bit0。

- 1) bit0：预留；
- 2) bit1：预留；
- 3) bit2：注册过程中，是否要求模块在关键步骤，返回当前状态，0-要求返回，1-不要求返回；
- 4) bit3：是否允许覆盖 ID 号，0-不允许，1-允许；
- 5) bit4：允许指纹重复注册控制位，0-允许，1-不允许；
- 6) bit5：注册时，多次指纹采集过程中，是否要求手指离开才能进入下一次指纹图像采集，0-要求离开；1-不要求离开；
- 7) Bit6~bit15：预留。

- 应答包格式：

包头	芯片地址	包标识	包长度	确认码	参数 2 bytes		校验和	备注
					参数 1	参数 2		
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	1 byte	2 bytes	

0xEF01	xxxx	07H	0005H	xxH	00H	00H	sum	指令合法性检测： 合法/ ..
0xEF01	xxxx	07H	0005H	xxH	01H	1	sum	采图结果： 成功/超时
0xEF01	xxxx	07H	0005H	xxH	02H	1	sum	生成特征结果： 成功/失败
0xEF01	xxxx	07H	0005H	xxH	03H	1	sum	手指离开， 第 1 次录入成功： 成功/超时
						...		
0xEF01	xxxx	07H	0005H	xxH	01H	n	sum	采图结果： 成功/超时
0xEF01	xxxx	07H	0005H	xxH	02H	n	sum	生成特征结果： 成功/失败
0xEF01	xxxx	07H	0005H	xxH	04H	F0H	sum	合并模板
0xEF01	xxxx	07H	0005H	xxH	05H	F1H	sum	已注册检测
0xEF01	xxxx	07H	0005H	xxH	06H	F2H	sum	模板存储结果

● 确认码、参数 1 和参数 2 的返回值

确认码	释义	参数 1	释义	参数 2	释义
00H	成功	00H	指纹合法性检测	00H	指纹合法性检测
01H	失败	01H	获取图像	F0H	合并模板
07H	生成特征失败	02H	生产特征	F1H	检验该手指是否已注册
08H	注册过程中要求每次录入相似的手指区域，当次特征与前次特征不相似（默认关闭此功能）	03H	判断手指离开	F2H	存储模板
0aH	合并失败	04H	合并模板	n	当前录入第 n 次数
0bH	ID 号超出范围	05H	注册检验		
18H	读写 FLASH 出错	06H	存储模板		
1fH	指纹库已满				
22H	指纹模板非空				
25H	录入次数设置错误				
26H	超时				
27H	指纹已注册				
28H	注册过程中要求每次录入不同的手指区域，当次特征与前次特征重合面积过多（默认关闭此功能）				
31H	功能与加密等级不匹配				
35H	非法数据				

确认码	释义	参数 1	释义	参数2	释义
3bH	模板质量差				

- 指令说明：
 - 1) 若指定 ID 号无效，则确认码、参数 1 和参数 2 返回（以下直接描述为返回）：0b 00 00H。
合法性检测：
 - 若指定 ID 号无效，则返回：0b 00 00H。
 - 若录入次数配置错误，则返回25 00 00H。在不覆盖指纹状态下，若指纹库已满则返回 1f 00 00H；
 - 若指定 ID 号已存在模板则返回22 00 00H。
 - 指令合法性检测成功，则返回00 00 00H，并进入第一次指纹录入。
 - 2) 等待采图成功（返回 00 01 0nH）。
 - 3) 等待生成特征成功（00 02 0nH），如果生成特征失败（07 02 0nH），如果合并特征失败（0a 02 0nH），重新等待采图成功。

①如果注册逻辑模式配置成 1，即要求每次录入不同的手指区域，如果在生成特征时当次与前次特征重合面积过多（返回28 02 0nH），重新等待采图成功；

②如果注册逻辑模式配置成 2，即要求每次录入相似的手指区域，如果在生成特征时当此与前次特征不相似（返回 08 02 0nH），重新等待采图成功。
 - 4) 等待手指离开，第一次录入成功（00 03 0nH），手指离开后跳转到步骤 2，进入下一次循环直到 n 为设置录入的次数。注：若录入过程中设置为手指不需要离开，那么直接返回第一次录入成功，并跳转到步骤 2；最后一次采集指纹，没有手指离开录入成功的应答。
 - 5) 合成模板，得到模板的模板质量，成功返回 00 04 F0H，模板质量太差返回 3B 04 F0H。
 - 6) 指纹重复检查，指将新录入的手指与已经存储的手指进行匹配检查（通过设置参数 bit4 开启或者关闭此功能），若有相同指纹，则返回 27 05 F1H，结束流程；若没有相同指纹，则返回 00 05 F1H。
 - 7) 登记该模板数据，存储失败返回 01 06 F2H，结束流程；成功返回 00 06 F2H。
 - 8) 若收到 PS_Cancel 指令，则终止该指令并返回应答。

3.2.3 自动验证指纹 PS_Autoidentify

- 功能说明：自动采集指纹，在指纹库中搜索目标模板或整个指纹模板，并返回搜索结果。如果目标模板同当前采集的指纹比对得分大于最高阈值，并且目标模板为不完整特征则以采集的特征更新目标模板的空白区域。一站式搜索包含获取图像，生成特征，搜索指纹等功能。加密等级设置为 0 或 1 情况下支持此功能。
- 输入参数：安全等级、ID 号、参数
- 返回参数：确认字，页码（相配指纹模板）
- 指令代码：32H
- 指令包格式：

包头	设备地址	包标识	包长度	指令码	安全等级	ID 号	参数	校验和
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes	2 bytes	2 bytes
0xEF01	xxxx	01H	0008H	32H	xxH	xxxxH	xxxxH	sum

- 辅助说明：

ID 号：2bytes，大端模式。比如录入 1 号指纹，则是0001H。ID 号为0xFFFF，则进行 1：N 搜索；否进行

1:1 匹配。

参数：最低位为 bit0。

- 1) bit0: 预留;
- 2) bit1: 预留;
- 3) bit2: 注册过程中, 是否要求模块在关键步骤, 返回当前状态, 0-要求返回, 1-不要求返回;
- 4) bit3~bit15: 预留。

● 应答包格式:

包头	芯片地址	包标识	包长度	确认码	参数	ID 号	得分	校验和	备注
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes	2 bytes	2 bytes	
0xEF01	xxxx	07H	0008H	xxH	00H	xxxxH	xxxxH	sum	指令合法性检测: 合法/ ..
0xEF01	xxxx	07H	0008H	xxH	01H	xxxxH	xxxxH	sum	采图结果: 成功/超时
0xEF01	xxxx	07H	0008H	xxH	05H	xxxxH	xxxxH	sum	搜索结果: 成功/失败

● 确认码、参数 1 和参数 2 的返回值

确认码	释义	参数	释义
00H	成功	00H	指纹合法性检测
01H	失败	01H	获取图像
07H	生成特征失败	05H	已注册指纹比对
08H	指纹不匹配		
09H	没搜索到指纹		
0bH	ID 号超出范围		
17H	残留指纹		
23H	指纹模板为空		
24H	指纹库为空		
26H	超时		
27H	表示指纹已存在		
31H	功能与加密等级不匹配		

● 指令说明:

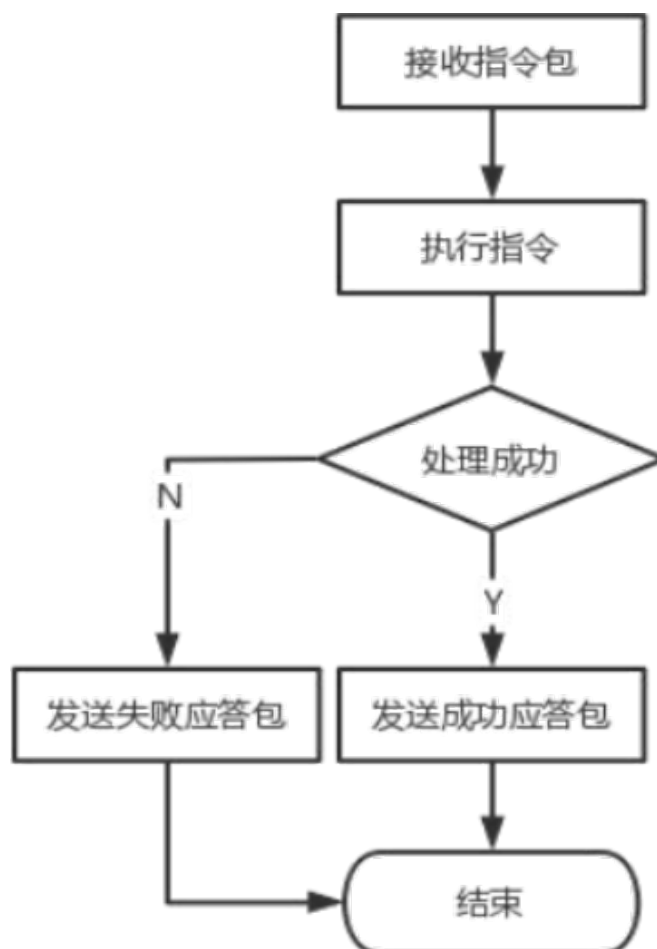
- 1) 若指纹库为空, 则确认码和参数返回 (以下直接描述为返回): 24 00H。若指定 ID 号无效, 则返回 0b 00H。若已登记的 Template 不存在, 则返回 23 00H。
- 2) 指令合法性检测成功, 返回 00 00H, 并进入指纹录入。
- 3) 在设定的超时时间内, 若没有完成一次完整的指纹录入, 则返回 26 00H, 结束流程。
- 4) 检查输入的指纹图像的正确性。若不正确, 则等待下次采集图像。
- 5) 若输入指纹正确, 则返回 00 01H, 即录入指纹获取图像成功。

- 6) 若生产特征失败，则返回 09 05H，结束流程。
- 7) 生成特征成功后，把当前采集到的指纹模板与已登记的指纹模板之间进行比对，并返回其结果。若比对失败，则返回 09 05H，结束流程；若比对成功，则返回 00 05H，以及正确的 ID 号码和得分。
- 8) 若收到 PS_Cancel 指令，则终止该指令并返回应答。

4、功能演示示例

该演示要确保内部指纹模组属于激活状态，并且如果需要验证口令，要确保口令验证通过。

4.1 UART 命令包的处理过程



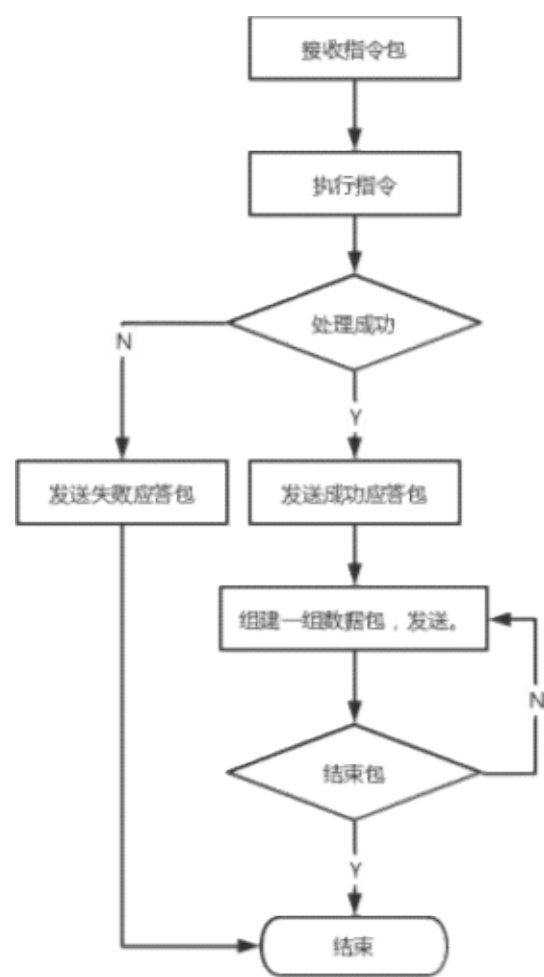
功能实现示例 1：UART 命令包的处理过程

4.2 UART 数据包的发送过程

UART 传输数据包前，首先要接收到传输数据包的指令包，做好传输准备后发送成功应答包，最后才开始传输数据包。数据包主要包括：包头、设备地址、包标识、包长度、数据和校验和。

数据包的包标识主要分为两种：02H 和 08H。02H：数据包，且有后续包。08H：最后一个数据包，即结束包。数据长度是预先设置好的，主要分为：32、64、128、和 256 四种。

例如，要传输的数据长度为 1K bytes，数据包中预先设置的数据长度为 128 bytes，那么就要把 1K bytes 的数据分为 8 个数据包传输。每个数据包包括：2 bytes 包头、4 bytes 芯片地址、1 byte 包标识、2 bytes 包长度、128 bytes 数据和 2 bytes 校验和，每个数据包长度为 139 bytes。另外，8 个数据包中，前 7 个数据包的报标识是 02H，最后一个结束数据包报标识是 08H。最后需要注意的是，结束包如果长度没有达到 139 bytes 时，以实际长度传输，不会以其他方式扩充到 139 bytes。



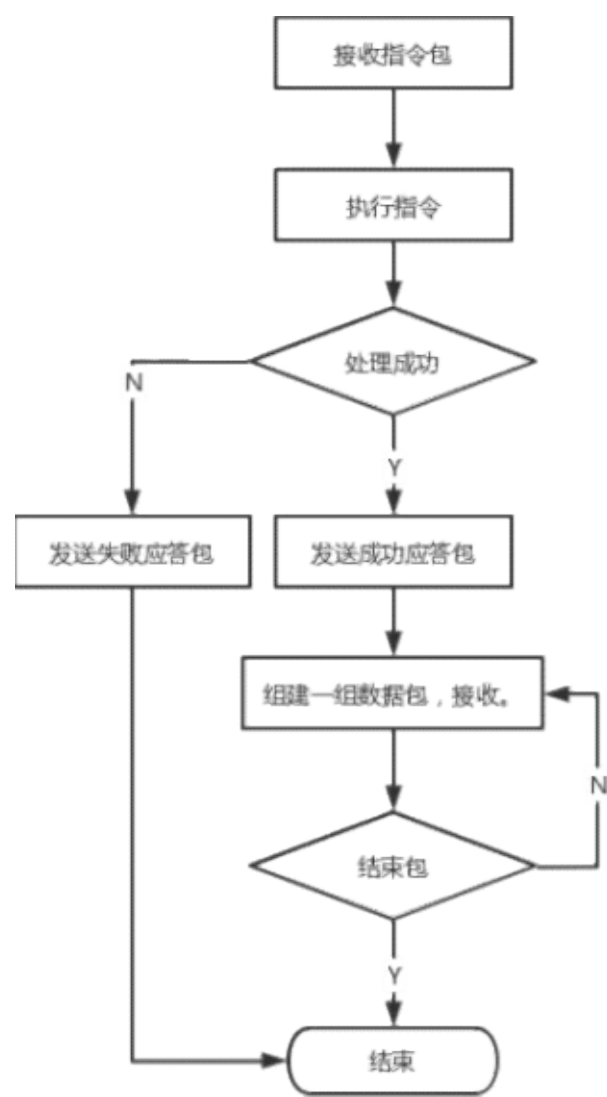
功能实现示例 2：UART 数据包的发送过程

4.3 UART 数据包的接收过程

UART 传输数据包前，首先要接收到传输数据包的指令包，做好传输准备后发送成功应答包，最后才开始 传输数据包。数据包主要包括：包头、设备地址、包标识、包长度、数据和校验和。

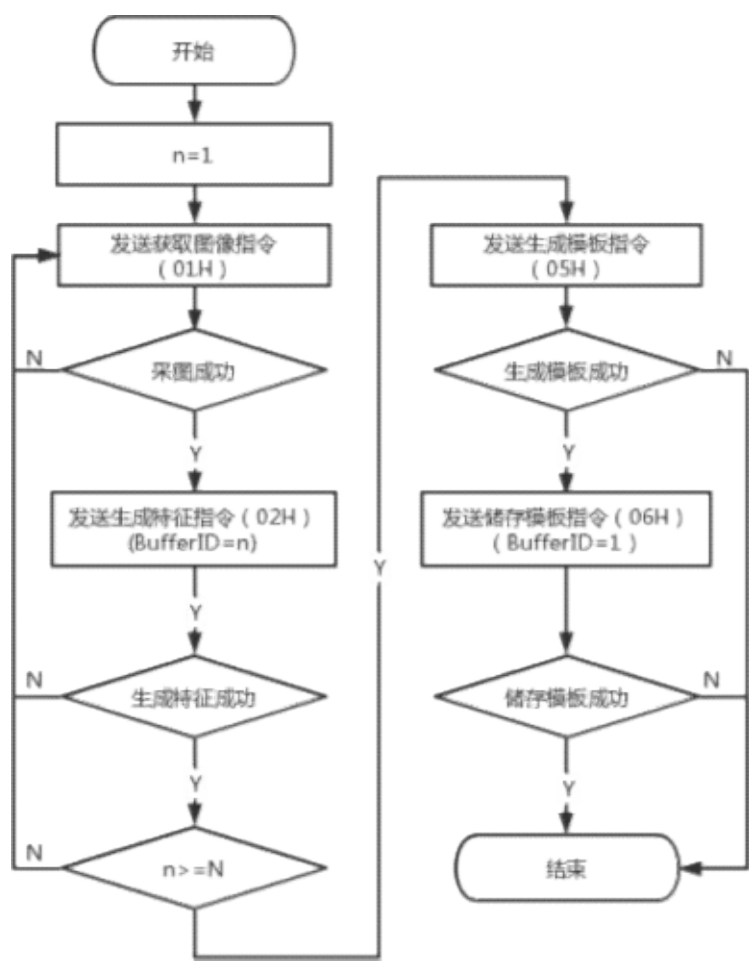
数据包的包标识主要分为两种：02H 和 08H。02H：数据包，且有后续包。08H：最后一个数据包，即结 束包。数据长度是预先设置好的，主要分为：32、64、128、和256 四种,。

例如，要传输的数据长度为 1K bytes，数据包中预先设置的数据长度为 128 bytes，那么就要把 1K bytes 的数据分为 8 个数据包传输。每个数据包包括：2 bytes 包头、4 bytes 芯片地址、1 byte 包标识、2 bytes 包 长度、128 bytes 数据和2 bytes 校验和，每个数据包长度为 139 bytes。另外，8 个数据包中，前 7 个数据包 的报标识是02H，最后一个结束数据包报标识是08H。最后需要注意的是，结束包如果长度没有达到 139 bytes 时，以实际长度传输，不会以其他方式扩充到 139 bytes。



功能实现示例 3：UART 数据包的接收过程

4.4 注册指纹流程

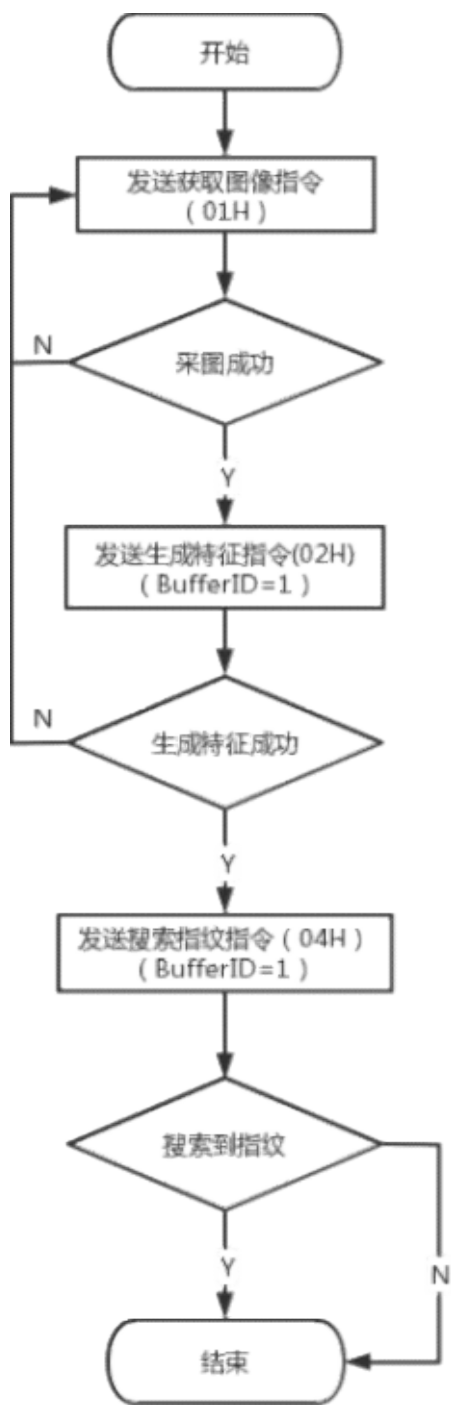


功能实现示例 4：注册指纹流程

如表 3 中，在注册过程中，如要控制每次录入的手指区域，可以通过写系统寄存器指令来配置注册逻辑模式（注册逻辑模式默认为 0，每次录入时对手指区域不作控制）。

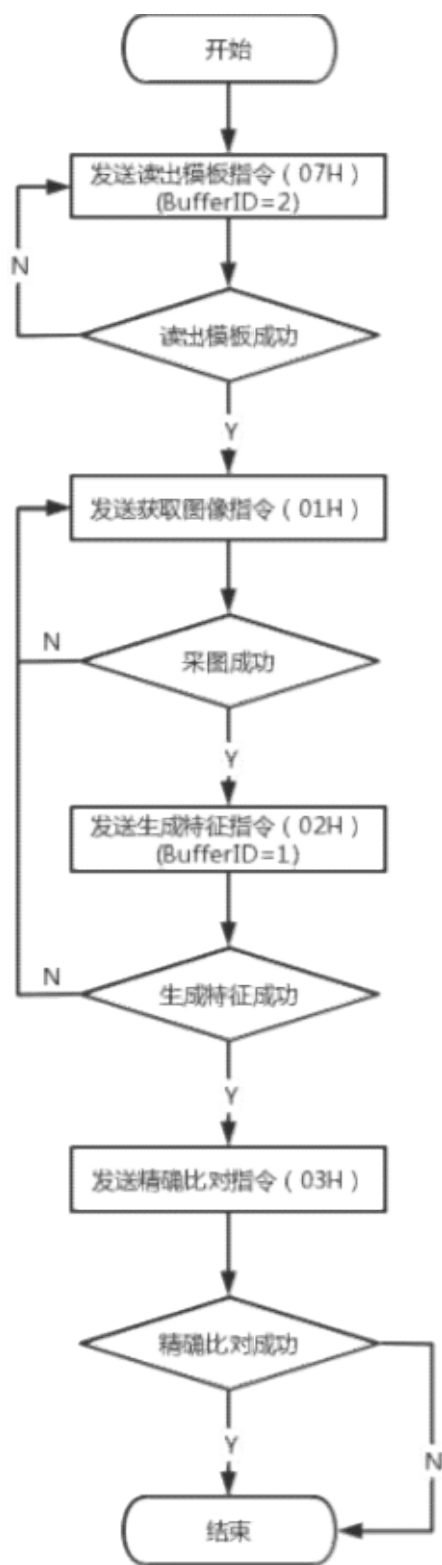
- 1、如要求每次录入不同的手指区域，则将注册逻辑模式配置成 1，发送该指令 EF 01 FF FF FF FF 01 00 05 0E 03 01 00 18，在生成特征时如果当次与前次特征重合面积过多，模组应答包中的确认码为 0x28；
- 2、如要求每次录入相似的手指区域，则将注册逻辑模式配置成 2，发送该指令 EF 01 FF FF FF FF 01 00 05 0E 03 02 00 19，在生成特征时如果当此与前次特征不相似，模组应答包中的确认码为 0x08。

4.5 搜索指纹流程



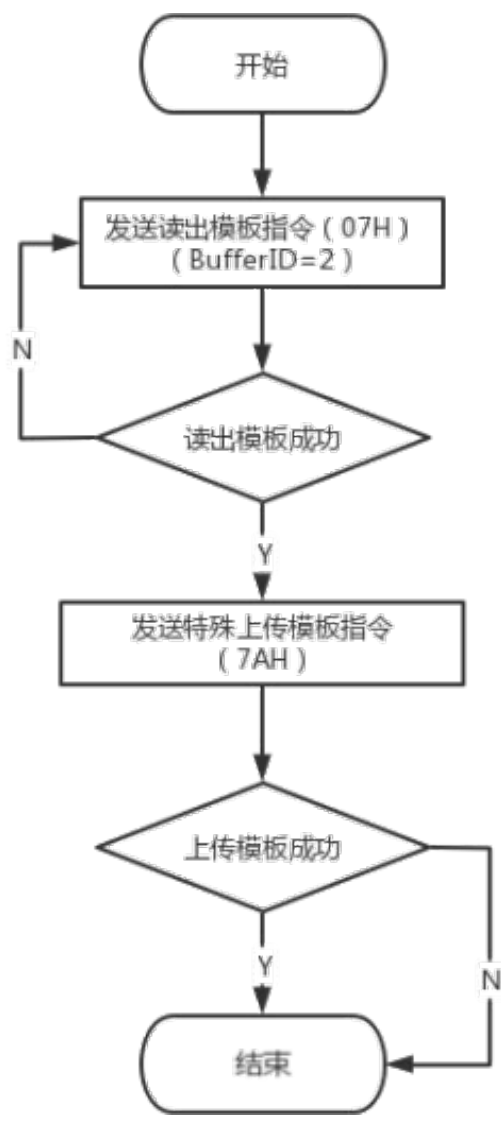
功能实现示例 5：搜索指纹流程

4.6 主控加载一个指纹特征或者模板进行精确比对



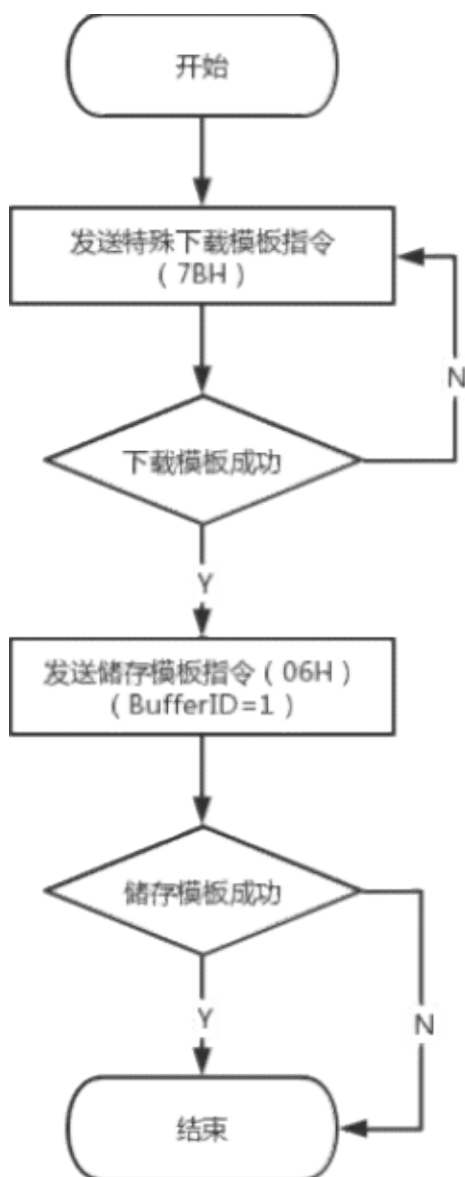
功能实现示例 6：主控加载一个指纹特征或者模板进行精确比对

4.7 特殊上传模板流程



功能示例 7：特殊上传模板流程

4.8 特殊下载模板流程



功能示例 8：特殊下载模板流程

5、返回类型表

序号	值	含义
1	0x00	指令执行完毕或 OK
2	0x01	数据包接收错误
3	0x02	传感器上没有手指
4	0x03	录入指纹图像失败
5	0x04	指纹图像太干、太淡而生不成特征
6	0x05	指纹图像太湿、太糊而生不成特征
7	0x06	指纹图像太乱而生不成特征
8	0x07	指纹图像正常，但特征点太少（或面积太小）
9	0x08	指纹不匹配
10	0x09	没搜索到指纹
11	0x0A	特征合并失败
12	0x0B	访问指纹库时地址序号超出指纹库范围
13	0x0C	从指纹库读模板出错或无效
14	0x0D	上传特征失败
15	0x0E	模块不能接收后续数据包
16	0x0F	上传图像失败
17	0x10	删除模板失败
18	0x11	清空指纹库失败
19	0x12	不能进入低功耗状态
20	0x13	口令不正确
21	0x14	系统复位失败
22	0x15	缓冲区内没有有效原始图而生不成图像
23	0x16	在线升级失败
24	0x17	残留指纹或两次采集之间手指没有移动过
25	0x18	读写 FLASH 出错
26	0x19	随机数生成失败
27	0x1A	无效寄存器号
28	0x1B	寄存器设定内容错误号
29	0x1C	记事本页码指定错误
30	0x1D	端口操作失败
31	0x1E	自动注册（enroll）失败
32	0x1F	指纹库满
33	0x20	设备地址错误
34	0x21	密码有误
35	0x22	指纹模板非空
36	0x23	指纹模板为空
37	0x24	指纹库为空
38	0x25	录入次数设置错误
39	0x26	超时

序号	值	含义
40	0x27	指纹已存在
41	0x28	指纹特征有关联
42	0x29	传感器初始化失败
43	0x2A	模组信息非空
44	0x2B	模组信息为空
45	0x2C	OTP 操作失败
46	0x2D	密钥生成失败
47	0x2E	密钥不存在
48	0x2F	安全算法执行失败
49	0x30	安全算法加解密结果有误
50	0x31	功能与加密等级不匹配
51	0x32	密钥已锁定
52	0x33	图像面积小
53	0x34	图像中静态异物（Orange）
54	0x35	非法数据
56	0x37	特征中静态异物（Orange）
57	0xFB	模组忙碌（正询问是否关闭模组，返回之后模组退出询问，定时器记时复位）
58	0xFC	该操作被屏蔽
59	0xFD	参数错误
60	0xFE	指纹模组未打开
61	0xFF	被动激活

6. CRC 计算方法示例（C 语言）

buffer 表示指令包缓存，size 表示指令包长度。

6.1 校验

```
uint8_t crc_check(uint8_t *buffer, uint16_t size) {
    uint16_t data = 0;
    for(uint8_t i = 0; i < (size-8); i++) {
        data += buffer[6+i];
    }
    if(data == (buffer[size-2]<<8 | buffer[size-1])){
        return 1;
    }
    return 0;
}
```

6.2 计算

```
uint16_t crc_sum(uint8_t *buffer, uint16_t size) {  
    uint16_t data = 0;  
    for(uint8_t i = 0; i < (size-8); i++) {  
        data += buffer[6+i];  
    }  
    return data;  
}
```