

SM30指纹识别模块

使用手册

Fingerprint Identification Module User's Guide

(Ver 4.3)

深圳市凯迈生物识别技术有限公司

<http://www.szcama.com> <http://www.camabio.com>

2021年2月

目 录

一、概述.....	1
二、模块硬件描述.....	2
2.1 硬件特性.....	2
2.2 外形及接口定义.....	2
2.2.1 外观图片.....	2
2.2.3 指纹模块及传感器尺寸图.....	3
三、模块的技术特性.....	4
3.1 技术特性.....	4
3.2 出厂设置值.....	4
四、通讯协议.....	6
4.1 通讯处理过程.....	6
4.2 通讯包 PACKET 的分类.....	7
4.2.1 命令包 Command packet.....	7
4.2.2 响应包 Response packet.....	7
4.2.3 指令/响应的数据包 Data Packet.....	7
4.3 通讯包的帧结构.....	8
4.3.1 通讯包Packet识别代码.....	8
4.3.2 命令包 (Command packet) 的帧结构.....	8
4.3.3 响应包 (Response packet) 的帧结构.....	9
4.3.4 指令数据包的帧结构.....	9
4.3.5 响应数据包 (Response data packet) 的帧结构.....	10
五、通讯命令COMMAND的详细说明.....	11
5.1 指纹特征模板 (TEMPLATE RECORD) 的数据结构.....	11
5.2 命令列表表 COMMAND LIST.....	12
5.3 COMMAND 的详细说明.....	14
5.3.1 指纹验证 Verify.....	14
5.3.2 指纹识别 Identify.....	16
5.3.3 注册指令 Enroll.....	18
5.3.4 一次注册命令 Enroll One Time.....	20
5.3.5 删除指定的指纹模板 Clear Template.....	22
5.3.6 清空所有指纹模板 Clear All Template.....	23
5.3.7 获取可注册的指纹模板编号 Get Empty ID.....	24
5.3.8 获取指定模板编号的状态 Get Template Status.....	25
5.3.9 获取指纹模板数据库的完整性 Get Broken Template.....	26
5.3.10 读取指定的指纹模板数据 Read Template.....	27

5.3.11 写入指纹模板数据至指定的模板编号中 Write Template.....	29
5.3.12 设置识别安全等级命令 Set Security Level.....	32
5.3.13 获取上次设置的安全等级 Get Security Level.....	35
5.3.14 设置指纹采集超时参数 Set Finger Time Out.....	36
5.3.15 获取模块当前的超时参数值 Get Finger Time Out.....	37
5.3.16 设置设备编号 Set Device ID.....	38
5.3.17 获取设备编号 Get Device ID.....	39
5.3.18 获取固件版本号 Get F/W Version.....	40
5.3.19 检测是否有指纹按下指令 Finger Detect.....	41
5.3.20 设置通讯波特率指令 Set Baudrate.....	42
5.3.21 设置指纹重复检查的允许/禁止 Set Duplication Check.....	43
5.3.22 获取模块当前的指纹重复检查的状态 Get Duplication Check.....	44
5.3.23 控制模块进入待机状态 Enter Standby Mode.....	45
5.3.24 注册指纹并暂存在RAM中 Enroll And Store in RAM.....	46
5.3.25 获取注册指纹的模板数据 Get Enroll Data.....	48
5.3.26 采集图像并提取指纹特征值数据上传 Get Feature Data of Captured FP.....	50
5.3.27 下载特征数据与采集指纹比对VerifyDownloadedFeatureWithCapturedFP.....	52
5.3.28 下载2枚模板与采集的指纹比对IdentifyDownloadedFeatureWithCapturedFP..	55
5.3.29 获取设备名称 Get Device Name.....	58
5.3.30 指纹采集器背光控制命令 Sensor LED Control.....	59
5.3.31 连续指纹识别 Identify Free.....	60
5.3.32 获取已注册指纹总数 Get Enroll Count.....	62
5.3.33 取消有关指纹采集的操作 FP Cancel.....	63
5.3.34 连接状态检测命令 Test Connection.....	65
5.3.35 错误操作命令 Incorrect Command.....	67
5.4 注意事项.....	68
六、附录.....	69
6.1 响应 (RESPONSE) 及错误代码表.....	69
6.2 登记及比对流程图.....	71
6.2.1 注册流程 Enroll.....	71
6.2.2 验证及识别流程 Verify & Identify.....	72

一、概述

SM30为指纹采集和单片指纹处理器二合一设计的指纹识别二次开发模块，具有体积小、功耗低、接口简单、可靠性高、指纹模板小（496字节）、大容量指纹识别（2000枚指纹识别响应时间小于1秒）等优点，可以非常方便将其嵌入用户系统，组成满足客户需求的指纹识别产品。

尤其具备自学习功能，指纹识别过程中，最新提取的指纹特征值识别成功后将该特征值融合到指纹数据库中，使用户在使用过程中越用越好用。

SM30模块通讯接口为UART，本模块作为从设备，由主设备发送相关命令对其进行控制。

该模块具有可调节的安全等级功能、指纹特征数据的读/写功能和 1:N 识别及 1:1 验证功能等。

主要功能包括如下：

- 通过 UART 收发指令
- 最小存储条件下实现指纹数据的登记及比对：指纹模板为 496 字节
- 光学全反射指纹传感器
- 1:N 识别 及 1:1 验证功能
- 具有循环采集比对功能（Identify Free）：主机 HOST 发送该指令后模块一直处于采集指纹比对并将比对结果返回 HOST
- 指纹特征数据的读/写功能
- 可适当调节的安全等级
- 采集指纹并提取特征值上传，下载指纹特征值与当前采集的指纹比对功能（该功能特别适合于将指纹存储在接触式 IC 卡或非接触式 IC 卡中以验证是否持卡人本人）
- 可设置的波特率
- 可设置的设备编号
- 可以设置本模块访问密码（14 个数字字符组成的密码），使本模块处于口令保护模式，极大提高了通讯的安全性。

应用领域：

- 指纹门禁控制
- 指纹签到、指纹考勤机等
- 指纹锁、指纹保险柜等
- 指纹 POS 终端机等手持设备应用

二、模块硬件描述

2.1 硬件特性

表2-1 硬件特性

No	项 目	内 容
1	型号	SM30
2	CPU	AS602
3	Sensor	FPC1011半导体指纹传感器
4	指纹图像大小	210 x 250 (pixel)
5	输入电压	DC3.3V
6	工作电流	小于100mA
8	通讯方式	UART

2.2 外形及接口定义

2.2.1 外观图及接口信号定义

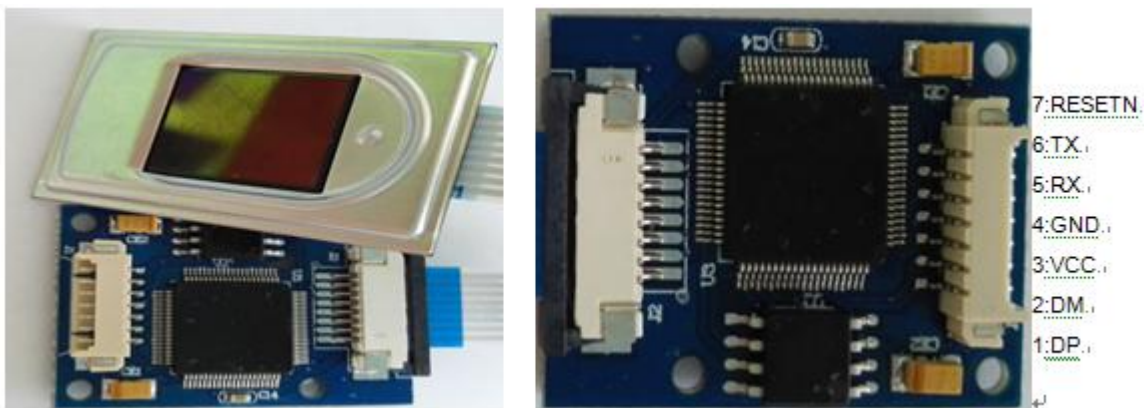


图2-1 SM30指纹识别模块外观图

- 信号定义：1: DP USB +端
 2: DM USB - 端
 3: Vcc 电源输入正端 +3.3V
 4: GND 电源地
 5: Rx 模块接收端
 6: Tx 模块发送端
 7: RESET 复位(不用保留)

注：串行通讯为TTL电平

2.2.2 SM30 主处理板尺寸图

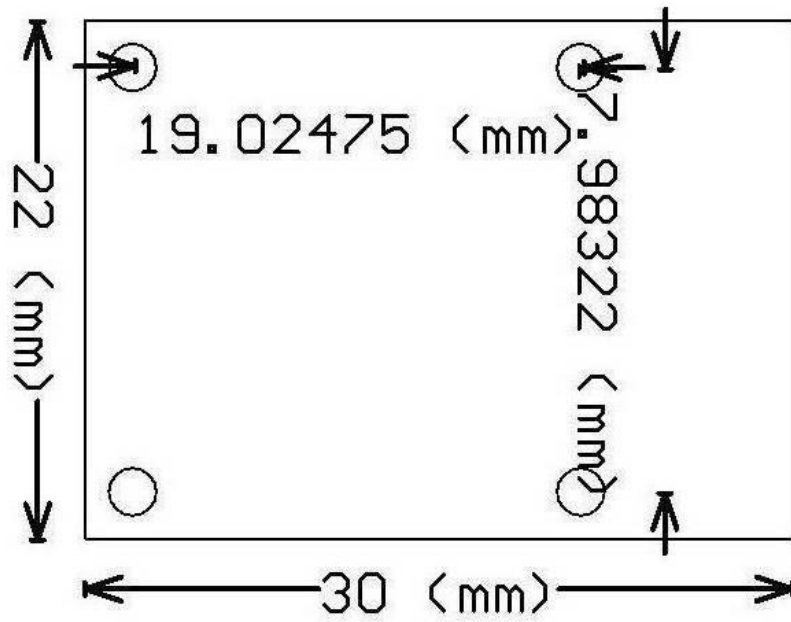


图2-2 SM1S-30主处理板尺寸图

2.2.3 指纹传感器尺寸图

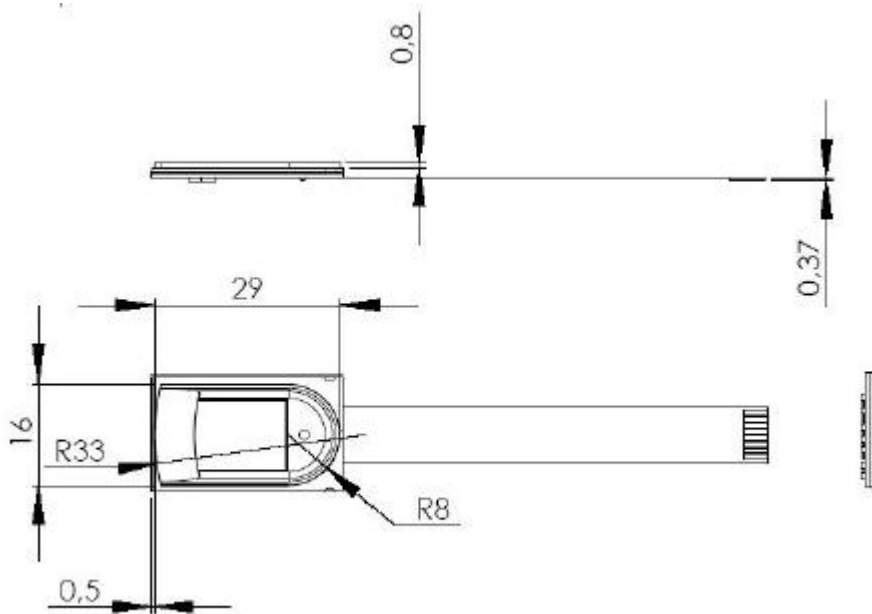


图2-3 FPC1011指纹传感器尺寸图

三、模块的技术特性

3.1 技术特性

表3-1 SM30模块技术特性

项目	描述	备注
CPU	CPU: AS602 120MHz主频、1M Bits Flash Memory、128KB SRAM	
指纹采集器	FPC1011半导体指纹传感器	
指纹登记容量	标准配置为3000枚(可扩展)	
认假率 (FAR)	< 0.001 % (安全等级Security Level 为 3)	
拒真率 (FRR)	< 0.1 % (安全等级Security Level 为 3)	
比对方式	1:N 及 1:1 比对	
指纹模板大小	496字节 Byte	
安全等级设置	Level 1 至Level 5可设置, 出厂设置值: Level 3	
处理速度	指纹特征提取取时间< 0.45 秒	
	1:N 比对时间 (2000 枚满注册时) < 1.5 秒	
通讯接口	UART: TTL电平	
串行通讯参数	无奇偶校验Parity = NONE, 1个停止位Stop Bit = 1 无流控Flow Control = NONE 波特率为9600、19200、38400、57600、115200 (bps) 出厂设置: 115200bps	
工作电压	DC3.3V(可定制DC5V)	
工作电流	小于100mA	
工作环境	工作温度: -10 ℃- 60 ℃ 相对湿度: 20% - 80%	

3.2 出厂设置值

表3-2 出厂设置值

项目	初始值
安全等级 Security Level (1~5)	3
指纹采集超时时间 Finger Print Time Out (0~60S)	5 秒
波特率 BaudRate (9600, 19200, 38400, 57600, 115200BPS)	115200 bps

指纹重复检查 Duplication Check (ON/OFF)

ON

四、通讯协议

4.1 通讯处理过程

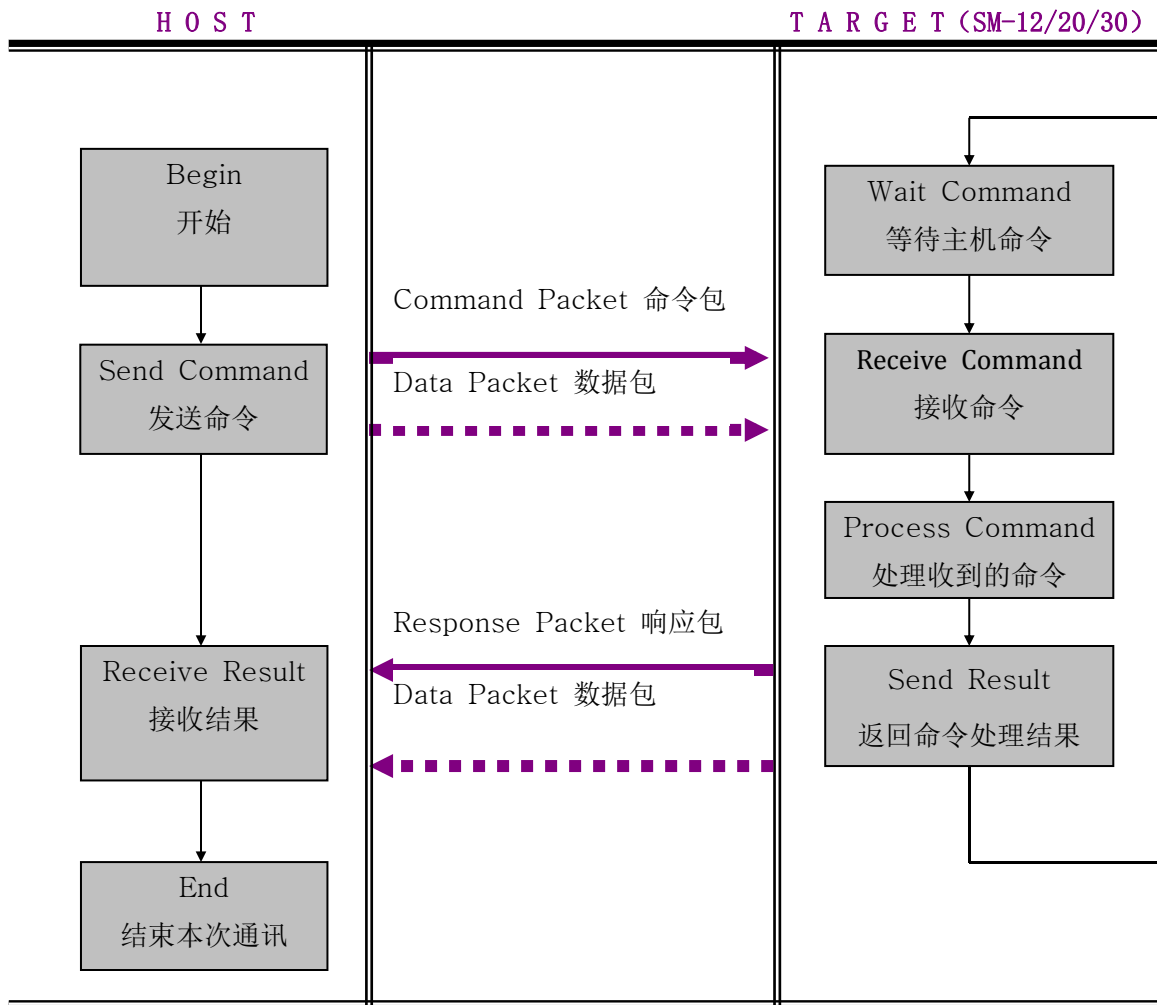


图4-1 通讯过程

注:

通讯过程除FP Cancel指令之外，其它指令的发送、接收必须要遵循一发一收的原则。

Host在没有收到应答时，请不要向 TARGET 发送指令。

4.2 通讯包 Packet 的分类

4.2.1 命令包 Command packet

- 命令包说明从 Host 至 Target 的指令内容。
- 从 Host 中发出的所有指令，都通过命令包Command packet传输。
- 命令包Command packet 的帧长度为 **24字节bytes**。

4.2.2 响应包 Response packet

- 响应包指从 Target (SM2A-12/20) 至 Host 的应答内容。
- 所有指令收到相应处理结果即Response packet 后终止其使命。
- 响应包Response packet 的长度为 **24字节byte** 。

4.2.3 指令/响应的数据包 Data Packet

- 当指令参数或响应数据的长度大于**16byte**时，利用指令/响应数据包Data Packet传输数据。
- Host须在发送指令数据包之前，利用命令包Command packet将指令数据包Data Packet的长度告知模块Target
- 指令参数或相应数据包的最大长度为 **512byte** 。

4.3 通讯包的帧结构

4.3.1 通讯包Packet识别代码

通讯包Packet的开始 2byte 为表示通讯包packet种类的识别码，其如下表4-1：

表4-1 Packet识别代码

Packet类别	Code包类别识别码
命令包Command packet	0xAA55
响应包Response packet	0x55AA
指令数据包Data Packet	0xA55A
响应数据包Response Data Packet	0x5AA5

4.3.2 命令包（Command packet）的帧结构

PREFIX 包识别码		CMD 命令字		LEN 长度		DATA 数据				CKS 校验和	
0x55	0xAA	L	H	L	H	D0	D1	...	D15	L	H
0	1	2	3	4	5	6	7	...	21	22	23

表4-2 Command packet 的结构

OFFSET	FIELD	TYPE	SIZE	DESCRIPTION
0	PREFIX	WORD	2byte	通讯包识别码 Packet Identify code
2	CMD	WORD	2byte	命令码 Command Code
4	LEN	WORD	2byte (=n, n < 17)	命令参数的长度 Length of Command Parameter
6	DATA	Byte array	16byte	命令参数Command Parameter (实际数据为n byte)
22	CKS	WORD	2byte	校验和Checksum 计算方法：从PREFIX 开始到DATA 数据结束按字节进行算数和运算，取计算结果的最低2字节

4.3.3 响应包 (Response packet) 的帧结构

PREFIX 包识别码		RCM 响应命令		LEN 长度		RET 结果码		DATA 数据				CKS 校验和	
0xAA	0x55	L	H	L	H	L	H	D0	D1	...	D13	L	H
0	1	2	3	4	5	6	7	8	9	...	21	22	23

表4-3 Response packet的结构

OFFSET	FIELD	TYPE	SIZE	DESCRIPTION
0	PREFIX	WORD	2byte	通讯包识别码 Packet Identify code
2	RCM	WORD	2byte	响应码Response Code
4	LEN	WORD	2byte(=n, n < 17)	响应包的长度: RET and DATA
6	RET	WORD	2byte	结果码 Result Code (0 :成功, 1 :失败)
8	DATA	Byte array	14byte	响应的数据 Response Data 实际数据为 (n-2) byte
22	CKS	WORD	2byte	校验和Checksum 计算方法: 从PREFIX 开始到DATA 数据结束按字节进行算数和运算, 取计算结果的最低2 byte

4.3.4 指令数据包的帧结构

PREFIX 包识别码		CMD 命令字		LEN 长度		DATA 数据				CKS 校验和	
0x5A	0xA5	L	H	L	H	D0	D1	...	Dn-1	L	H
0	1	2	3	4	5	6	7	...	6+n-1	6+n	6+n+1

表4-4 指令数据packet的结构

OFFSET	FIELD	TYPE	SIZE	DESCRIPTION
0	PREFIX	WORD	2byte	通讯包识别码 Packet Identify code
2	CMD	WORD	2byte	命令码 Command Code
4	LEN	WORD	2byte(=n , n < 512)	命令参数的长度 Length of DATA

6	DATA	Byte Array	nbyte	命令参数 Command parameter
6+n	CKS	WORD	2byte	校验和Checksum 计算方法：从PREFIX 开始到DATA 数据结束按字节进行算数和运算， 取计算结果的最低2 byte

Host须在发送指令数据包之前先传输命令包（Command packet），使得模块Target 进入指令数据包接收等待状态。在该命令包（Command packet）的数据域（DATA field）中，须设定待传输的指令数据包的长度。

Host 应在确认模块Target 的指令数据 packet 接收等待状态后传输指令数据包。

4.3.5 响应数据包（Response data packet）的帧结构

PREFIX 包识别码		RCM 响应码		LEN 长度		RET 结果码		DATA 数据				CKS 校验和	
0xA5	0x5A	L	H	L	H	L	H	D0	D1	...	Dn-3	L	H
0	1	2	3	4	5	6	7	8	9	...	6+n-1	6+n	6+n+1

表4-5 Response data packet的结构

OFFSET	FIELD	TYPE	SIZE	DESCRIPTION
0	PREFIX	WORD	2byte	通讯包识别码 Packet Identify code
2	CMD	WORD	2byte	响应码 Response Code
4	LEN	WORD	2byte(=n, n < 512)	结果数据长度 Length of result data (RET + DATA)
6	RET	WORD	2byte	结果码 Result code (0 : 成功、1 : 失败)
8	DATA	Byte Array	(n-2) byte	响应数据 Response data
6+n	CKS	WORD	2byte	校验和Checksum 计算方法：从PREFIX 开始到DATA 数据结束按字节进行算数和运算， 取计算结果的最低2 byte

注：从模块Target 至 Host 中传输14byte以上数据时，利用响应数据包（Response data packet）

五、通讯命令Command 的详细说明

5.1 指纹特征模板 (Template Record) 的数据结构

表5-1 Template Record的结构

Template Data	Checksum
496 byte	2 byte
Template Data	对模板数据Template Data依次按字节进行算数和运算，取计算结果的最低2 byte

注：

每个指纹特征模板数据为498字节：Template Data (496Bytes) +Checksum (2Bytes)

5.2 命令列表表 Command List

表5-2 Command列表

No	Function	Command			Response			
		CMD	LEN	DATA	RCM	LEN	RET	DATA
1	Verify	0x0101	2	Template No.	0x0101	4	0/1	Template No /Error Code
2	Identify	0x0102	0	-	0x0102h	4	0/1	Template No /Error Code
3	Enroll	0x0103	2	Template No.	0x0103	4/6	0/1	Template No /Error Code + Duplicaion ID
4	Enroll One Time	0x0104	2	Template No.	0x0104	4/6	0/1	Template No /ErrorCode
5	Clear Template	0x0105	2	Template No.	0x0105	4	0/1	Template No /ErrorCode
6	Clear All Template	0x0106	0	-	0x0106	4	0/1	Cleared Template Count /Error Code
7	Get Empty ID	0x0107	0	-	0x0107	4	0/1	Template No /Error Code
8	Get Template Status	0x0108	2	Template No.	0x0108	4	0/1	Template Status /Error Code
9	Get Broken Template	0x0109	0	-	0x0109	6	0/1	Broken Template Count + First Broken Template No /Error Code
10	Read Template	0x010A	2	Template No.	0x010A	(Template Record Size + 4) or (4)	0/1	Template No + Template Record Data /Error Code
11	Write Template	0x010B	Template Record Size + 2	Template No + Template Record Data	0x010B	4	0/1	Template No /Error Code
12	Set Security Level	0x010C	2	Security Level Value	0x010C	4	0/1	Security Level Value /Error Code
13	Get Security Level	0x010D	0	-	0x010D	4	0	Security Level Value
14	Set Finger Time Out	0x010E	2	Time Out Value	0x010E	4	0/1	TimeOut Value /Error Code
15	Get Finger Time Out	0x010F	0	-	0x010F	4	0	TimeOut Value

No	Function	Command			Response			
		Command	Length	Content	Response	Length	Content	Error Code
16	Set Device ID	0x0110	2	Device ID	0x0110	4	0	Device ID
17	Get Device ID	0x0111	0	-	0x0111	4	0/1	Device ID /Error Code
18	Get F/W Version	0x0112	0	-	0x0112	4	0	F/W Version
19	Finger Detect	0x0113	0	-	0x0113	4	0	Detect Result
20	Set BaudRate	0x0114	2	BaudRate Index	0x0114	4	0/1	BaudRate Index /Error Code
21	Set Duplication Check	0x0115	2	Duplication Check Option (1/0)	0x0115	4	0/1	Duplication Check Option /Error Code
22	Get Duplication Check	0x0116	0	-	0x0116	4	0	Duplication Check Option
23	Enter Standby Mode	0x0117	0	-	0x0117	4	0	-
24	Enroll And Store in RAM	0x0118	0	-	0x0118	4	0/1	0 / ErrorCode
25	Get Enroll Data	0x0119	0	-	0x0119	(Template Record Size + 2) or (4)	0/1	Template Record Data /Error Code
26	Get Feature Data of Captured FP	0x011A	0	-	0x011A	(Template Record Size + 2) or (4)	0/1	Template Record Data /Error Code
27	Verify Downloaded Feature with Captured FP	0x011B	Template Record Size	Template Record Data	0x011B	4	0/1	0/Error Code
28	Identify Downloaded Feature with Captured FP	0x011C	Template Record Size + 2	Index + Template Record Data	0x011C	4	0/1	0/Error Code
29	Get Device Name	0x0121	0	-	0x0121	16	0/1	"STO20-OEM" /Error Code
30	Sensor LED Control	0x0124	2	0/1	0x0124	4	0	0
31	Identify Free	0x0125	0	-	0x0125	4	0/1	Template No /Error Code
32	Set Device Password	0x0126	14	Password	0x0126	4	0/1	0/ Error Code
33	Verify Device Password	0x0127	14	Password	0x0127	4	0/1	0/ Error Code
34	Get Enroll Count	0x0128	0	-	0x0128	4	0/1	Enroll Count /Error Code
35	FP Cancel	0x0130	0	-	0x0130	2	0	-
36	Test Connection	0x0150	0	-	0x0150	2	0	-
37	Incorrect Command	-	-	-	0x0160	2	0	-

5.3 Command的详细说明

5.3.1 指纹验证 Verify

- [功能]

对指定编号的指纹模板与当前采集的指纹之间进行 1:1 验证并返回其结果。

- [工作过程 Sequence]

- ①、若指定 Template 号码无效，则返回错误码 ERR_INVALID_TMPL_NO 。
- ②、若指定号码中不存在指纹 Template ，则返回错误码 ERR_TMPL_EMPTY 。
- ③、若在 Time Out 时间内没检测到指纹的输入，则返回 ERR_TIME_OUT 。
- ④、检查被输入图像的正确性。若不正确，则返回相应错误码（ERR_BAD_QUALITY 等）。
- ⑤、检查被输入图像的正确性。若正确，则返回 GD_NEED_RELEASE_FINGER。
- ⑥、指定号码上的 Template 与输入图像之间进行比对并返回其结果。
 - 若比对成功，则 DATA 为指纹模板号 Template 号码。
 - 否则，RET 为 ERR_FAIL 且 DATA 为 ERR_VERIFY。
- ⑦、若收到 FpCancel 指令，则终止该指令并返回 ERR_FP_CANCEL。

- [Command 及 Response]

表5-3 Verify指令

PREFIX	0xAA55	
CMD	0x0101	
LEN	2	
DATA	待比对的 Template 号码	
PREFIX	0x55AA	
RCM	0x0101	
LEN	4	
RET	成功：ERR_SUCCESS，失败：ERR_FAIL	
DATA	2byte	成功：GD_NEED_RELEASE_FINGER 或 Template 号码 失败：错误码

- [指令实例]

Host发送的命令包：

待比对的指纹与编号为1的指纹模板1：1验证

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x01	0x01	0x02	0x00	0x01	0x00	0x00	0x04	0x01

模块验证指纹成功的响应包：

在TimeOut时间内采集到指纹后：

先返回离开手指（GD_NEED_RELEASE_FINGER）提示

再返回比对结果，如下例：

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x01	0x01	0x04	0x00	0x00	0x00	0xF4	0xFF	...	0x00	0xF8	0x02
0xAA	0x55	0x01	0x01	0x04	0x00	0x00	0x00	0x01	0x00	...	0x00	0x06	0x01

5.3.2 指纹识别 Identify

- [功能]

当前采集的指纹与已登记的所有指纹模板之间进行 1:N 比对并返回其结果。

- [工作 Sequence]

- ①、若已登记的 Template 不存在，则返回错误码 ERR_ALL_TMPL_EMPTY。
- ②、若 Time Out 时间内没检测到指纹的输入，则返回 ERR_TIME_OUT。
- ③、检查输入图像的正确性。若不正确，则返回相应错误码（ERR_BAD_QUALITY 等）。
- ④、检查输入图像的正确性。若正确，则返回 GD_NEED_RELEASE_FINGER。
- ⑤、已登记的所有模板与当前采集到的指纹模板之间进行 1: N 比对并返回其结果。
 - 若比对成功，则返回 ERR_SUCCESS 和其号码。
 - 否则，返回 ERR_FAIL 且 DATA 为 ERR_IDENTIFY
- ⑥、若收到 FpCancel 指令，则终止该指令并返回 ERR_FP_CANCEL。

- [Command 及 Response]

表5-4 Identify指令

PREFIX	0xAA55	
CMD	0x0102	
LEN	0	
DATA	数据不存在	
PREFIX	0x55AA	
RCM	0x0102	
LEN	4	
RET	成功: ERR_SUCCESS, 失败: ERR_FAIL	
DATA	2byte	成功: GD_NEED_RELEASE_FINGER / Template 号码, 失败: 错误码

- [指令实例]

命令包:

指纹识别

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x02	0x01	0x00	0x00	0x00	0x00	0x00	0x02	0x01

响应包: 先返回离开手指 (GD_NEED_RELEASE_FINGER) 提示, 再返回识别结果

指纹识别成功

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x02	0x01	0x04	0x00	0x00	0x00	0xF4	0xFF	...	0x00	0xF9	0x02
0xAA	0x55	0x02	0x01	0x04	0x00	0x00	0x00	0x01	0x00	...	0x00	0x07	0x01

5.3.3 注册指令 Enroll

- [功能]

- 利用指纹登记过程中获取的三个临时的模板融合，生成一个模板数据并登记入库。
- 在三次指纹输入过程中，若临时模板不正确，则重新进行当次指纹输入。
- 若合成过程中出错，则重新进行三次指纹输入。

- [工作 Sequence]

- ①、若指定 Template 号码无效，则返回 ERR_INVALID_TMPL_NO 。
- ②、若指定 Template 号码中已登记有另外 Template，则返回 ERR_TMPL_NOT_EMPTY 。
- ③、返回 GD_NEED_FIRST_SWEEP 表示等待第一次指纹输入，在 Time Out 时间内检查图像输入。
- ④、若在设定的超时参数 Time Out 时间内检测不到指纹的输入，则返回 ERR_TIME_OUT 。
- ⑤、检查输入的指纹图像的正确性。若不正确，则返回相应错误码（ERR_BAD_QUALITY 等），并重新进入相应指纹等待状态。
- ⑥、若第一次输入指纹正确，则返回 GD_NEED_RELEASE_FINGER 告知离开手指并监测是否已离开。若已离开，则跳到⑦。
- ⑦、返回 GD_NEED_SECOND_SWEEP，表示第二次指纹输入的请求，且重复④、⑤。
- ⑧、若第二次输入指纹正确，则返回 GD_NEED_RELEASE_FINGER 告知离开手指并监测是否已离开，若已离开，则跳到⑨。
- ⑨、若第二次指纹输入成功，则返回 GD_NEED_THIRD_SWEEP，表示第三次指纹输入的请求并重复④、⑤。
- ⑩、若第三次输入指纹正确，则返回 GD_NEED_RELEASE_FINGER 告知离开手指。
- ⑪、利用前述过程中已获取的三个模板生成一个登记模板数据。若失败，则返回 ERR_GENERALIZE 。
- ⑫、若没有设定指纹重复检查 Duplication，则登记该模板并返回登记的模板号码。
- ⑬、若已设定了重复检查 Duplication Check，则该模板与所有已登记的所有模板进行比对。
 - 若有相同指纹，则返回该指纹的模板号码和 ERR_DUPLICATION_ID 。
 - 若没有相同指纹，则登记该模板数据并返回当前登记的指纹模板号码。
- ⑭、若收到 FpCancel 指令，则终止该指令并返回 ERR_FP_CANCEL。

- [命令及响应 Command and Response]

表5-5 Enroll指令

PREFIX	0xAA55	
CMD	0x0103	
LEN	2	
DATA	待登记的指纹模板编号	
PREFIX	0x55AA	
RCM	0x0103	
LEN	4/6	
RET	成功: ERR_SUCCESS, 失败: ERR_FAIL	
DATA	2byte	成功: GD_NEED_FIRST_SWEEP、GD_NEED_SECOND_SWEEP、 GD_NEED_THIRD_SWEEP、GD_NEED_RELEASE_FINGER、 新登记的 Template 号码。
	2byte	失败: 错误码 0 或 ERR_DUPLICATION_ID重复的ID编号

- [指令实例]

命令包:

注册第1枚指纹

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x03	0x01	0x02	0x00	0x01	0x00	0x00	0x06	0x01

响应包:

注册第1枚指纹成功

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x03	0x01	0x04	0x00	0x00	0x00	0xF1	0xFF	...	0x00	0xF7	0x02
0xAA	0x55	0x03	0x01	0x04	0x00	0x00	0x00	0xF4	0xFF	...	0x00	0xFA	0x02
0xAA	0x55	0x03	0x01	0x04	0x00	0x00	0x00	0xF2	0xFF	...	0x00	0xF8	0x02
0xAA	0x55	0x03	0x01	0x04	0x00	0x00	0x00	0xF4	0xFF	...	0x00	0xFA	0x02
0xAA	0x55	0x03	0x01	0x04	0x00	0x00	0x00	0xF3	0xFF	...	0x00	0xF9	0x02
0xAA	0x55	0x03	0x01	0x04	0x00	0x00	0x00	0xF4	0xFF	...	0x00	0xFA	0x02
0xAA	0x55	0x03	0x01	0x06	0x00	0x00	0x00	0x01	0x00	...	0x00	0x0A	0x01

5.3.4 一次注册命令 Enroll One Time

- [功能]

- 只进行一次指纹图像采集即完成登记。

- [工作过程 Sequence]

- ①、若指定的模板号码无效，则返回 ERR_INVALID_TMPL_NO 。
- ②、若指定的模板号码中已被登记，则返回 ERR_TMPL_NOT_EMPTY 。
- ③、若在 Time Out 时间内检测不到指纹的输入，则返回 ERR_TIME_OUT 。
- ④、检查指纹图像的正确性。若不正确，则返回相应错误码（ERR_BAD_QUALITY 等）。
- ⑤、检查指纹图像的正确性。若正确，则返回 GD_NEED_RELEASE_FINGER。
- ⑥、若没设定指纹重复检查 Duplication Check，则登记该 Template Data 模板数据并返回登记的 Template 模板号码。
- ⑦、若已设定了指纹重复检查 Duplication Check，则该模板与已登记的所有模板进行比对。
 - 若有相同指纹，则返回该指纹的 Template 模板号码和 ERR_DUPLICATION_ID 。
 - 若没有相同指纹，则登记该 Template Data 并返回登记的 Template 模板号码。
- ⑧、若收到 FpCancel 指令，则终止该指令并返回 ERR_FP_CANCEL。

- [命令及响应 Command and Response]

表5-6 Enroll One Time指令

PREFIX	0xAA55	
CMD	0x0104	
LEN	2	
DATA	待登记的 Template 号码	
PREFIX	0x55AA	
RCM	0x0104	
LEN	4/6	
RET	成功：ERR_SUCCESS，失败：ERR_FAIL	
DATA	2byte	成功：GD_NEED_RELEASE_FINGER / 新登记的 Template 号码。 失败：错误码
	2byte	0 或 ERR_DUPLICATION_ID重复的ID编号

- [指令实例]

命令包:

注册第1枚指纹

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x04	0x01	0x02	0x00	0x01	0x00	0x00	0x07	0x01

响应包:

注册第1枚指纹成功

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x04	0x01	0x04	0x00	0x00	0x00	0xF4	0xFF	...	0x00	0xFB	0x02
0xAA	0x55	0x04	0x01	0x04	0x00	0x00	0x00	0x01	0x00	...	0x00	0x09	0x01

5.3.5 删除指定的指纹模板 Clear Template

- [功能]
 - 删除指定编号中的指纹模板。
- [工作过程 Sequence]
 - ①、若指定的指纹模板编号无效，则返回 ERR_INVALID_TMPL_NO 。
 - ②、若指定的模板编号中不存在指纹模板数据，则返回 ERR_TMPL_EMPTY 。
 - ③、删除指定号码中的模板数据并返回其结果。
- [命令及响应Command and Response]

表5-7 Clear Template指令

PREFIX	0xAA55	
CMD	0x0105	
LEN	2	
DATA	待删除的Template号码	
PREFIX	0x55AA	
RCM	0x0105	
LEN	4	
RET	成功：ERR_SUCCESS，失败：ERR_FAIL	
DATA	2byte	成功：已删除的Template号码，失败：错误码

- [指令实例]

命令包：

删除第1枚指纹

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x05	0x01	0x02	0x00	0x01	0x00	0x00	0x08	0x01

响应包：

删除第1枚指纹成功

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x05	0x01	0x04	0x00	0x00	0x00	0x01	0x00	...	0x00	0x0A	0x01

5.3.6 清空所有指纹模板 Clear All Template

- [功能]
 - 删除已登记的所有指纹模板数据。
- [工作 Sequence]
 - 删除已登记的所有模板数据并返回其结果。
- [命令及响应 Command and Response]

表5-8 Clear All Template指令

PREFIX	0xAA55	
CMD	0x0106	
LEN	0	
DATA	数据不存在	
PREFIX	0x55AA	
RCM	0x0106	
LEN	4	
RET	成功: ERR_SUCCESS, 失败: ERR_FAIL	
DATA	2byte	成功: 已删除的 Template 个数, 失败: 错误码

- [指令实例]

命令包:

清空所有指纹

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x06	0x01	0x00	0x00	0x00	0x00	0x00	0x06	0x01

响应包:

清空所有指纹成功

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x06	0x01	0x04	0x00	0x00	0x00	0x01	0x00	...	0x00	0x0B	0x01

5.3.7 获取可注册的指纹模板编号 Get Empty ID

- [功能]
 - 获取可登记的指纹模板编号。
- [工作过程 Sequence]
 - 查找没有登记模板的第一个ID号。若存在，则返回该值。否则，返回 ERR_EMPTY_ID_NOEXIST。
- [命令及响应 Command and Response]

表5-9 Get Empty ID指令

PREFIX	0xAA55		
CMD	0x0107		
LEN	0		
DATA	数据不存在		
PREFIX	0x55AA		
RCM	0x0107		
LEN	4		
RET	成功: ERR_SUCCESS, 失败: ERR_FAIL		
DATA	2byte	成功: 可登记的第一个 Template 号码, 失败: 错误码	

● [指令实例]

命令包:

获取可注册指纹模板编号

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x07	0x01	0x00	0x00	0x00	0x00	0x00	0x07	0x01

响应包:

获取可注册指纹模板编号成功

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x07	0x01	0x04	0x00	0x00	0x00	0x02	0x00	...	0x00	0x0D	0x01

5.3.8 获取指定模板编号的状态 Get Template Status

- [功能]
 - 返回指定号码中的指纹登记状态。
- [工作 Sequence]
 - 若指定号码中已登记有Template，则返回GD_TEMPLATE_NOT_EMPTY。
否则，返回GD_TEMPLATE_EMPTY。
- [命令及响应 Command and Response]

表5-10 Get Template Status指令

PREFIX	0xAA55		
CMD	0x0108		
LEN	2		
DATA	Template 号码		
PREFIX	0x55AA		
RCM	0x0108		
LEN	4		
RET	成功: ERR_SUCCESS, 失败: ERR_FAIL		
DATA	2byte	成功: 登记状态值, 失败: 错误码	

● [指令实例]

命令包:

获取第1枚系统模板的状态

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x08	0x01	0x02	0x00	0x01	0x00	0x00	0x0B	0x01

响应包:

获取第1枚系统模板的状态成功

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x08	0x01	0x04	0x00	0x00	0x00	0x01	0x00	...	0x00	0x0D	0x01

5.3.9 获取指纹模板数据库的完整性 Get Broken Template

- [功能]
 - 检查已登记的所有指纹模板数据是否有损坏的情况。
 - 在写入Flash Memory过程中，因意外断电等的原因有可能损坏指纹模板数据库。Host在任意时刻发出该指令，可以检测指纹模板数据的破损情况。对于已损坏的指纹模板数据，可以删除再登记。
- [工作 Sequence]
 - 检查所有Template的损坏情况。若有已损坏的Template存在，则返回已损坏模板Template的个数及第一个已损坏的模板编号（Template号码）。否则返回的模板个数及第一个已损坏的模板编号都为“0”。
- [Command 及 Response]

表5-11 Get Broke Template指令

PREFIX	0xAA55		
CMD	0x0109		
LEN	0		
DATA	数据不存在		
PREFIX	0x55AA		
RCM	0x0109		
LEN	6		
RET	成功: ERR_SUCCESS, 失败: ERR_FAIL		
DATA	D0, D1	成功: Template 个数 失败: 错误码	
	D2, D3	成功: 第一个已损坏的 Template 号码 失败: 0	

- [指令实例]

命令包:

获取指纹模板数据库的完整性

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x09	0x01	0x00	0x00	0x00	0x00	0x00	0x09	0x01

响应包:

获取指纹模板数据库的完整性成功

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x09	0x01	0x06	0x00	0x00	0x00	0x00	0x00	...	0x00	0x0F	0x01

5.3.10 读取指定的指纹模板数据 Read Template

- [功能]
 - 读出指定编号中的指纹模板数据Template Record Data并上传至Host。
- [工作 Sequence]
 - ①、若指定的指纹模板编号无效，则返回 ERR_INVALID_TMPL_NO 。
 - ②、若指定的指纹模板编号中不存在模板数据 ，则返回 ERR_EMPTY_TMPL 。
 - ③、利用数据响应包（Data response packet）将 Host 要接收的数据（模板编号 Template No + 模板数据 Template Record Data）的大小作为返回数据传输。
 - ④、利用响应数据包 Response data packet 传输指定号码中的模板数据 Template Data。
- [命令及响应 Command and Response]

表5-12 Read Template指令

PREFIX	0xAA55
CMD	0x010A
LEN	2
DATA	Template 号码
PREFIX	0x55AA
RCM	0x010A
LEN	4
RET	ERR_SUCCESS or ERR_FAIL
DATA	成功：下次数据响应包Data response packet 的数据大小 (Template Record Size + 2) 失败：错误码
成功时	
PREFIX	0x5AA5
RCM	0x010A
LEN	模板数据大小Template Record Size + 4
RET	ERR_SUCCESS
DATA	模板编号Template No(2byte) + 模板数据Template Record Data

- [指令实例]

命令包:

读取第1枚指纹模板数据

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x0A	0x01	0x02	0x00	0x01	0x00	0x00	0x0D	0x01

响应包及响应数据包:

返回接收数据的大小、第1枚指纹模板数据

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x0A	0x01	0x04	0x00	0x00	0x00	0xF4	0x01	...	0x00	0x03	0x02
0xA5	0x5A	0x0A	0x01	0xF6	0x01	0x00	0x00	0x01	0x00	498字节数据		CKS	

5.3.11 写入指纹模板数据至指定的模板编号中 Write Template

- [功能]

- 从Host接收指纹模板数据 (Template Data) 并将其写入到指定号码中。
- 不管指定号码中是否已存在Template, 新的指纹模板数据都会被写入进去。

- [工作 Sequence]

Host 发送命令包使得模块进入指纹模板数据接收状态。在该命令包的数据域 (DATA field) 中已设定有下次发送的指令数据包的长度。

- ①、模块检测收到的该命令包 (Command packet) 的正确性。
 - 若不正确, 则返回错误码并结束处理, 若指纹模板数据大小 (Template Record Size) 不正确, 则返回 ERR_INVALID_PARAM。
 - 若正确, 则本模块进入指纹模板数据接收等待状态, 向 Host 发送响应包, 并进入指纹模板数据接收等待状态。
- ②、若 Host 收到模块已进入指纹模板 Template Record 数据接收状态的响应包, 则在指令数据包中设置指纹模板编号和指纹模板数据并发送至模块。
- ③、模块收到指令数据包后, 若指定的指纹模板编号号码无效, 则返回 ERR_INVALID_TMPL_NO。检测已收到的指纹模板数据的校验和 (Checksum of TemplateRecord)。若不正确, 则返回 ERR_INVALID_PARAM。
- ④、将已收到的指纹模板数据 (Template Data) 写入至指定号码中。

注:

建议先用“获取指定模板编号的状态 Get Template Status”命令检查指定的号码中是否已存在指纹数据, 以防止复盖已有的指纹模板数据。

- [命令及响应 Command Response]

表5-13 Write Template指令

命令包 Command Packet		
PREFIX		0xAA55
CMD		0x010B
LEN		2
DATA		指纹模板大小Template Record Size
PREFIX		0x55AA
RCM		0x010B
LEN		4
RET		成功: ERR_SUCCESS, 失败: ERR_FAIL
DATA	2byte	成功: 0 失败: 错误码 (ERR_INVALID_PARAM)
指令数据包 Data Packet		
PREFIX		0xA55A
CMD		0x010B
LEN		指纹模板大小 Template Record Size + 2
DATA		指纹模板编号Template No(2byte) + Template Record Data
PREFIX		0x5AA5
RCM		0x010B
LEN		4
RET		成功: ERR_SUCCESS, 失败: ERR_FAIL
DATA	2byte	成功: 指纹模板编号 失败: 错误码

- [指令实例]

命令包:

给定指纹模板数据大小

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x0B	0x01	0x02	0x00	0xF2	0x01	0x00	0x00	0x02

响应包及响应数据包:

通知HOST模块已进入指纹模板数据接收状态

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x0B	0x01	0x04	0x00	0x00	0x00	0x00	0x00	...	0x00	0x0F	0x01

指令数据包:

指纹特征数据写入编号为1的位置

PREFIX		CMD		LEN		DATA				CKS	
0x5A	0xA5	0x0B	0x01	0xF4	0x01	0x01	0x00	498字节数据		CKS	

响应数据包:

写指纹特征数据成功

PREFIX		RCM		LEN		RET		DATA		CKS	
0xA5	0x5A	0x0B	0x01	0x04	0x00	0x00	0x00	0x01	0x00	0x10	0x01

5.3.12 设置识别安全等级命令 Set Security Level

- [功能]
 - 设置安全等级Security Level。可设的有效范围为 1 至 5。
- [工作过程 Sequence]
 - ①、若指定 Security Level 无效，则返回 ERR_INVALID_SEC_VAL。
 - ②、设置指定的 Security Level 并返回其结果。
- [命令及响应 Command and Response]

表5-14 Set Security Level指令

PREFIX	0xAA55
CMD	0x010C
LEN	2
DATA	Security Level 安全等级的值
PREFIX	0x55AA
RCM	0x010C
LEN	4
RET	成功: ERR_SUCCESS, 失败: ERR_FAIL
DATA	成功: 已设定的 Security Level 值, 失败: 错误码

- [各安全等级对应的识别精确度]

表5-15 安全等级对应的识别精度

安全等级 Security Level	识别精确度	
Level 1	认假率 FAR (False Acceptance Rate)	0.01%
	拒识率 FRR (False Rejection Rate)	0.005%
Level 2	认假率 FAR (False Acceptance Rate)	0.003%
	拒识率 FRR (False Rejection Rate)	0.01%
Level 3	认假率 FAR (False Acceptance Rate)	0.001 %
	拒识率 FRR (False Rejection Rate)	0.1 %
Level 4	认假率 FAR (False Acceptance Rate)	0.0003%
	拒识率 FRR (False Rejection Rate)	0.5%
Level 5	认假率 FAR (False Acceptance Rate)	0.0001%
	拒识率 FRR (False Rejection Rate)	1%

注:

安全等级Security Level出厂设置值: 3

- [指令实例]

命令包:

设置识别安全等级为3

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x0C	0x01	0x02	0x00	0x03	0x00	0x00	0x11	0x01

响应包:

设置识别安全等级3成功

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x0C	0x01	0x04	0x00	0x00	0x00	0x03	0x00	...	0x00	0x13	0x01

5.3.13 获取上次设置的安全等级 Get Security Level

- [功能]
 - 返回目前已设定的安全等级Security Level值。
- [工作 Sequence]
 - 读出目前已设定的Security Level值并返回其结果。
- [命令及响应 Command and Response]

表5-16 Get Security Level指令

PREFIX	0xAA55
CMD	0x010D
LEN	0
DATA	数据不存在
PREFIX	0x55AA
RCM	0x010D
LEN	4
RET	成功: ERR_SUCCESS
DATA	目前已设定的 Security Level 值

- [指令实例]

命令包:

获取安全等级

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x0D	0x01	0x00	0x00	0x00	0x00	0x00	0x0D	0x01

响应包:

获取安全等级成功

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x0D	0x01	0x04	0x00	0x00	0x00	0x03	0x00	...	0x00	0x14	0x01

5.3.14 设置指纹采集超时参数 Set Finger Time Out

- [功能]

- 处理Verify、Identify、Enroll、Enroll One Time、Enroll And Store in RAM、Get Feature Data of Captured FP、Verify Downloaded Feature with Captured FP、Identify Downloaded Feature with Captured FP、Identify Free等指令时，设置等待采集指纹的超时限制时间。
- 可设的有效范围为 0 至 60 秒,默认值为 5 秒。

- [工作过程 Sequence]

- ①、若指定 Time Out 值无效，则返回 ERR_INVALID_TIME_OUT。
- ②、设置指定的 Time Out 值并返回其结果。

- [Command 及 Response]

表5-17 Set Finger Time Out指令

PREFIX	0xAA55
CMD	0x010E
LEN	2
DATA	Time Out 值
PREFIX	0x55AA
RCM	0x010E
LEN	4
RET	成功: ERR_SUCCESS, 失败: ERR_FAIL
DATA	成功: 已设定的 Time Out 值, 失败: 错误码

- [指令实例]

命令包:

设置指纹采集超时参数10秒

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x0E	0x01	0x02	0x00	0x0A	0x00	0x00	0x1A	0x01

响应包:

设置指纹采集超时参数10秒成功

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x0E	0x01	0x04	0x00	0x00	0x00	0x0A	0x00	...	0x00	0x1C	0x01

5.3.15 获取模块当前的超时参数值 Get Finger Time Out

- [功能]
 - 返回目前已设定的Time Out值。
- [工作 Sequence]
 - 读出目前已设定的Time Out值并返回其结果。
- [Command 及 Response]

表5-18 Get Finger Time Out指令

PREFIX	0xAA55
CMD	0x010F
LEN	0
DATA	数据不存在
PREFIX	0x55AA
RCM	0x010F
LEN	4
RET	成功: ERR_SUCCESS
DATA	目前已设定的 Time Out 值

● [指令实例]

命令包:

获取当前超时参数值

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x0F	0x01	0x00	0x00	0x00	0x00	0x00	0x0F	0x01

响应包:

获取当前超时参数成功

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x0F	0x01	0x04	0x00	0x00	0x00	0x05	0x00	...	0x00	0x18	0x01

5.3.16 设置设备编号 Set Device ID

- [功能]
 - 设置本设备编号 (Device ID)。设备编号为 1 至 254 的1个字节数字。
- [工作 Sequence]
 - ①、设置指定的 Device ID 并返回其结果。
 - ②、若超出其设置范围，则返回 ERR_INVALID_PARAM。
- [Command 及 Response]

表5-19 Set Device ID指令

PREFIX	0xAA55
CMD	0x0110
LEN	2
DATA	Device ID
PREFIX	0x55AA
RCM	0x0110
LEN	4
RET	成功: ERR_SUCCESS, 失败: ERR_FAIL
DATA	成功: 已设置的设备编号Device ID 失败: 2byte错误码

- [指令实例]

命令包:

设置设备编号为1

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x10	0x01	0x02	0x00	0x01	0x00	0x00	0x13	0x01

响应包:

设置设备编号1成功

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x10	0x01	0x04	0x00	0x00	0x00	0x01	0x00	...	0x00	0x15	0x01

5.3.17 获取设备编号 Get Device ID

- [功能]
 - 返回模块中已设定的设备编号 (Device ID)。设备编号为 1 至 254 的数字。
 - 出厂时设置为：0
- [工作 Sequence]
 - 读出Device ID并返回其结果。
- [Command 及 Response]

表5-20 Get Device ID指令

PREFIX	0xAA55
CMD	0x0111
LEN	0
DATA	数据不存在
PREFIX	0x55AA
RCM	0x0111
LEN	4
RET	成功：ERR_SUCCESS，失败：ERR_FAIL
DATA	成功：目前已设定的 Device ID 失败：错误码

● [指令实例]

命令包：

获取设备编号

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x11	0x01	0x00	0x00	0x00	0x00	0x00	0x11	0x01

响应包：

获取设备编号成功

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x11	0x01	0x04	0x00	0x00	0x00	0x01	0x00	...	0x00	0x16	0x01

5.3.18 获取固件版本号 Get F/W Version

- [功能]
 - 返回本设备的固件（Firmware）的版本。
- [工作 Sequence]
 - 返回Firmware的版本信息。
- [命令及响应Command and Response]

表5-21 Get F/W Version指令

PREFIX	0xAA55	
CMD	0x0112	
LEN	0	
DATA	数据不存在	
PREFIX	0x55AA	
RCM	0x0112	
LEN	4	
RET	成功: ERR_SUCCESS	
DATA	DO	主版本号 Version Major
	D1	辅助版本号 Version Minor

- [指令实例]

命令包:

获取固件版本号

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x12	0x01	0x00	0x00	0x00	0x00	0x00	0x12	0x01

响应包:

获取固件版本号为2.9成功

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x12	0x01	0x04	0x00	0x00	0x00	0x02	0x09	...	0x00	0x21	0x01

5.3.19 检测是否有指纹按下指令 Finger Detect

- [功能]
 - 检测接收到该指令时刻是否有指纹按下并返回其结果。
- [工作 Sequence]
 - 返回接收相应指令时刻指纹输入状态。
- [Command 及 Response]

表5-22 Finger Detect指令

PREFIX	0xAA55
CMD	0x0113
LEN	0
DATA	数据不存在
PREFIX	0x55AA
RCM	0x0113
LEN	4
RET	成功: ERR_SUCCESS
DATA	指纹输入状态: 1: 有指纹, 0: 无指纹

注:

- [指令实例]

命令包:

检测是否有指纹按下

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x13	0x01	0x00	0x00	0x00	0x00	0x00	0x13	0x01

响应包:

检测结果: 有指纹按下

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x13	0x01	0x04	0x00	0x00	0x00	0x01	0x00	...	0x00	0x18	0x01

5.3.20 设置通讯波特率指令 Set Baudrate

- [功能]
 - 设置本模块通讯的波特率 (Baudrate) 值。
 - 设该波特率值后，在模块未重启之前无效。只有Reset或Power On后才有效。
- [工作 Sequence]
 - ①、若指定的 Baudrate 值无效，则返回错误码 ERR_INVALID_BAUDRATE。
 - ②、设置指定的 Baudrate 值并返回其结果。
- [Command 及 Response]

表5-23 Set Baudrate指令

PREFIX	0xAA55
CMD	0x0114
LEN	2
DATA	Baudrate Index 1 : 9600bps 2 : 19200bps 3 : 38400bps 4 : 57600bps 5 : 115200bps
PREFIX	0x55AA
RCM	0x0114
LEN	4
RET	成功: ERR_SUCCESS, 失败: ERR_FAIL
DATA	成功: 已设定的 Baudrate Index , 失败: 错误码

● [指令实例]

命令包:

设置通讯波特率为9600

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x14	0x01	0x02	0x00	0x01	0x00	0x00	0x17	0x01

响应包:

设置通讯波特率9600成功

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x14	0x01	0x04	0x00	0x00	0x00	0x01	0x00	...	0x00	0x19	0x01

5.3.21 设置指纹重复检查的允许/禁止 Set Duplication Check

- [功能]
 - 检查在指纹登记指令过程 (Enroll、Enroll One Time) 中，指纹数据是否有重复。
 - 若选择了允许重复检查选项，则处理Enroll、Enroll One Time指令时，当发现刚注册的指纹与指纹模板库中有相同的模板数据时返回相应模板编号及错误码ERR_DUPLICATION_ID。
- [工作 Sequence]
 - ①、若指定指纹重复检查 Duplication Check Option 选项无效，则返回 ERR_INVALID_DUP_VAL。
 - ②、设置指定的指纹重复检查选项并返回其结果。
- [Command 及 Response]

表5-24 Set Duplication Check指令

PREFIX	0xAA55
CMD	0x0115
LEN	2
DATA	指纹重复检查选项 (1为需要检查, 0为不检查)
PREFIX	0x55AA
RCM	0x0115
LEN	4
RET	成功: ERR_SUCCESS, 失败: ERR_FAIL
DATA	成功: 已设定的检查选项, 失败: 错误码

● [指令实例]

命令包:

设置指纹重复检查允许

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x15	0x01	0x02	0x00	0x01	0x00	0x00	0x18	0x01

响应包:

设置指纹重复检查允许成功

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x15	0x01	0x04	0x00	0x00	0x00	0x01	0x00	...	0x00	0x1A	0x01

5.3.22 获取模块当前的指纹重复检查的状态 Get Duplication Check

- [功能]
 - 返回模块中当前设置的指纹重复检查的选项（On/Off）。
- [工作 Sequence]
 - 读出目前已设定的Duplication检查选项并返回其结果。
- [Command 及 Response]

表5-25 Get Duplication Check指令

PREFIX	0xAA55
CMD	0x0116
LEN	0
DATA	数据不存在
PREFIX	0x55AA
RCM	0x0116
LEN	4
RET	成功: ERR_SUCCESS
DATA	已设定的检查选项 1: 为需要检查指纹重复, 0: 为不检查

- [指令实例]

命令包:

获取模块当前的指纹重复检查状态

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x16	0x01	0x00	0x00	0x00	0x00	0x00	0x16	0x01

响应包:

获取模块当前的指纹重复检查状态为允许

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x16	0x01	0x04	0x00	0x00	0x00	0x01	0x00	...	0x00	0x1B	0x01

5.3.23 控制模块进入待机状态 Enter Standby Mode

- [功能]
 - 使模块Target进入休眠模式Standby Mode。
 - 进入待机状态的工作电流大约5mA。
- [工作 Sequence]
 - 使模块Target进入Standby Mode（省电状态）并返回ERR_SUCCESS。
 - 若要从休眠Standby Mode状态唤醒，则必须重启（Reset）模块，或Power On。
- [Command 及 Response]

表5-26 Enter Standby Mode指令

PREFIX	0xAA55
CMD	0x0117
LEN	0
DATA	数据不存在
PREFIX	0x55AA
RCM	0x0117
LEN	2
RET	ERR_SUCCESS
DATA	-

- **注：关断模块电源前请先发送本指令（Enter Standby Mode），使本模块进入待机状态**

- [指令实例]

命令包：

设置模块进入待机状态

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x17	0x01	0x00	0x00	0x00	0x00	0x00	0x17	0x01

响应包：

设置模块进入待机状态成功

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x17	0x01	0x04	0x00	0x00	0x00	0x00	0x00	...	0x00	0x1B	0x01

5.3.24 注册指纹并暂存在RAM中 Enroll And Store in RAM

- [功能]

- 利用指纹登记过程中获取的三个临时性模板融合生成一个指纹模板数据并暂存至RAM中。
- 在三次输入过程中，若临时模板Template不正确，则重新进行本当次指纹输入。
- 若合成过程中出错，则重新进行三次指纹输入。
- 注册在RAM中的指纹特征数据可以用Get Enroll Data指令上传至Host

- [工作过程 Sequence]

- ①、返回 GD_NEED_FIRST_SWEEP，表示第一次指纹输入的请求，且在 Time Out 时间内检查按手指情况。
- ②、若在 Time Out 时间内检测不到指纹的输入，则返回 ERR_TIME_OUT。
- ③、检查被输入图像的正确性。若不正确，则返回相应错误码（ERR_BAD_QUALITY 等）。
- ④、若第一次输入指纹正确，则返回 GD_NEED_RELEASE_FINGER 告知离开手指，并监测是否已离开。若已离开，则跳到⑤。
- ⑤、返回 GD_NEED_SECOND_SWEEP，表示第二次指纹输入的请求，且重复②、③。
- ⑥、若第二次输入指纹正确，则返回 GD_NEED_RELEASE_FINGER 告知离开手指，并监测是否已离开。若已离开，则跳到⑦。
- ⑦、若第二次指纹输入成功，则返回 GD_NEED_THIRD_SWEEP，表示第三次指纹输入的请求并重复②、③。
- ⑧、若第三次输入指纹正确，则返回 GD_NEED_RELEASE_FINGER 告知离开手指。
- ⑨、利用已得到的三个模板 Template 融合生成一个可登记的模板 Template 数据。若失败，则返回 ERR_GENERALIZE。
- ⑩、将已生成的 Template Data 保存至 RAM 并返回 0。
- ⑪、若收到 FpCancel 指令，则终止该指令并返回 ERR_FP_CANCEL。

- [Command 及 Response]

表5-27 Enroll And Store In RAM指令

PREFIX	0xAA55
CMD	0x0118
LEN	0
DATA	数据不存在
PREFIX	0x55AA
RCM	0x0118
LEN	4
RET	成功: ERR_SUCCESS, 失败: ERR_FAIL
DATA	成功: GD_NEED_FIRST_SWEEP、GD_NEED_SECOND_SWEEP、 GD_NEED_THIRD_SWEEP、GD_NEED_RELEASE_FINGER、0 失败: 错误码

- [指令实例]

命令包:

注册指纹并暂存在RAM

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x18	0x01	0x00	0x00	0x00	0x00	0x00	0x18	0x01

响应包:

注册指纹并暂存在RAM成功

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x18	0x01	0x04	0x00	0x00	0x00	0xF1	0xFF	...	0x00	0x0C	0x03
0xAA	0x55	0x18	0x01	0x04	0x00	0x00	0x00	0xF4	0xFF	...	0x00	0x0F	0x03
0xAA	0x55	0x18	0x01	0x04	0x00	0x00	0x00	0xF2	0xFF	...	0x00	0x0D	0x03
0xAA	0x55	0x18	0x01	0x04	0x00	0x00	0x00	0xF4	0xFF	...	0x00	0x0F	0x03
0xAA	0x55	0x18	0x01	0x04	0x00	0x00	0x00	0xF3	0xFF	...	0x00	0x0E	0x03
0xAA	0x55	0x18	0x01	0x04	0x00	0x00	0x00	0xF4	0xFF	...	0x00	0x0F	0x03
0xAA	0x55	0x18	0x01	0x04	0x00	0x00	0x00	0x00	0x00	...	0x00	0x1C	0x01

5.3.25 获取注册指纹的模板数据 Get Enroll Data

- [功能]
 - 读出利用Enroll And Store in RAM指令保存至RAM中的模板数据并上传至Host。
 - 在处理该指令之前，Host必须先发Enroll And Store in RAM指令。
- [工作过程 Sequence]
 - 将保存在RAM 中的指纹模板数据利用响应包响应数据包Response data packet发送至Host。（请参考Read Template指令）
- [Command 及 Response]

表5-28 Get Enroll Data指令

PREFIX	0xAA55
CMD	0x0119
LEN	0
DATA	数据不存在
PREFIX	0x55AA
RCM	0x0119
LEN	4
RET	ERR_SUCCESS or ERR_FAIL
DATA	成功：下次数据响应包Data Response Packet的数据大小 (Template Record Size) ， 失败：错误码
成功时	
PREFIX	0x5AA5
RCM	0x0119
LEN	指纹模板大小Template Record Size + 2
RET	ERR_SUCCESS
DATA	Template Record Data

注：

必须先执行Enroll And Store in RAM指令。

- [指令实例]

命令包:

获取注册指纹模块数据

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x19	0x01	0x00	0x00	0x00	0x00	0x00	0x19	0x01

响应包及响应数据包:

返回接收数据的大小、暂存在RAM中指纹模板数据

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x19	0x01	0x04	0x00	0x00	0x00	0xF2	0x01	...	0x00	0x10	0x02
0xA5	0x5A	0x19	0x01	0xF4	0x01	0x00	0x00	498字节数据				CKS	

5.3.26 采集图像并提取指纹特征值数据上传 Get Feature Data of Captured FP

● [功能]

➤ 提取进行一次指纹输入生成的指纹模板数据Template Record Data送至Host。

● [工作 Sequence]

- ①、若在 Time Out 时间内检测不到指纹的输入，则返回 ERR_TIME_OUT。
- ②、检查指纹图像的正确性。若不正确，则返回相应错误码（ERR_BAD_QUALITY 等）。
- ③、将已生成的指纹模板数据利用响应数据包（Response data packet）发送。
- ④、若收到 FpCancel 指令，则终止该指令并返回 ERR_FP_CANCEL。

● [Command 及 Response]

表5-29 Get Feature Data of Captured FP指令

PREFIX	0xAA55
CMD	0x011A
LEN	0
DATA	数据不存在
PREFIX	0x55AA
RCM	0x011A
LEN	4
RET	ERROR_SUCCESS or ERR_FAIL
DATA	成功：下次数据响应包Data Response Packet的数据大小 (Template Record Size)， 失败：错误码
成功时	
PREFIX	0x5AA5
RCM	0x011A
LEN	指纹模板大小Template Record Size + 2
RET	ERR_SUCCESS
DATA	Template Record Data

- [指令实例]

命令包:

采集图像并提取指纹特征值数据

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	01A	0x01	0x00	0x00	0x00	0x00	0x00	0x1A	0x01

响应包及响应数据包:

返回接收数据的大小、指纹模板数据

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x1A	0x01	0x04	0x00	0x00	0x00	0xF2	0x01	...	0x00	0x11	0x02
0xA5	0x5A	0x1A	0x01	0xF4	0x01	0x00	0x00	498字节数据				CKS	

5.3.27 下载特征数据与采集的指纹比对Verify Downloaded Feature With Captured FP

- [功能]

- 从Host下传的指纹模板数据Template Data与当前采集生成的指纹模板数据Template Data之间进行 1:1 比对。

- [工作过程 Sequence]

- ①、Host 发送命令包 Command packet 使得模块 Target 进入指纹模板数据 Template Record 数据接收等待状态。在该命令包的 DATA field 中，已设定有下次发送的指令数据包的长度。
- ②、Target 为了通知自己进入 Template Record 数据接收等待状态，向 Host 发送 Response Packet 并进入 Template Record 数据接收等待状态。
- ③、若 Host 收到模块 Target 进入指纹模板 Template Record 数据接收等待状态的响应包 Response Packet，则在指令数据包中设置指纹模板数据，并发送至模块。
- ④、模块 Target 收到指令数据包后进入指纹输入等待状态。
- ⑤、若在 Time Out 时间内检测不到指纹的输入，则返回 ERR_TIME_OUT。
- ⑥、检查被输入图像的正确性。若不正确，则返回相应错误码（ERR_BAD_QUALITY 等）。
- ⑦、生成的 Template Data 与收到的 Template Data 之间进行 1:1 比对并返回其结果。
- ⑧、若收到 FpCancel 指令，则终止该指令并返回 ERR_FP_CANCEL。

- [命令及响应 Command and Response]

表5-30 Verify Downloaded Feature With Captured FP指令

命令包 Command Packet	
PREFIX	0xAA55
CMD	0x011B
LEN	2
DATA	Template Record Size
PREFIX	0x55AA
RCM	0x011B
LEN	4
RET	成功: ERR_SUCCESS, 失败: ERR_FAIL
DATA	成功时: 0, 失败时: 错误码
命令数据包 Command Data Packet	
PREFIX	0xA55A
CMD	0x011B
LEN	Template Record Size
DATA	Template Record Data
PREFIX	0x5AA5
RCM	0x011B
LEN	4
RET	成功: ERR_SUCCESS, 失败: ERR_FAIL
DATA	成功时: 0, 失败时: 错误码

- [指令实例]

命令包:

指纹模板数据大小

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x1B	0x01	0x02	0x00	0xF2	0x01	0x00	0x10	0x02

响应包及响应数据包:

检查数据大小正确

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x1B	0x01	0x04	0x00	0x00	0x00	0x00	0x00	...	0x00	0x1F	0x01

指令数据包:

下载指纹特征数据

PREFIX		CMD		LEN		DATA				CKS	
0x5A	0xA5	0x1B	0x01	0xF2	0x01	498字节数据				CKS	

响应数据包:

下载指纹特征数据与当前指纹比对成功

PREFIX		RCM		LEN		RET		DATA		CKS	
0xA5	0x5A	0x1B	0x01	0x04	0x00	0x00	0x00	0x00	0x00	0x1F	0x01

5.3.28 下载2枚模板与采集的指纹比对 Identify Downloaded Feature With Captured FP

- [功能]

- 指纹模块从Host收到的两个指纹模板数据 Template Data 与当前采集生成的指纹模板 Template Data 之间进行 1:N 比对。

- [工作过程 Sequence]

- ①、Host 发送 Command packet 使得 Target 进入 Template Record 数据接收等待状态。在该命令包的 DATA field 中，已设定有下次发送的第一个指令数据的长度。
- ②、指纹模块 Target 为了使自己进入 Template Record 数据接收等待状态，向 Host 发送 Response Packet 并进入第一个指纹模板 Template Record 数据接收等待状态。
- ③、若 Host 收到指纹模块 Target 进入指纹模板 Template Record 数据接收等待状态的 响应包 Response Packet，则在指令数据 packet 中设置 1（表示第一个 Template Data）和第一个 Template Record 数据并发送至 Target。
- ④、指纹模块 Target 收到第一个指令数据包后发送表示接收状态的响应包 Response packet（Response Code: GD_DOWNLOAD_SUCCESS），并进入第二个 Template Record 数据等待状态。
- ⑤、若 Host 收到指纹模块 Target 已进入第二个 Template Record 数据接收等待状态的 响应包 Response Packet，则在指令数据包中设置 2（表示第二个 Template Data）和第二个 Template Record 数据并发送至 Target。
- ⑥、指纹模块 Target 收到第二个指令数据包后进入指纹输入等待状态。
- ⑦、若在 Time Out 时间内检测不到指纹的输入，则返回 ERR_TIME_OUT。
- ⑧、检查被输入图像的正确性。若不正确，则返回相应错误码（ERR_BAD_QUALITY 等）。
- ⑨、生成 Template Data 与收到的两个 Template Data 之间进行 1:N 比对并返回其结果。
- ⑩、若收到 FpCancel 指令，则终止该指令并返回 ERR_FP_CANCEL。

- [Command 及 Response]

表5-31 Identify Downloaded Feature with Captured FP指令

命令包 Command packet	
PREFIX	0xAA55
CMD	0x011C
LEN	2
DATA	Template Record Size
数据包 Data packet	
PREFIX	0xA55A
CMD	0x011C
LEN	指纹模板数据大小Template Record Data + 2
DATA	指纹模板索引号Template Index(2Byte) + 指纹板数据Template Record Data
PREFIX	0x55AA
RCM	0x011C`
LEN	4
RET	成功: ERR_SUCCESS, 失败: ERR_FAIL
DATA	成功时: 0, 失败时: 错误码

- [指令实例]

命令包:

指纹模板数据大小

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x1C	0x01	0x02	0x00	0xF2	0x01	0x00	0x11	0x02

响应包及响应数据包:

检查数据大小正确

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x1C	0x01	0x04	0x00	0x00	0x00	0x00	0x00	...	0x00	0x20	0x01

指令数据包:

下载第1枚指纹特征数据

PREFIX		CMD		LEN		DATA				CKS	
0x5A	0xA5	0x1C	0x01	0xF4	0x01	0x01	0x00	498字节数据			CKS

响应数据包:

下载第1枚指纹特征数据成功

PREFIX		RCM		LEN		RET		DATA		CKS	
0xA5	0x5A	0x1C	0x01	0x04	0x00	0x00	0x00	0xA1	0x00	0xC1	0x01

指令数据包:

下载第2枚指纹特征数据

PREFIX		CMD		LEN		DATA				CKS	
0x5A	0xA5	0x1C	0x01	0xF4	0x01	0x02	0x00	498字节数据			CKS

响应数据包:

下载2枚指纹特征与当前指纹比对成功

PREFIX		RCM		LEN		RET		DATA		CKS	
0xA5	0x5A	0x1C	0x01	0x04	0x00	0x00	0x00	0x00	0x00	0x20	0x01

5.3.29 获取设备名称 Get Device Name

- [功能]
 - 返回指纹模块Target的设备名称Device Name。本设备名固定为“ST020-OEM”。
- [工作 Sequence]
 - 返回设备名称Device Name。
- [Command 及 Response]

表5-32 Get Device Name指令

PREFIX	0xAA55
CMD	0x0121
LEN	0
DATA	数据不存在
PREFIX	0x55AA
RCM	0x0121
LEN	16
RET	成功: ERR_SUCCESS, 失败: ERR_FAIL
DATA	成功时: “ST020-OEM”, 失败时: 2byte错误码

● [指令实例]

命令包:

获取设备编号

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x21	0x01	0x00	0x00	0x00	0x00	0x00	0x21	0x01

响应包:

获取设备编号成功

PREFIX		RCM		LEN		RET		DATA	CKS
0xAA	0x55	0x21	0x01	0x10	0x00	0x00	0x00	ASCII码表示的设备名称	CKS

5.3.30 指纹采集器背光控制命令 Sensor LED Control

注：对本型号该指令无效

- [功能]
 - 打开或关闭采集器背光灯。
- [工作 Sequence]
 - 参数为1：则开灯。
 - 参数为0：则关灯。
- [Command 及 Response]

表5-33 Sensor LED Control指令

PREFIX	0xAA55
CMD	0x0124
LEN	2
DATA	LED状态（1：开，0：关）
PREFIX	0x55AA
RCM	0x0124
LEN	4
RET	成功：ERR_SUCCESS
DATA	0

● [指令实例]

命令包：

打开采集器背光

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x24	0x01	0x02	0x00	0x01	0x00	0x00	0x27	0x01

响应包：

打开采集器背光成功

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x24	0x01	0x04	0x00	0x00	0x00	0x00	0x00	...	0x00	0x28	0x01

5.3.31 连续指纹识别 Identify Free

- [功能]

- 该指令使模块处于循环采集指纹并和已登记指纹之间进行1:N比对，并返回其结果。直到收到FpCancel指令为止并返回其结果。
- 与Identify指令的区别在于，不受Finger Timeout的时间限制，而循环执行“采集指纹→1:N比对”操作直到收到FpCancel指令为止。
- Finger TimeOut的时间内未采集到指纹，返回ERR_TIME_OUT，并自动重装载TimeOut定时器后继续扫描指纹输入情况。

- [工作 Sequence]

- ①、若已登记的 Template 不存在，则返回错误码 ERR_ALL_TMPL_EMPTY。
- ②、一直等待指纹的输入直到有指纹输入为止。
- ③、检查输入的图像的正确性。若不正确，则返回相应错误码（ERR_BAD_QUALITY 等）。
- ④、检查输入的图像的正确性。若正确，则返回 GD_NEED_RELEASE_FINGER。
- ⑤、已登记的所有 Template 与输入的图像之间进行比对并返回其结果。若比对成功，则返回 ERR_SUCCESS 和其对应的模板 Template 号码。否则，返回 ERR_FAIL 且 DATA 为 ERR_IDENTIFY 或 ERR_NO_RELEASE（同一手指一直接在传感器上未离开，且识别失败）
- ⑥、反复进行 2 至 5 步骤直到收到 FpCancel 指令为止。
- ⑦、若收到 FpCancel 指令，则终止该指令并返回 ERR_FP_CANCEL。

- [Command 及 Response]

表5-34 Identify Free指令

PREFIX	0xAA55	
CMD	0x0125	
LEN	0	
DATA	数据不存在	
PREFIX	0x55AA	
RCM	0x0125	
LEN	4	
RET	成功：ERR_SUCCESS，失败：ERR_FAIL	
DATA	2byte	成功：GD_NEED_RELEASE_FINGER / Template 号码， 失败：错误码

- [指令实例]

命令包:

连续指纹识别

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x25	0x01	0x00	0x00	0x00	0x00	0x00	0x25	0x01

响应包:

连续指纹识别成功

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x25	0x01	0x04	0x00	0x00	0x00	0xF4	0xFF	...	0x00	0x1C	0x03
0xAA	0x55	0x25	0x01	0x04	0x00	0x00	0x00	0x01	0x00	...	0x00	0x2A	0x01

5.3.32 获取已注册指纹总数 Get Enroll Count

- [功能]
 - 获取已登记在模块中的指纹数据总数。
- [工作 Sequence]
 - 返回已登记在模块中的指纹数据总数。
- [Command 及 Response]

表5-37 Get Enroll Count指令

PREFIX	0xAA55
CMD	0x0128
LEN	0
DATA	数据不存在
PREFIX	0x55AA
RCM	0x0128
LEN	4
RET	成功: ERR_SUCCESS, 失败: ERR_FAIL
DATA	成功: 当前已登记的 Template 总数, 失败: 错误码

● [指令实例]

命令包:

获取已注册指纹总数

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x28	0x01	0x00	0x00	0x00	0x00	0x00	0x28	0x01

响应包:

获取已注册指纹总数成功

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x28	0x01	0x04	0x00	0x00	0x00	0x01	0x00	...	0x00	0x2D	0x01

5.3.33 取消有关指纹采集的操作 FP Cancel

- [功能]

- 取消有关指纹采集的指令。
- 若在处理Verify、Identify、Enroll、Enroll One Time、Enroll And Store in RAM、Get Feature Data of Captured FP、Verify Downloaded Feature with Captured FP、Identify Downloaded Feature with Captured FP指令时收到FP Cancel指令，则在该时刻，中止当前处理的指令并进入初始状态。在以上的指令应答中返回ERR_FP_CANCEL，表示该指令已被取消操作，且对FP Cancel本身指令也有响应返回ERR_SUCCESS，表示以上操作指令取消成功。
- 对其它指令不影响操作。

- [工作 Sequence]

设置目前所处理指令的取消标记flag 后，返回 ERR_SUCCESS

- [Command 及 Response]

表5-38 FP Cancel指令

PREFIX	0xAA55
CMD	0x0130
LEN	0
DATA	数据不存在
PREFIX	0x55AA
RCM	0x0130
LEN	4
RET	ERR_SUCCESS
DATA	-

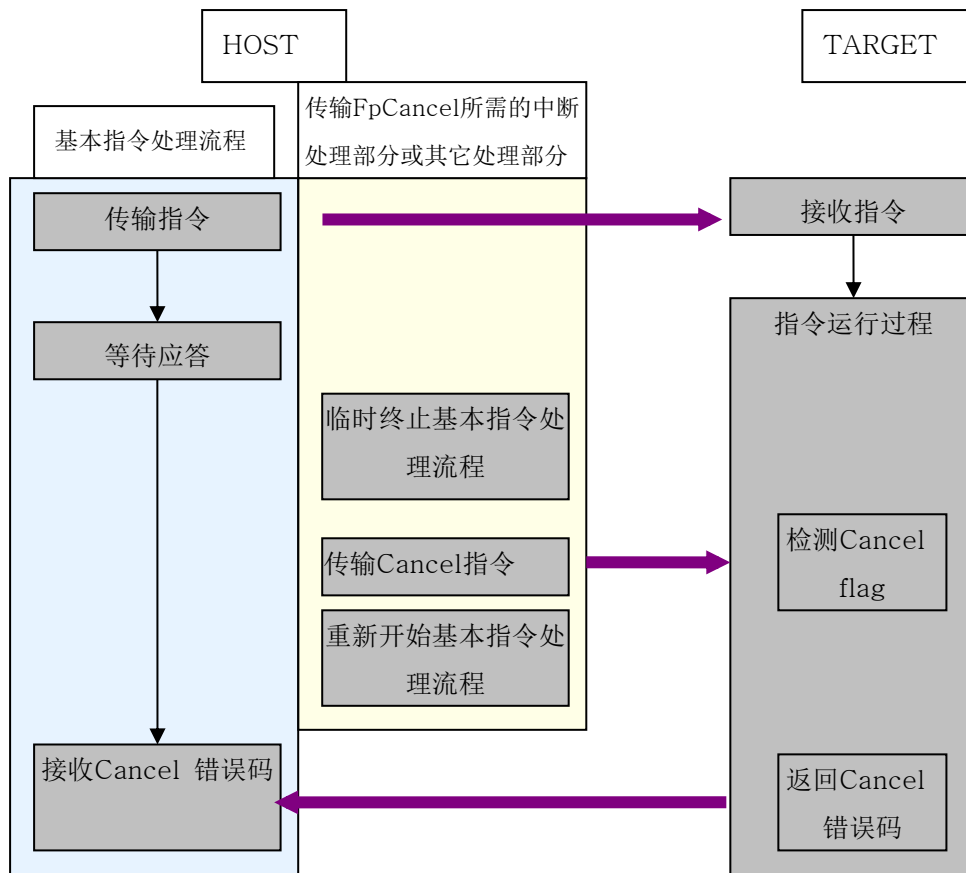


图5-1 FP Cancel过程

- [指令实例一]: 无操作时发送FP Cancel指令

命令包:

取消操作指令

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x30	0x01	0x00	0x00	0x00	0x00	0x00	0x30	0x01

响应包:

取消操作成功

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x30	0x01	0x04	0x00	0x00	0x00	0x00	0x00	...	0x00	0x34	0x01

- [指令实例二]: Verify、Identify、Enroll、Enroll One Time、Enroll And Store in RAM、Get Feature Data of Captured FP、Verify Downloaded Feature with Captured FP、Identify Downloaded Feature with Captured FP指令时发送FP Cancel指令

命令包: 取消操作指令 (Enroll过程)

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x30	0x01	0x00	0x00	0x00	0x00	0x00	0x30	0x01

响应包: 取消操作成功 (取消Enroll)

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x03	0x01	0x04	0x00	0x01	0x00	0x41	0x00	...	0x00	0x49	0x01
0xAA	0x55	0x30	0x01	0x04	0x00	0x00	0x00	0x00	0x00	...	0x00	0x34	0x01

5.3.34 连接状态检测命令 Test Connection

- [功能]
 - 检查Target与Host之间的连接状态。
 - Host在上电初始发送该指令以检查与Target的连接状态。
- [工作 Sequence]
 - 若连接无误, 则返回若连接无误, 则返回ERR_SUCCESS。
- [Command 及 Response]

表5-39 Test Connection指令

PREFIX	0xAA55
CMD	0x0150
LEN	0
DATA	数据不存在
PREFIX	0x55AA
RCM	0x0150
LEN	4
RET	ERR_SUCCESS or ERR_FAIL
DATA	-

- [指令实例]

命令包:

检测连接状态

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x50	0x01	0x00	0x00	0x00	0x00	0x00	0x50	0x01

响应包:

连接成功

PREFIX	RCM	LEN	RET	DATA	CKS

0xAA	0x55	0x50	0x01	0x04	0x00	0x00	0x00	0x00	0x00	...	0x00	0x54	0x01
------	------	------	------	------	------	------	------	------	------	-----	------	------	------

5.3.35 错误操作命令 Incorrect Command

- [功能]
 - 因通讯错误、误码等原因，当指纹模块收到不正确的指令时向Host返回该响应包。
- [Command 及 Response]

表5-40 Incorrect Command指令

PREFIX	0x55AA
RCM	0x0160
LEN	4
RET	ERR_SUCCESS
DATA	-

- [指令实例]

命令包:

发送一错误指令

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	0x01	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x01

响应包:

响应错误操作指令

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x60	0x01	0x04	0x00	0x00	0x00	0x00	0x00	...	0x00	0x64	0x01

5.4 注意事项

1. 在处理 Enroll, Enroll One Time, Identify, Verify, Enroll And Store in RAM, Get Feature Data of Captured FP, Verify Downloaded Feature with Captured FP, Identify Downloaded Feature with Captured FP指令时，模块会自动开启采集器背光灯，若采集到正确指纹，则自动关闭背光灯。若没采集到正确指纹，则采集器灯一直亮着。用户可也可根据自己需求开/关背光灯。可以根据自己需求利用 Sensor LED Control 指令控制开启或关闭采集器背光灯。

注：上述只针对光学采集器的需要的注意事项，对半导体传感器不需要执行背光控制

2. 本设备可以以口令保护模式来使用本模块。设备密码已设定后，如果不通过密码确认使模块进入通讯状态，则不能使用所有指令。因此不可能利用未经许可的设备来控制该模块。因此，尽管该模块被盗，但可以保护已登记的指纹数据并更安全使用该模块。若没设定设备密码，则不用通过密码确认也可以使用所有指令。

六、附录

6.1 响应（Response）及错误代码表

No	Response 及错误代码	值	说明
1	ERR_SUCCESS	0x00	指令处理成功。
2	ERR_FAIL	0x01	指令处理失败。
3	ERR_VERIFY	0x11	与指定号码中 Template 的 1:1 比对失败。
4	ERR_IDENTIFY	0x12	已进行 1:N 比对，但相同 Template 不存在。
5	ERR_TMPL_EMPTY	0x13	在指定号码中不存在已登记的 Template 。
6	ERR_TMPL_NOT_EMPTY	0x14	在指定号码中已存在 Template 。
7	ERR_ALL_TMPL_EMPTY	0x15	不存在已登记的 Template 。
8	ERR_EMPTY_ID_NOEXIST	0x16	不存在可登记的 Template ID 。
9	ERR_BROKEN_ID_NOEXIST	0x17	不存在已损坏的 Template 。
10	ERR_INVALID_TMPL_DATA	0x18	指定的 Template Data 无效。
11	ERR_DUPLICATION_ID	0x19	该指纹已登记。
12	ERR_BAD_QUALITY	0x21	指纹图像质量不好。
13	ERR_TIME_OUT	0x23	在 Time Out 时间内没检测到指纹的输入。
14	ERR_NOT_AUTHORIZED	0x24	没有进行设备密码确认。 若设备密码已被设定且没有利用 Verify Device Password 指令进行确认，则除了 Test Connection, Verify Device Password 指令之外的所有指令都返回该错误码。 若没有设定设备密码，则该设备不用通过密码确认也可以执行所有功能。
15	ERR_GENERALIZE	0x30	登记 Template 的制作失败。
16	ERR_FP_CANCEL	0x41	指令已被取消。
17	ERR_INTERNAL	0x50	软件内部错误。
18	ERR_MEMORY	0x51	软件内部错误。
19	ERR_EXCEPTION	0x52	软件内部错误。
20	ERR_INVALID_TMPL_NO	0x60	指定的 Template 号码无效。
21	ERR_INVALID_SEC_VAL	0x61	指定的 Security Level 值无效。
22	ERR_INVALID_TIME_OUT	0x62	指定的 Time Out 值无效。
23	ERR_INVALID_BAUDRATE	0x63	指定的 Baud Rate 值无效。

24	ERR_DEVICE_ID_EMPTY	0x64	没有设定 Device ID 。
25	ERR_INVALID_DUP_VAL	0x65	指定的 Duplication Check Option值无效。
26	ERR_INVALID_PARAM	0x70	使用了无效参数。
27	ERR_NO_RELEASE	0x71	在IdentifyFree指令执行过程中， 识别不成功的指纹未离开
28	GD_DOWNLOAD_SUCCESS	0xA1	已收到 Template Record 数据。
29	GD_NEED_FIRST_SWEEP	0xFFF 1	第一次指纹输入等待状态。
30	GD_NEED_SECOND_SWEEP	0xFFF 2	第二次指纹输入等待状态。
31	GD_NEED_THIRD_SWEEP	0xFFF 3	第三次指纹输入等待状态。
32	GD_NEED_RELEASE_FINGER	0xFFF 4	离开手指
33	GD_DETECT_FINGER	0x01	发出 FingerDetect 指令时刻，已检测到有指纹输入
34	GD_NO_DETECT_FINGER	0x00	发出 FingerDetect 指令时刻，没检测到指纹的输入
35	GD_TEMPLATE_NOT_EMPTY	0x01	Template 不为空。
36	GD_TEMPLATE_EMPTY	0x00	Template 为空。

6.2 登记及比对流程图

6.2.1 注册流程 Enroll

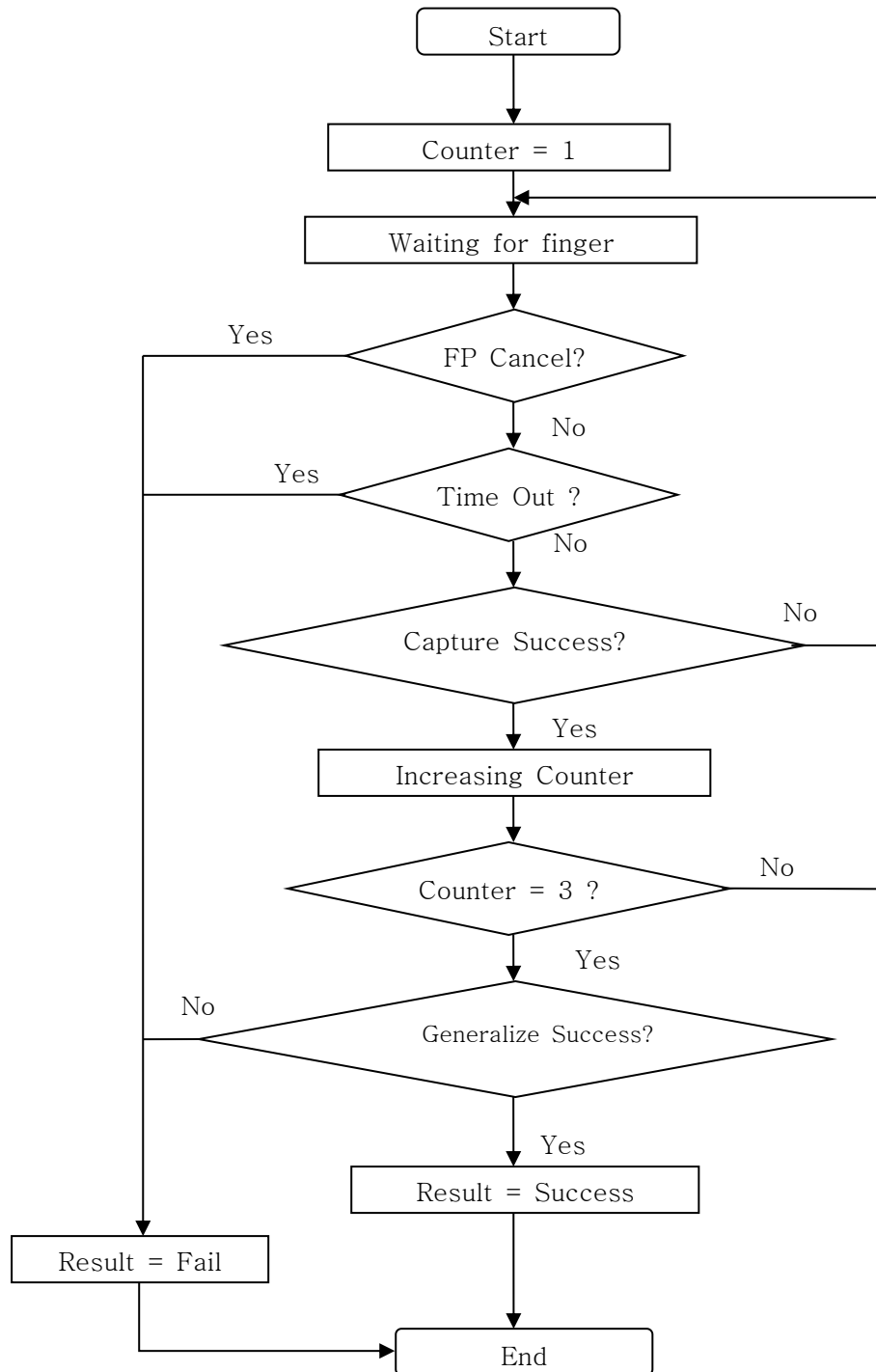


图6-1 注册流程

6.2.2 验证及识别流程 Verify & Identify

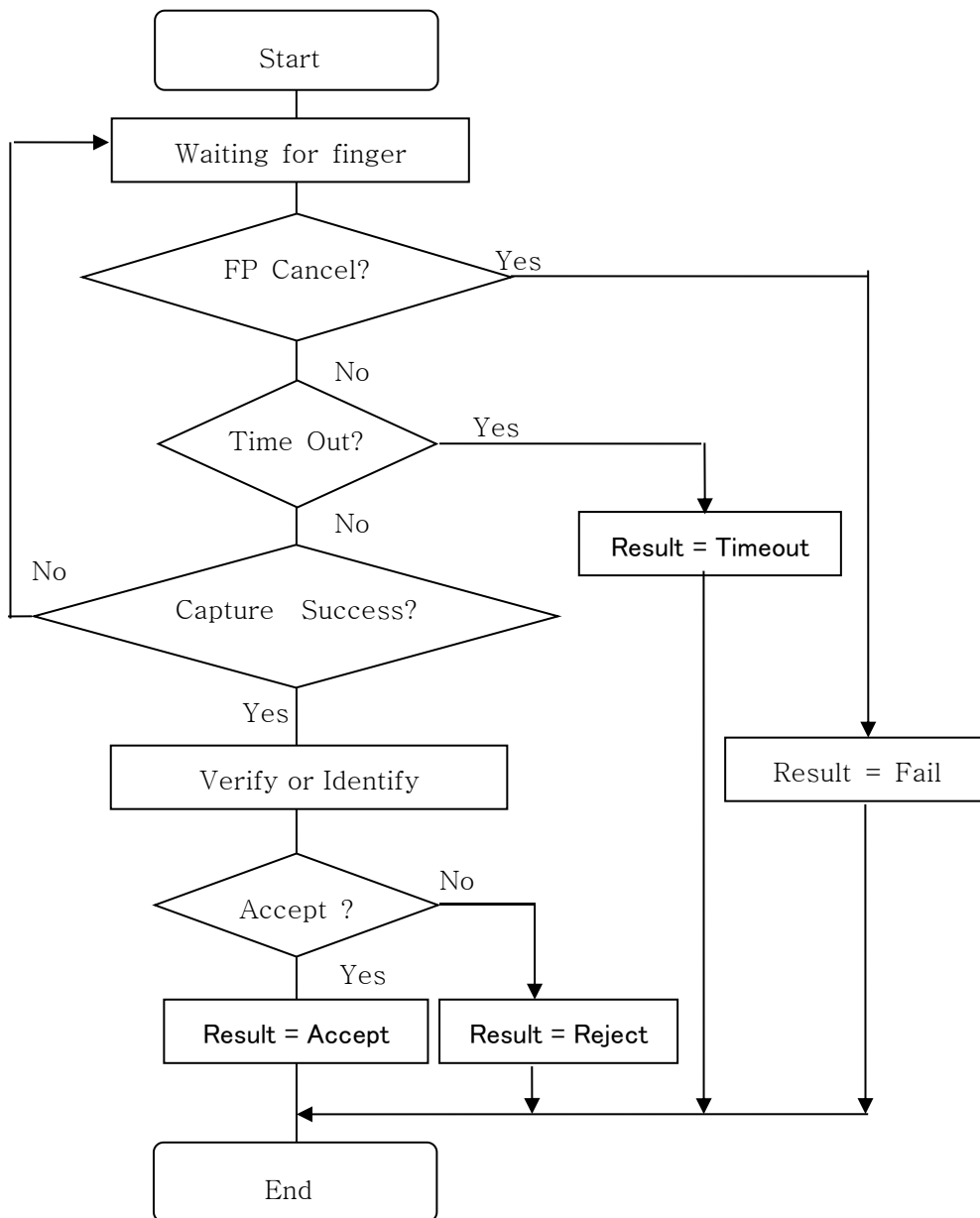


图6-2 验证及识别流程