

UART FINGERPRINT SENSOR (F)

COMMAND MANUAL

PREFACE

This manual is mainly used to introduce the UART Fingerprint Scanner (F) module (hereinafter referred to as "module" or "fingerprint module") serial communication protocol and command list, communication commands, and data packet formats. The manual will be updated intermittently according to product conditions, and no further notice will be provided.

Version	Update Time	Notes
V1.1	2025-07	First edition
V1.2	2025-10	Added LED control instructions

TABLE OF CONTENTS

Preface	1
Table of Contents	2
Communication Instructions	4
Serial Communication Parameters	4
Communication Process	4
Frame Sequence Transmission Order	4
Communication Protocol Description	5
Communication Processing Process	5
Packet Classification	6
Command packet	6
Response packet	6
Data Packet	6
Packet Frame Structure	6
Packet ID Code	6
Command Packet Frame Structure	6
Response Packet Frame Structure	7
Command Data Packet Frame Structure	7
Response Data Packet Frame Structure	7
Fingerprint Enrollment and Comparison Process	8
Command Introduction	10
Command Basic Concepts	10
Command List	10
Response and Error Code List	11
Command Detailed Description	12
Connection Test (CMD_TEST_CONNECTION 0x0001)	12
Set Parameter (CMD_SET_PARAM 0x0002)	12
Read Parameter (CMD_GET_PARAM 0x0003)	14
Read Device Information (CMD_DEVICE_INFO 0x0004)	14
Get fingerprint image (CMD_GET_IMAGE 0x0020)	15
Detect Finger (CMD_FINGER_DETECT 0x0021)	16
Upload Fingerprint Image to Host (CMD_UP_IMAGE_CODE 0x0022)	17
Download Fingerprint Image to Module (CMD_DOWN_IMAGE 0x0023)	18
Control Collector Backlight On/Off (CMD_SLED_CTRL 0x0024)	20
Save Fingerprint Template Data to Module Fingerprint Library (CMD_STORE_CHAR 0x0040)	20
Read Fingerprint from Module and Temporarily Store in Ram Buffer (CMD_LOAD_CHAR 0x0041)	21
Upload Fingerprint Template Temporarily Stored in Ram Buffer to Host (CMD_UP_CHAR 0x0042)	22

Download Fingerprint Template Data to Ram Buffer Specified by Module (CMD_DOWN_CHAR 0x0043)	23
Delete Fingerprints Within Specified ID Range (CMD_DEL_CHAR 0x0044)	24
Get First Available ID Within Specified ID Range (CMD_GET_EMPTY_ID 0x0045)	25
Check If Specified ID Is Registered (CMD_GET_STATUS 0x0046)	26
Check for Data Corruption in Fingerprint Library Within Specified Range (CMD_GET_BROKEN_ID 0x0047)	26
Get Total Number of Registered Fingerprints Within Specified ID Range (CMD_GET_ENROLL_COUNT 0x0048)	27
Generate Template from Fingerprint Image in ImageBuffer (CMD_GENERATE 0x0060)	28
Merge Fingerprint Template Data for Enrollment (CMD_MERGE 0x0061)	28
Compare Templates Between Two Ram Buffers (CMD_MATCH 0x0062)	29
Identify 1:N Within Specified ID Range (CMD_SEARCH 0x0063)	30
Compare Specified Ram Buffer Template with Library Template by ID (CMD_VERIFY 0x0064)	31
Set Module Serial Number (CMD_SET_MODULE_SN 0x0008)	32
Get Module Serial Number (CMD_GET_MODULE_SN 0x0009)	33
Get Registered ID List (CMD_GET_ENROLLED_ID_LIST 0x0049)	33
Communication Error Returned (Incorrect Command)	35
Precautions	36

COMMUNICATION INSTRUCTIONS

This chapter mainly introduces the serial port parameters, communication process, and communication transmission format of the fingerprint module

SERIAL COMMUNICATION PARAMETERS

Start bit: 1bit

Data bit: 8 bits

Stop bit: 1bit

Parity bit: None

Baud rate: 9600/19200/38400/57600/115200/230400/460800/921600, default value: 115200BPS

COMMUNICATION PROCESS

All commands must be sent and received in accordance with the principle of one send and one receive.

The host (Host) cannot send commands to the target module (TARGET) without receiving a response.

FRAME SEQUENCE TRANSMISSION ORDER

The byte (Byte) transmission follows the rule of sending the least significant bit first.

The word (Word) transmission follows the rule of low byte first and high byte last.

COMMUNICATION PROTOCOL DESCRIPTION

It takes time for the firmware to BOOT after the fingerprint module is powered on (i.e., the hardware and algorithm initialization time), and the host must wait for the fingerprint module to complete the initialization before sending commands to the fingerprint module.

After the fingerprint module is powered on and the GPIO and UART ports of the MCU are successfully initialized, it will send a 0x55 byte via UART as a handshake signal to notify the host.

When the host is waiting for the fingerprint module to initialize after powering on, it can enter the working state in advance by receiving this handshake signal.

Notes:

After the host control module is powered on, the communication process can be started in the following two ways:

1. Once the host receives the handshake signal 0x55 from the module, it can start sending commands to the module.
2. After the host control module is powered on, a delay of 280ms is required before commands can be sent to the module.

COMMUNICATION PROCESSING PROCESS

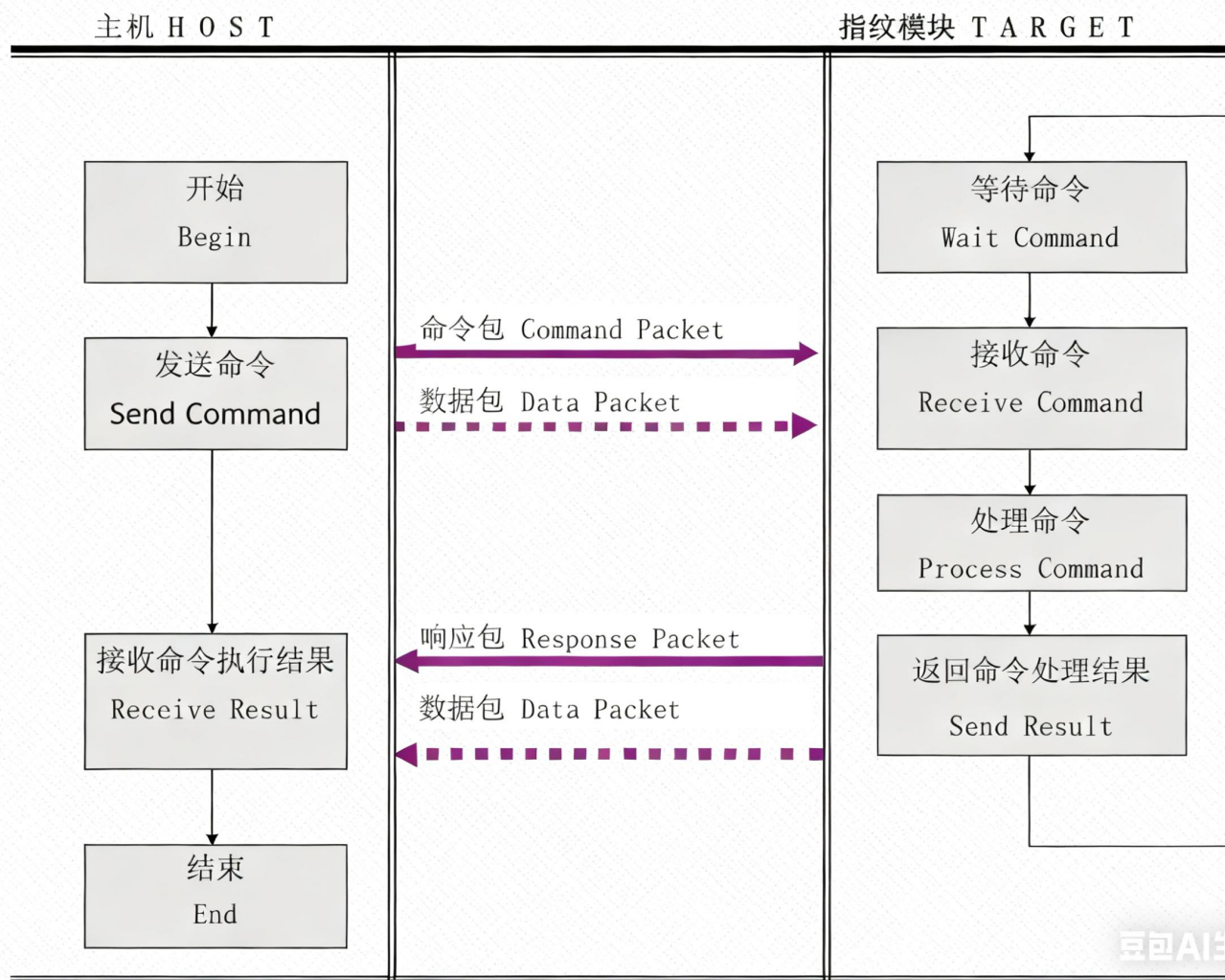


Figure 1 Fingerprint module communication process

Note:

During communication, the sending and receiving of all commands must follow the principle of one send and one receive. The Host cannot send commands to the TARGET module without receiving a response from the module.

PACKET CLASSIFICATION

COMMAND PACKET

- The command packet describes the command content from Host to Target.
- All commands issued from the Host are transmitted via the Command packet.
- The Command packet has a frame length of 26 bytes.

RESPONSE PACKET

- The response packet refers to the response content from Target to Host.
- All commands terminate their mission after receiving the corresponding processing result, i.e., the Response packet.
- The Response packet has a length of 26 bytes.

DATA PACKET

- When the length of the command parameters or response data exceeds 16 bytes, use the Data Packet to transmit data.
- The Host must inform the fingerprint module Target of the length of the command data packet using the Command packet before sending the packet
- The command parameters or the corresponding data packets have a maximum length of 500bytes

PACKET FRAME STRUCTURE

PACKET ID CODE

The first 2 bytes of the communication Packet indicate the ID code for the type of communication packet, as follows in the table:

Packet Type	ID Code
Command packet	0xAA55
Response packet	0x55AA
Command Data Packet	0xA55A
Response Data Packet	0x5AA5

1 Packet ID Code

COMMAND PACKET FRAME STRUCTURE

PREFIX		SID	DID	CMD		LEN		DATA				CKS	
0x55	0xAA	Source ID	Target ID	L	H	L	H	D0	D1	...	D15	L	H
0	1	2	3	4	5	6	7	8	9	...	23	24	25

The structure of the Command packet is as follows:

Byte Offset	Definition Field	Data Type	Type LEN	Description
0	PREFIX	WORD	2 bytes	Packet ID Code
2	SID	BYTE	1 byte	Source Device ID
3	DID	BYTE	1 byte	Destination Device ID
4	CMD	WORD	2 bytes	Command Code
6	LEN	WORD	2 bytes (=n, n < 16)	Length of DATA
8	DATA	Byte Array	16 bytes	Command Parameter (actual data is n bytes)
24	CKS	WORD	2 bytes	Check Sum: The lowest 2 bytes of arithmetic sum of all data from PREVIX to DATA field

RESPONSE PACKET FRAME STRUCTURE

PREFIX		SID	DID	RCM		LEN		RET		DATA				CKS	
0xAA	0x55	Source ID	Target ID	L	H	L	H	L	H	D0	D1	...	D15	L	H
0	1	2	3	4	5	6	7	8	9	10	11	...	23	24	25

The structure of the Response packet is as follows:

Byte Offset	Definition Field	Data Type	Type LEN	Description
0	PREFIX	WORD	2 bytes	Packet ID Code
2	SID	BYTE	1 byte	Source Device ID
3	DID	BYTE	1 byte	Destination Device ID
4	RCM	WORD	2 bytes	Command Code
6	LEN	WORD	2 bytes (=n, n < 16)	Length of DATA
8	RET	WORD	2 bytes	Result Code (0: Success, 1: Failure)
10	DATA	Byte Array	14 bytes	Response Data (actual data is n-2 bytes)
24	CKS	WORD	2 bytes	Check Sum: The lowest 2 bytes of arithmetic sum of all data from PREVIX to DATA field

COMMAND DATA PACKET FRAME STRUCTURE

PREFIX		SID	DID	CMD		LEN		DATA				CKS	
0xA5	0xA5	Source ID	Target ID	L	H	L	H	D0	D1	...	Dn-1	L	H
0	1	2	3	4	5	6	7	8	9	...	8+n-1	8+n	8+n

The structure of the Command Data Packet is as follows:

Byte Offset	Definition Field	Data Type	Type LEN	Description
0	PREFIX	WORD	2 bytes	Packet ID Code
2	SID	BYTE	1 byte	Source Device ID
3	DID	BYTE	1 byte	Destination Device ID
4	CMD	WORD	2 bytes	Command Code
6	LEN	WORD	2 bytes (=n, n < 16)	Length of DATA
8	DATA	Byte Array	n bytes	Command Parameter
8+n	CKS	WORD	2 bytes	Check Sum: The lowest 2 bytes of arithmetic sum of all data from PREVIX to DATA field

The Host must transmit a Command packet before sending the command data packet, so that the fingerprint module Target enters a Command Data packet reception waiting state.

In the DATA field of the Command packet, the length of the command data packet to be transmitted must be set.

The Host should transmit the Command Data Packet after confirming that the Target is in the waiting state for receiving command packets.

RESPONSE DATA PACKET FRAME STRUCTURE

PREFIX		SID	DID	RCM		LEN		RET		DATA				CKS	
0xA5	0x5A	Source ID	Target ID	L	H	L	H	L	H	D0	D1	...	Dn-3	L	H
0	1	2	3	4	5	6	7	8	9	10	11	...	8+n-1	8+n	8+n+1

The structure of the Response Data Packet is as follows:

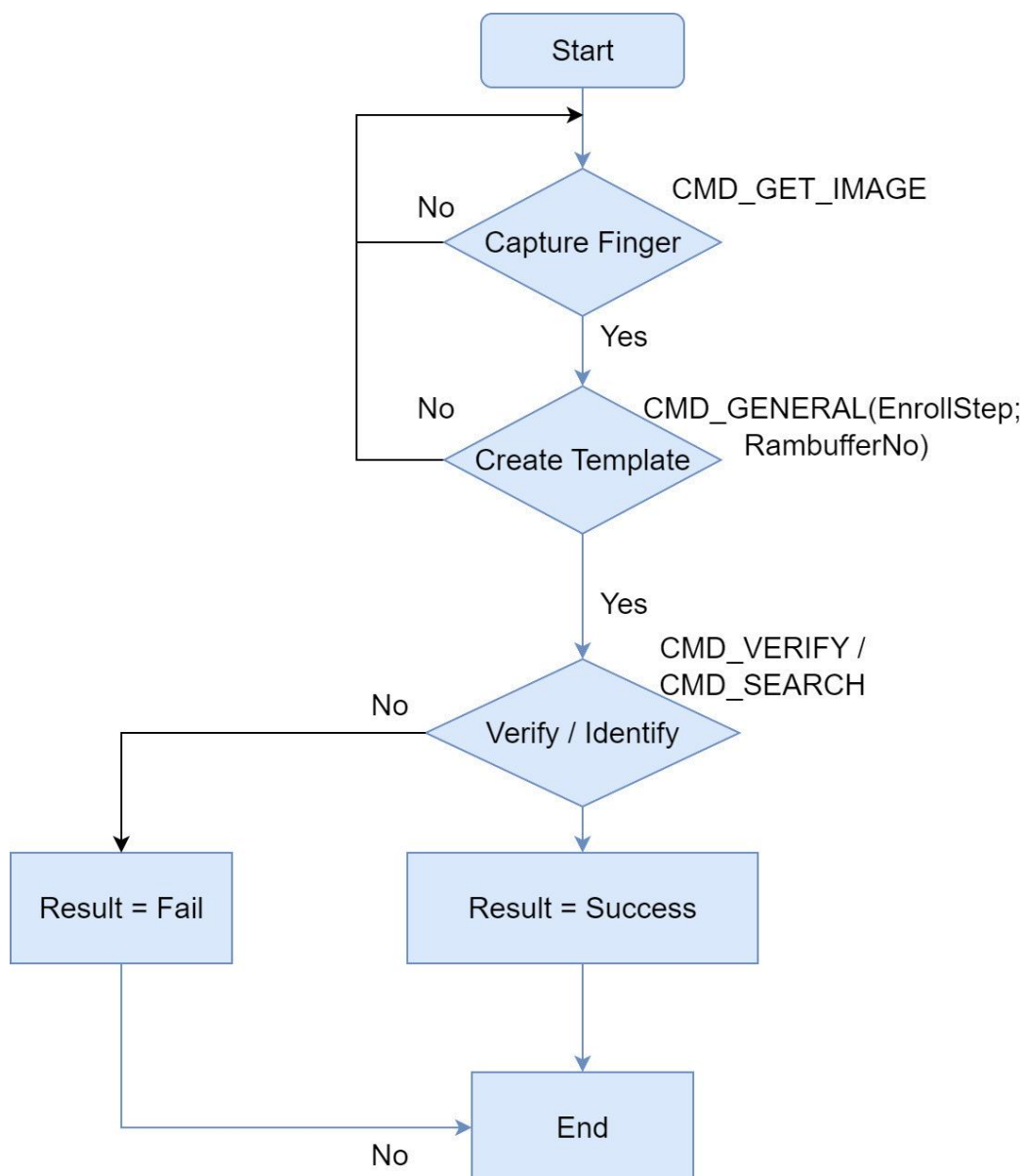
Byte Offset	Definition Field	Data Type	Type LEN	Description
0	PREFIX	WORD	2 bytes	Packet ID Code
2	SID	BYTE	1 byte	Source Device ID
3	DID	BYTE	1 byte	Destination Device ID
4	RCM	WORD	2 bytes	Response Code
6	LEN	WORD	2 bytes (=n, n < 500)	Length of DATA (RET + DATA)
8	RET	WORD	2 bytes	Result Code (0: Success, 1: Failure)
10	DATA	Byte Array	n-2 bytes	Response Data
8+n	CKS	WORD	2 bytes	Check Sum: The lowest 2 bytes of arithmetic sum of all data from PREVIX to DATA field

Note: When transferring data larger than 14 bytes from the module Target to Host, it is necessary to use the Response data packet.

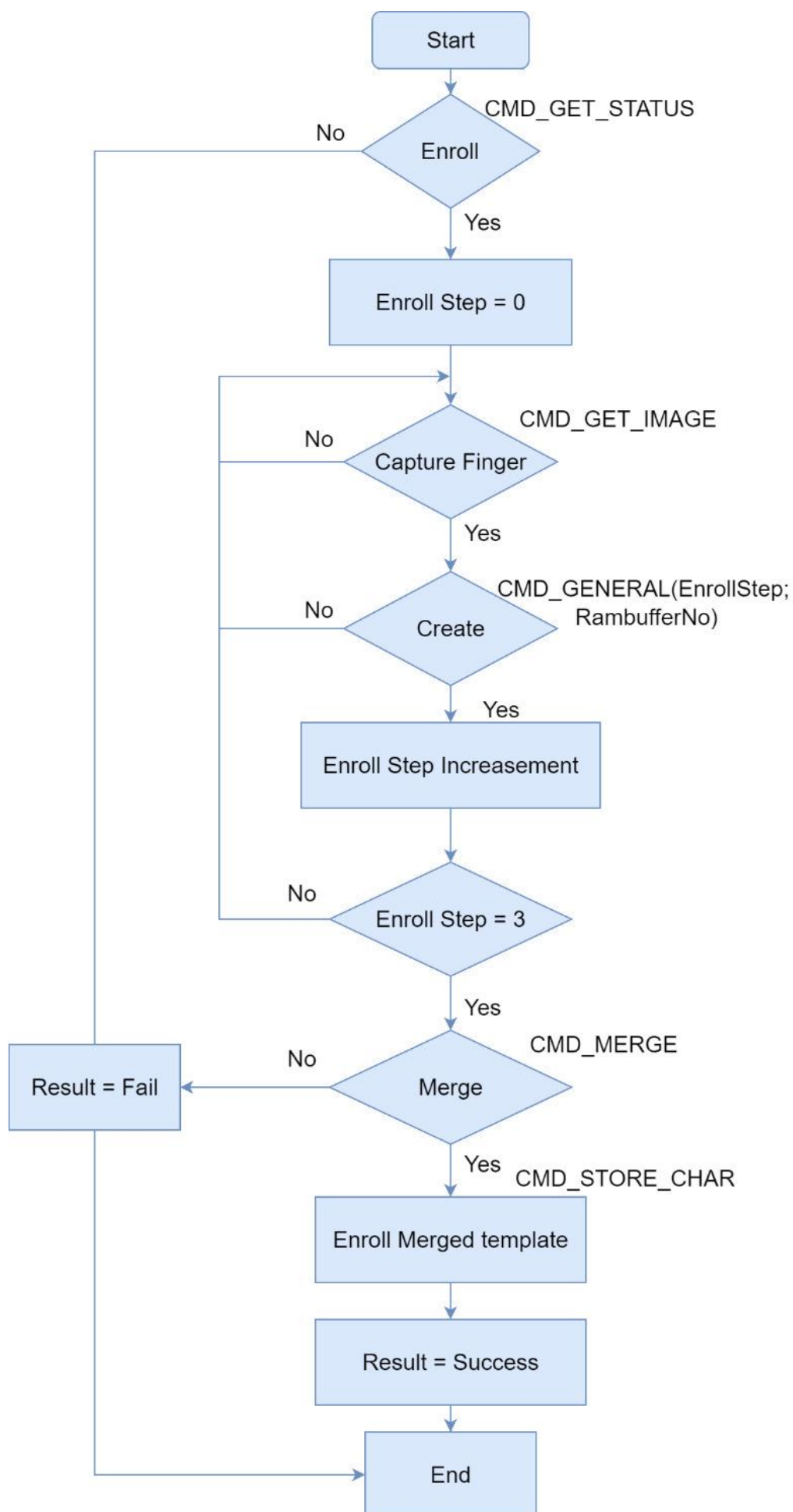
FINGERPRINT ENROLLMENT AND COMPARISON PROCESS

To facilitate user secondary development, a flowchart is used here to roughly illustrate the complete process of enrolling fingerprints into the fingerprint library and recognition.

FINGERPRINT COMPARISON PROCESS



FINGERPRINT ENROLLMENT PROCESS



COMMAND INTRODUCTION

COMMAND BASIC CONCEPTS

The execution of the command requires the ImageBuffer, which caches the temporary fingerprint image, and the Ram Buffer, which caches the temporary fingerprint template data.

ImageBuffer: Used for temporarily storing fingerprint image data.

Ram Buffer: Used for temporarily storing fingerprint template data. There are three Ram Buffers available in the module: Ram Buffer0/Ram Buffer1/Ram Buffer2.

Note: In the event of a power outage, data temporarily stored in ImageBuffer and Ram Buffer will be lost.

Fingerprint library: The collection of fingerprint templates stored in FLASH Memory can be written/read to/from fingerprint template data through CMD-STORE_CHAR/CMD_LOAD_CHAR.

Fingerprint template data length: 448 bytes (subject to change based on firmware version updates)

COMMAND LIST

No.	Command Name	Code	Function
1	CMD_TEST_CONNECTION	0x0001	Communication connection test of the device
2	CMD_SET_PARAM	0x0002	Set fingerprint module parameters: Device ID Security Level Duplication Check Baudrate Auto Learn
3	CMD_GET_PARAM	0x0003	Get fingerprint module parameters: Device ID Security Level Duplication Check Baudrate Auto Learn
4	CMD_GET_DEVICE_INFO	0x0004	Get fingerprint module device information
5	CMD_GET_IMAGE	0x0020	Get fingerprint images from the collector and store them in the domain ImageBuffer
6	CMD_FINGER_DETECT	0x0021	Check fingerprint input status (with/without fingers)
7	CMD_UP_IMAGE	0x0022	Upload fingerprint image stored in ImageBuffer to HOST
8	CMD_DOWN_IMAGE	0x0023	Download fingerprint images from HOST to ImageBuffer
9	CMD_SLED_CTRL	0x0024	Control the collector backlight switch
10	CMD_STORE_CHAR	0x0040	Save template data in specified ID Ram Buffer to fingerprint library with specified ID
11	CMD_LOAD_CHAR	0x0041	Read template data with a specified ID and store it into specified Ram Buffer (0/1/2)
12	CMD_UP_CHAR	0x0042	Upload fingerprint template data temporarily stored in specified Ram Buffer to HOST
13	CMD_DOWN_CHAR	0x0043	Download a fingerprint template from HOST to specified ID Ram Buffer of module
14	CMD_DEL_CHAR	0x0044	Delete all fingerprint templates within specified range from fingerprint library

15	CMD_GET_EMPTY_ID	0x0045	Get first template ID that can be registered (has not been registered) within specified range
16	CMD_GET_STATUS	0x0046	Get registration status of template with specified ID
17	CMD_GET_BROKEN_ID	0x0047	Check if there are any damaged fingerprint templates in the specified ID range
18	CMD_GET_ENROLL_COUNT	0x0048	Get number of registered templates within specified ID range
19	CMD_GENERATE	0x0060	Generate template data from fingerprint image stored in ImageBuffer and save it in specified Ram Buffer
20	CMD_MERGE	0x0061	Merge the data from two or three templates temporarily stored in Ram Buffer into the data of one template Note: The merged template data is temporarily stored in Ram Buffer 0
21	CMD_MATCH	0x0062	Perform a 1:1 comparison (verification) between two fingerprint templates in specified Ram Buffer
22	CMD_SEARCH	0x0063	Perform a 1:N search (recognition) between the template in specified Ram Buffer and all templates within specified ID range in fingerprint library
23	CMD_VERIFY	0x0064	Specify the fingerprint template in Ram Buffer and the fingerprint template with specified ID in fingerprint library
24	CMD_SET_MODULE_SN	0x0008	Set module serial number information (Module SN) in the device
25	CMD_GET_MODULE_SN	0x0009	Get module serial number (Module SN) of this device
26	CMD_GET_ENROLLED_ID_LIST	0x0049	Get the list of registered User IDs

RESPONSE AND ERROR CODE LIST

No.	Response and Error Code	Value	Description
1	ERR_SUCESS	0x00	Command processing successful
2	ERR_FAIL	0x01	Command processing failed
3	ERR_VERIFY	0x10	The 1:1 comparison with the Template in specified ID failed
4	ERR_IDENTIFY	0x11	A 1:N comparison has been performed, but no identical Template exists
5	ERR_TMPL_EMPTY	0x12	There is no registered Template in the specified ID
6	ERR_TMPL_NO_EMPTY	0x13	The Template already exists in the specified ID
7	ERR_ALL_TMPL_EMPTY	0x14	There is no registered Template
8	ERR_EMPTY_ID_NOEXIST	0x15	There is no Template ID available for registration
9	ERR_BROKEN_ID_NOEXIST	0x16	There is no damaged Template
10	ERR_INVALID_TMPL_DATA	0x17	The specified Template Data is invalid
11	ERR_DUPLICATION_ID	0x18	This fingerprint has been registered
12	ERR_BAD_QUALITY	0x19	Poor fingerprint image quality
13	ERR_MERGE_FAIL	0x1A	Template merge failed
15	ERR_MEMORY	0x1C	External Flash flashing error
16	ERR_INVALID_TMPL_NO	0x1D	Invalid specified Template ID
17	ERR_INVALID_PARAM	0x22	Incorrect parameters are used
18	ERR_GEN_COUNT	0x25	Invalid number of fingerprints to merge
19	ERR_TIME_OUT	0x23	No fingerprint input within the TimeOut period
20	ERR_INVALID_BUFFER_ID	0x26	Buffer ID value is incorrect
21	ERR_FP_NOT_DETECTED	0x28	No fingerprint input on the collector
22	ERR_FP_CANCEL	0x41	Command cancelled

COMMAND DETAILED DESCRIPTION

CONNECTION TEST (CMD_TEST_CONNECTION 0X0001)

FUNCTION

Check the connection status between Target and Host.

The Host needs to send this command first to check the connection status with the Target.

If it fails, it can be considered that the connection with Target is abnormal, or Target is not working properly, or the serial port baud rate is incorrect.

SEQUENCE

If the connection is normal, it returns ERR_SUCCESS.

COMMAND AND RESPONSE

PREFIX	0xAA55
SID	Source Device ID
DID	Destination Device ID
CMD	0x0001
LEN	0
DATA	No data
PREFIX	0x55AA
SID	Source Device ID
DID	Destination Device ID
RCM	0x0001
LEN	2
RET	Result Code (ERR_SUCCESS/ ERR_FAIL)
DATA	No data

Example - Host sends CMD_TEST_CONNECTION command and module response

Host command: 55 AA 00 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01

Module response: AA 55 01 00 01 00 02 00 00 00 00 00 00 00 00 00 00 00 00 00 03 01

SET PARAMETER (CMD_SET_PARAM 0X0002)

FUNCTION

Set the values of device parameters (Device ID, Security Level, Baudrate, Duplication Check, Auto Learn, FP TimeOut) based on the specified Parameter Type and return the result.

SEQUENCE

- If the specified Parameter Type is invalid, return ERR_INVALID_PARAM.

- If the specified Parameter Value is invalid, return ERR_INVALID_PARAM.
- Set the Parameter Value according to the Parameter Type and return the result.

COMMAND AND RESPONSE

PREFIX	0xAA55	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0002	
LEN	5	
DATA	1 byte	Parameter Type
	4 bytes	Parameter Value
PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0002	
LEN	2	
RET	Result Code (ERR_SUCCESS/ ERR_FAIL)	
DATA	No data	

PARAMETER TYPE

Type	Parameter Value Description
0	Indicates the Device ID of this module. Adjustable range: 1-255
1	Indicates Security Level, the default value is: 3 Adjustable value: 1-5 (Note: For some firmware with special requirements, it can be set to 1-10) The larger the value: the lower the False Acceptance Rate (FAR), but the higher the False Rejection Rate (FRR). Simply put, it is as follows: The higher the security level set: the less likely it is to have false recognition, but it is also not easy to identify; On the contrary, the lower the security level: the easier it is to pass, but the more likely it is to result in false recognition
2	Fingerprint (Duplication Check) status on/off. It can be set to 0 or 1. If it is 1, when processing the CMD_STORE_CHAR command, check if there are any registered duplicate fingerprints. If it is 0, no duplicate check is performed. Note: Fingerprint duplicate check on/off, only effective for firmware of general algorithms (498-byte feature algorithm), other algorithms' firmware does not check for fingerprint duplicates (even if set to 1, it does not perform fingerprint duplicate check)
3	The index value of Baudrate. Adjustable index value: 1-8. Default: 5 (i.e. 115200bps) 1:9600bps; 2:19200bps; 3:38400bps; 4:57600bps; 5:115200bps; 6:230400bps; 7:460800bps; 8:921600bps
4	Indicates the state (ON/OFF) of fingerprint template Auto Learn. It can be set to 0 or 1. If 1: Then perform intelligent updates when processing CMD_SEARCH, CMD_VERIFY commands. If 0: Do not perform intelligent update.

Example 1: Set the baud rate to 57600bps

Host command: 55 AA 00 00 02 00 05 00 03 04 00 00 00 00 00 00 00 00 00 00 0d 01

Module response: AA 55 01 00 02 00 02 00 00 00 00 00 00 00 00 00 00 00 00 00 04 01

Example 2: Set the security level = 5

Host command: 55 AA 00 00 02 00 05 00 01 05 00 00 00 00 00 00 00 00 00 00 0c 01

Module response: AA 55 01 00 02 00 02 00 00 00 00 00 00 00 00 00 00 00 00 00 04 01

READ PARAMETER (CMD_GET_PARAM 0X0003)

FUNCTION

According to the specified Parameter Type, obtain parameter values such as module (Device ID, Security Level, Baudrate, Duplication Check, Auto Learn).

For Parameter Type, please refer to CMD_SET_PARAM.

SEQUENCE

- If the specified Parameter Type is invalid, return ERR_INVALID_PARAM.
- Returns the corresponding device parameter for the specified Parameter Type.

COMMAND AND RESPONSE

PREFIX	0xAA55	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0003	
LEN	1	
DATA	1 byte	Parameter Type
PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0003	
LEN	Success: 6, Failure: 2	
RET	Result Code (ERR_SUCCESS/ ERR_FAIL)	
DATA	4 bytes	When successful: Parameter Value

Example 1: Read current security level (return security level = 3)

Host command: 55 AA 00 00 03 00 01 00 01 00 00 00 00 00 00 00 00 00 00 00 04 01

Module response: AA 55 01 00 03 00 06 00 00 00 03 00 00 00 00 00 00 00 00 00 0C 01

Example 2: Read the current TimeOut value (TimeOut=5S); For sliding fingerprint modules

Host command: 55 AA 00 00 03 00 01 00 05 00 00 00 00 00 00 00 00 00 00 00 08 01

Module response: AA 55 01 00 03 00 06 00 00 00 05 00 00 00 00 00 00 00 00 00 0E 01

READ DEVICE INFORMATION (CMD_DEVICE_INFO 0X0004)

FUNCTION

Get the module version and other device information (Device Information of Target).

The information format of this device is as follows:

ID_Protocol Stack and Algorithm Name_Algorithm Chip Model_Collector Model_None or Module Hardware Name_None or Inner (Fingerprint Capacity) Version Number

Focus on the algorithm name and fingerprint capacity information in the module information.

SEQUENCE

- First, use the command response packet to send the data length of the next response packet to the HOST.
- Use response packets to send Device Information.

COMMAND AND RESPONSE

PREFIX	0xAA55	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0004	
LEN	0	
DATA	No data	
PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0004	
LEN	4	
RET	Result Code (ERR_SUCCESS/ ERR_FAIL)	
DATA	2 bytes	Data length of the data response packet
When successful		
PREFIX	0x5AA5	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0004	
LEN	2 + Device Information length	
RET	Result Code (ERR_SUCCESS)	
DATA	Device Information	

Example - Read module information CMD_DEVICE_INFO

Host command: 55 AA 00 00 04 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 03 01

Module response: AA 55 01 00 04 00 04 00 00 00 1A 00 00 00 00 00 00 00 00 00 00 00 22 01

Response packet: A5 5A 01 00 04 00 1F 00 00 00 53 45 4F 4E 5F 47 44 5F 46 50 43 31 30 32

30 28 32 30 30 30 66 70 29 20 56 31 2E 30 00 2B 08

The blue highlighted data block is the ASCII code for the device information "SEON_GD_FPC1020(2000fp) V1.0".

GET FINGERPRINT IMAGE (CMD_GET_IMAGE 0X0020)

FUNCTION

Collect fingerprint images from the collector and save them in ImageBuffer.

SEQUENCE

COMMAND AND RESPONSE

Target response: AA 55 01 00 21 00 03 00 00 00 00 00 00 00 00 00 00 00 00 00 00 24 01

Example 2: Fingerprint detected

Host command: 55 AA 00 00 21 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 20 01

Target response: AA 55 01 00 21 00 03 00 00 00 01 00 00 00 00 00 00 00 00 00 00 25 01

UPLOAD FINGERPRINT IMAGE TO HOST (CMD_UP_IMAGE_CODE 0X0022)

FUNCTION

According to the specified Image Type, send the image saved in ImageBuffer to the Host.

If Image Type is 0: send the full image (192x192);

If it is 1: send 1/4 of the image (1 out of every 4 points) (96x96).

SEQUENCE

- If the specified Image Type is invalid, return ERR_INVALID_PARAM.
- Use the command response packet to send the size of the image to be received by the HOST to the HOST.
- According to the Image Type, the image is divided into 496bytes and sent to the HOST using the response packet.

COMMAND AND RESPONSE

PREFIX	0xAA55	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0022	
LEN	0	
DATA	1 byte	Image Type (0: Full, 1: Quarter)
PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0022	
LEN	6/2	
RET	Result Code	
DATA	2 bytes	Success: Image width Full: 192 Quarter: 96
	2 bytes	Success: Image height Full: 192 Quarter: 96
When successful: Target sends response packet to HOST		
PREFIX	0x5AA5	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0022	
LEN	4 + Image data length	
RET	Result Code (ERR_SUCCESS)	
DATA	Image data length (2 bytes) + Image data	

Notes:

1. Before calling this command, you must first call CMD_GET_IMAGE to save the fingerprint image in the ImageBuffer.
2. High-resolution mode (Full Mode) Width * Height: 242*266/202*258/128*436
3. Low-resolution mode (Quarter Mode) Width * Height: 121*133/101*129/64*218

Example 1: Upload full-resolution 202*258 sensor fingerprint image

Host command: 55 AA 00 00 22 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 22 01

Module response: AA 55 01 00 22 00 06 00 00 00 CA 00 02 01 00 00 00 00 00 00 00 00 00 F5 01

Note: Full image width = 0xCA = 202, full image height = 0x102 = 258

Module response packet: The image data size is 202*258=52116 bytes, divided into 105*496 bytes + 1*36 bytes

A5 5A 01 00 22 00 F4 01 00 00 F0 01 496-byte data of this frame 2-byte checksum

... a total of 105 response packets containing 496 bytes of image data

A5 5A 01 00 22 00 28 00 00 00 24 00 36-byte data of last frame 2-byte checksum

The last response packet containing 36 bytes of image data

Example 2: Upload optical fingerprint image with 1/4 image resolution

Host command: 55 AA 00 00 22 00 01 00 01 00 00 00 00 00 00 00 00 00 00 00 23 01

Module response: AA 55 01 00 22 00 06 00 00 00 65 00 81 00 00 00 00 00 00 00 00 0E 02

Note: 1/4 image width=0x65=101, full image height=0x81=129

Module response packet: The image data size is 202*258/4=13029 bytes, divided into 26*496 bytes + 1*133 bytes

A5 5A 01 00 22 00 F4 01 00 00 F0 01 496-byte data of this frame 2-byte checksum

... a total of 26 response packets containing 496 bytes of image data

A5 5A 01 00 22 00 89 00 00 00 85 00 133-byte data of last frame 2-byte checksum

The last response packet containing 133 bytes of image data

DOWNLOAD FINGERPRINT IMAGE TO MODULE (CMD_DOWN_IMAGE 0X0023)
FUNCTION

Save the image data received from the Host in ImageBuffer.

The Host sends the image to the Target in units of 496bytes. At this time, send the image data ID simultaneously.

Note: Fingerprint image requirements: Resolution: 500DPI, Grayscale: 8-bit grayscale

Pixel size: Optical collector: 192*192

SEQUENCE

- If the image height or image width is incorrect, return ERR_INVALID_PARAM.
- Use the response packet to return ERR_SUCCESS.
- Receive command packet to save the image in ImageBuffer.

COMMAND AND RESPONSE

PREFIX	0xAA55	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0023	
LEN	4	
DATA	2 bytes	Image width: 192
	2 bytes	Image height: 192
PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0023	
LEN	2	
RET	Result Code	
DATA	0	
Command packet		
PREFIX	0xA55A	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0023	
LEN	2 + Image data size	
DATA	Image data ID (2 bytes) + Image data	
Response packet		
PREFIX	0x5AA5	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0023	
LEN	2	
RET	Result Code	
DATA	0	

...

Command packet	
PREFIX	0xA55A
SID	Source Device ID
DID	Destination Device ID
CMD	0x0023
LEN	2 + Image data size
DATA	Image data ID (2 bytes) + Image data
Response packet	
PREFIX	0x5AA5
SID	Source Device ID
DID	Destination Device ID
RCM	0x0023
LEN	2
RET	Result Code
DATA	0

For example: Download fingerprint image to ImageBuffer

Host command: 55 AA 00 00 23 00 04 00 CA 00 02 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 F3 01

Module response: AA 55 01 00 23 00 02 00 25 01

Command packet: 5A A5 00 00 23 00 F2 01 00 00 496-byte data of this block 2-byte checksum

- If Duplication Check is set to OFF, the fingerprint template data in the specified Ram Buffer is directly registered in the fingerprint library with the specified ID and the result is returned.
- If Duplication Check is set to ON, it will perform a 1:N comparison between the Template in the specified Ram Buffer and all Templates in the registered fingerprint library. If a successful comparison template exists, indicating that the fingerprint has been registered, then return (RET): ERR_DUPLICATION_ID, and DATA returns the ID of the successfully compared Template. Otherwise, register the template in the fingerprint library that specifies the Template ID and return its results.

COMMAND AND RESPONSE

PREFIX	0xAA55	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0040	
LEN	4	
DATA	2 bytes	Template ID
	2 bytes	Ram Buffer ID
PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0040	
LEN	LEN length is related to the result code: 2/4 When the Result Code is ERR_DUPLICATION_ID: LEN=4; Otherwise: LEN=2	
RET	Result Code	
DATA	2 bytes	Data is related to the result code: 0 or duplicate fingerprint ID When the result code is ERR_DUPLICATION_ID, it is: Duplicate fingerprint ID; Otherwise: 0

Example - Save the template data in RamBuffer0 to the module library with the specified ID 1:

Host command packet: 55 AA 00 00 40 00 04 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 44 01

Module response packet: AA 55 01 00 40 00 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 42 01

READ FINGERPRINT FROM MODULE AND TEMPORARILY STORE IN RAM BUFFER (CMD_LOAD_CHAR 0X0041)

FUNCTION

Retrieve the fingerprint template from the specified ID in the fingerprint library and store it in the specified Ram Buffer.

SEQUENCE

- If the specified Template ID is invalid, an error code ERR_INVALID_TMPL_NO is returned.
- If no Template is registered in the specified Template ID, an error code ERR_TMPL_EMPTY is returned.
- If the specified Ram Buffer ID is invalid, an error code ERR_INVALID_BUFFER_ID is returned.
- Save the Template with the specified ID in the specified Ram Buffer and return ERR_SUCCESS.- Save the Template with the specified ID in the specified Ram Buffer and return ERR_SUCCESS.

COMMAND AND RESPONSE

PREFIX	0xAA55	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0041	
LEN	4	
DATA	2 bytes	Template ID
	2 bytes	Ram Buffer ID
PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0041	
LEN	2	
RET	Result Code	
DATA	0	

Example - Read the template data with ID 1 and temporarily store it in RamBuffer0:

Host command packet: 55 AA 00 00 41 00 04 00 01 00 00 00 00 00 00 00 00 00 00 00 45 01

Module response packet: AA 55 01 00 41 00 02 00 00 00 00 00 00 00 00 00 00 00 00 00 43 01

UPLOAD FINGERPRINT TEMPLATE TEMPORARILY STORED IN RAM BUFFER TO HOST (CMD_UP_CHAR 0X0042)

FUNCTION

Send the Template from the specified Ram Buffer to Host.

SEQUENCE

- If the specified Ram Buffer ID is invalid, return ERR_INVALID_BUFFER_ID.
- Use the command response packet to send the size of the Template data to be received by the HOST to the HOST.
- Use the response packet to send the Template data with the specified ID to the HOST.

COMMAND AND RESPONSE

PREFIX	0xAA55
SID	Source Device ID
DID	Destination Device ID
CMD	0x0042
LEN	2
DATA	RamBuffer ID
PREFIX	0x55AA
SID	Source Device ID
DID	Destination Device ID
RCM	0x0042
LEN	4
RET	Result Code
DATA	Success: Next data response packet length (Template Record Size + 2) Failure: Error code
When successful, the module sends a response packet	
PREFIX	0x55AA
SID	Source Device ID

DID	Destination Device ID
RCM	0x0042
LEN	Template Record Size + 2
RET	Result Code
DATA	Template Record Data

Note:

1. Before calling this command, you must first call CMD_GENERATE, CMD_DOWN_CHAR, CMD_LOAD_CHAR

One of these commands will save the Template in a Ram Buffer.

Example - Upload template data from RamBuffer0 to HOST (SEONU7, fingerprint template data is 448 bytes)

Host command packet: 55 AA 00 00 42 00 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 43 01

Module response packet: AA 55 01 00 42 00 04 00 00 00 C0 01 00 00 00 00 00 00 00 00 00 00 00 00 07 02

Response packet: A5 5A 01 00 42 00 C2 01 00 00 496-byte data of this fingerprint template 2-byte checksum

DOWNLOAD FINGERPRINT TEMPLATE DATA TO RAM BUFFER SPECIFIED BY MODULE (CMD_DOWN_CHAR 0X0043)

FUNCTION

Receive fingerprint Template Data from the Host and save it in the specified Ram Buffer.

SEQUENCE

- Host sends a command packet to make the Target enter the waiting state for receiving the data (Ram Buffer + Template). The DATA field in this command packet has already set the length of the next command packet to be sent.
- Target checks the accuracy of the received command packet.

If incorrect, return the error code and end processing.

If the size of the data to be received is incorrect, return ERR_INVALID_PARAM.

If correct, send a response packet to the HOST to indicate that the module has entered the waiting state for data reception (Ram Buffer ID + Template data).

- If the Host receives a response packet indicating that the Target has entered the data reception waiting state, it uses the RamBuffer and Template data specified in the command packet and sends it to the Target.
- After receiving the command packet, if the Ram Buffer ID is invalid, return ERR_INVALID_BUFFER_ID.
- Check the CheckSum of the received Template. If incorrect, return ERR_INVALID_TMPL_DATA.
- Save the received Template in the specified Ram Buffer and return ERR_SUCCESS.

COMMAND AND RESPONSE

PREFIX	0xAA55
SID	Source Device ID
DID	Destination Device ID
CMD	0x0043
LEN	2

DATA	2 + Template Record Size
PREFIX	0x55AA
SID	Source Device ID
DID	Destination Device ID
RCM	0x0043
LEN	4
RET	Result Code
DATA	0
Command packet	
PREFIX	0xA55A
SID	Source Device ID
DID	Destination Device ID
CMD	0x0043
LEN	2 + Template size (448)
DATA	Ram Buffer ID (2byte) + Template data
PREFIX	0x5AA5
SID	Source Device ID
DID	Destination Device ID
RCM	0x0043
LEN	4
RET	Result Code
DATA	0

Note: For Templates saved in RamBuffer2, if you call (CMD_SEARCH, CMD_VERIFY, CMD_GENERATE, CMD_STORE_CHAR, CMD_DEL_CHAR, CMD_GET_EMPTY_ID, CMD_GET_STATUS, GET_BROKEN_ID, CMD_GETN_ENROLL_COUNT) the following commands, they will be cleared. Try to avoid using RamBuffer2.

Example - Download fingerprint template data to the module's RamBuffer0

Host command: 55 AA 00 00 43 00 02 00 C2 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 07 02

Command response: AA 55 01 00 43 00 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 45 01

Data block packet: 5A A5 00 00 43 00 C2 01 00 00 448-byte fingerprint template data 2-byte checksum

Response packet: A5 5A 01 00 43 00 02 00 00 00 45 01; The length of the response packet is 12 bytes due to the lack of data

DELETE FINGERPRINTS WITHIN SPECIFIED ID RANGE (CMD_DEL_CHAR 0X0044)

FUNCTION

Delete all registered Templates within the specified ID range (starting Template ID~ ending Template ID).

SEQUENCE

- If the specified range is invalid, return ERR_INVALID_PARAM.
- If no Template is registered within the specified ID, return ERR_TMPL_EMPTY.
- Delete all Templates registered within the specified range and return the result.

COMMAND AND RESPONSE

PREFIX	0xAA55	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0044	
LEN	4	
DATA	2 bytes	Starting Template ID
	2 bytes	Ending Template ID
PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0044	
LEN	2	
RET	Result Code	
DATA	0	

For example: Delete all fingerprints with IDs 1-500 from the library

Host command: 55 AA 00 00 44 00 04 00 01 00 F4 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 3D 02

Module response: AA 55 01 00 44 00 02 00 46 01

GET FIRST AVAILABLE ID WITHIN SPECIFIED ID RANGE (CMD_GET_EMPTY_ID 0X0045)

FUNCTION

Get the first Template ID within the specified range (starting Template ID ~ ending Template ID) that can be registered (i.e., a Template that has not been registered).

SEQUENCE

- If the specified range is invalid, return ERR_INVALID_PARAM.
- Search for the first ID that can be registered within the specified range.

If it exists, return its value. Otherwise, return ERR_EMPTY_ID_NOEXIST.

COMMAND AND RESPONSE

PREFIX	0xAA55	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0045	
LEN	4	
DATA	2 bytes	Starting Template ID
	2 bytes	Ending Template ID
PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0045	
LEN	Success: 4, Failure: 2	
RET	Result Code	
DATA	2 bytes	When successful: The first Template ID that can be registered

Example - Obtain the first available registration ID within the range of 1-2000 (0x1-0x07D0), the result is 11

Host command: 55 AA 00 00 45 00 04 00 01 00 D0 07 00 00 00 00 00 00 00 00 00 00 20 02

Module response: AA 55 01 00 45 00 04 00 00 00 0B 00 00 00 00 00 00 00 00 00 00 00 00 54 01

CHECK IF SPECIFIED ID IS REGISTERED (CMD_GET_STATUS 0X0046)

FUNCTION

Get the registration status of the specified Template ID.

SEQUENCE

If the specified Template ID is invalid, an error code ERR_INVALID_TMPL_NO is returned.

If a Template is already registered with the specified ID, return 1. Otherwise, return 0.

COMMAND AND RESPONSE

PREFIX	0xAA55	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0046	
LEN	2	
DATA	Specified Template ID	
PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0046	
LEN	Success: 3, Failure: 2	
RET	Result Code	
DATA	2 bytes	When successful: Registration status (1: Registered, 0: Not registered)

Example - Get the registration status with ID=1, which can be registered

Host command: 55 AA 00 00 46 00 02 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 48 01

Module response: AA 55 01 00 46 00 03 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 49 01

CHECK FOR DATA CORRUPTION IN FINGERPRINT LIBRARY WITHIN SPECIFIED RANGE (CMD_GET_BROKEN_ID 0X0047)

FUNCTION

Check if the registered templates within the specified range (starting Template number ~ ending Template number) are corrupted.

In the Flash Write operation, templates may be damaged due to reasons such as sudden power outages.

HOST uses this command at any time (for example, the initial startup of Target) to check the damage status of the template.

Damaged templates need to be deleted and re-registered.

SEQUENCE

- If the specified range is invalid, return ERR_INVALID_PARAM.
- Check the damage status of all registered templates within the specified range.

If there are damaged templates, return the number of damaged templates and the ID of the first damaged template.

Otherwise, the number of templates and the template ID are both 0.

COMMAND AND RESPONSE

PREFIX	0xAA55	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0047	
LEN	4	
DATA	2 bytes	Starting Template ID
	2 bytes	Ending Template ID
PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0047	
LEN	Success: 6, Failure: 2	
RET	Result Code	
DATA	2 bytes	When successful: Number of damaged templates
	2 bytes	When successful: First damaged Template ID

Example-Get the ID numbers of fingerprint damage within the range of 1-200

Host command: 55 AA 00 00 47 00 04 00 01 00 C8 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 13 02

Module response: AA 55 01 00 47 00 06 00 4D 01

GET TOTAL NUMBER OF REGISTERED FINGERPRINTS WITHIN SPECIFIED ID RANGE
(CMD_GET_ENROLL_COUNT 0X0048)

FUNCTION

Get the total number of registered fingerprints within the specified range (starting Template ID ~ ending Template ID).

SEQUENCE

- If the specified range is invalid, return ERR_INVALID_PARAM.
- Return the number of fingerprints registered in the module.

COMMAND AND RESPONSE

PREFIX	0xAA55	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0048	
LEN	4	
DATA	2 bytes	Starting Template ID
	2 bytes	Ending Template ID
PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	

RCM	0x0048	
LEN	Success: 4, Failure: 2	
RET	Result Code	
DATA	2 bytes	Number of registered Templates

Example - Get the total number of registered users within the range of 1-200 (0x0001~0x00C8), with a total of 10 (0x000A)

Host command: 55 AA 00 00 48 00 04 00 01 00 C8 00 00 00 00 00 00 00 00 00 00 00 00 00 00 14 02

Module response: AA 55 01 00 48 00 04 00 00 00 0A 00 00 00 00 00 00 00 00 00 00 00 00 00 00 56 01

GENERATE TEMPLATE FROM FINGERPRINT IMAGE IN IMAGEBUFFER (CMD_GENERATE 0X0060)

FUNCTION

Generate a fingerprint template from the fingerprint image in the ImageBuffer and save it in the specified Ram Buffer.

SEQUENCE

- If the specified Ram Buffer ID is invalid, an error code ERR_INVALID_BUFFER_ID is returned.
- Check the correctness of the image in the ImageBuffer. If incorrect, return ERR_BAD_QUALITY.
- Save the generated Template in the specified Ram Buffer and return ERR_SUCCESS.

COMMAND AND RESPONSE

PREFIX	0xAA55	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0060	
LEN	2	
DATA	2 bytes	RamBuffer ID
PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0060	
LEN	2	
RET	Result Code	
DATA	0	

Note:

The CMD_GENERATE command generates Template Data from ImageBuffer. Therefore, before calling this command, the CMD_GET_IMAGE command must be called first to save the image in the ImageBuffer.

Example - Generate template data from ImageBuffer and save it in RamBuffer0

Host command: 55 AA 00 00 60 00 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 61 01

Module response: AA 55 01 00 60 00 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 62 01

MERGE FINGERPRINT TEMPLATE DATA FOR ENROLLMENT (CMD_MERGE 0X0061)

FUNCTION

Merge the templates temporarily stored in the Ram Buffer to generate template data and then save in the specified Ram Buffer.

The number of templates that can be merged is 2 or 3:

If 2: Merge the Templates of Ram Buffer0 and Ram Buffer1.

If 3: Merge the Templates of Ram Buffer0, Ram Buffer1 and Ram Buffer2.

SEQUENCE

- If the specified Ram Buffer ID is invalid, an error code ERR_INVALID_BUFFER_ID is returned.
- If the number of merged templates is invalid, return ERR_GEN_COUNT.
- According to the number of merge templates, merge the Templates and generate one Template. If the merge fails, return an error code.
- Save the generated Template in the specified Ram Buffer and return ERR_SUCCESS.

COMMAND AND RESPONSE

PREFIX	0xAA55	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0061	
LEN	3	
DATA	2 bytes	RamBuffer ID
	1 byte	Number of Merges (2/3)
PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0061	
LEN	2	
RET	Result Code	
DATA	0	

Example - Merge 3 temporary fingerprint template in RamBuffer into 1 fingerprint template data

Host command: 55 AA 00 00 61 00 03 00 00 00 03 00 00 00 00 00 00 00 00 00 00 00 00 00 66 01

Module response: AA 55 01 00 61 00 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 63 01

COMPARE TEMPLATES BETWEEN TWO RAM BUFFERS (CMD_MATCH 0X0062)

FUNCTION

Compare the templates in the two specified Ram buffers.

SEQUENCE

- If the specified Ram Buffer ID is invalid, an error code ERR_INVALID_BUFFER_ID is returned.
- Compare two Templates within the specified Ram Buffer and return the result.

If the comparison is successful, then RET returns ERR_SUCCESS and DATA returns the intelligent update result.

Otherwise, RET returns ERR_VERIFY.

COMMAND AND RESPONSE

PREFIX	0xAA55	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0062	
LEN	4	
DATA	2 bytes	The first RamBuffer ID to be compared
	2 bytes	The second RamBuffer ID to be compared
PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0062	
LEN	2	
RET	Result Code	
DATA	No data	

Example - Compare the fingerprint templates in RamBuffer0 and RamBuffer1 1:1, result: Match successful

Host command: 55 AA 00 00 62 00 04 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 66 01

Module response: AA 55 01 00 62 00 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 64 01

IDENTIFY 1:N WITHIN SPECIFIED ID RANGE (CMD_SEARCH 0X0063)

FUNCTION

Perform a 1:N comparison between the template in the specified Ram Buffer and all registered fingerprint templates within the specified search range (starting template ID ~ending template ID) and return the result.

SEQUENCE

- If the specified Ram Buffer ID is invalid, an error code ERR_INVALID_BUFFER_ID is returned.
- If the specified search range is invalid, return error code ERR_INVALID_BUFFER_ID.
- If there are no registered Templates, return the error code ERR_ALL_TMPL_EMPTY.
- Compare the Template in the specified Ram Buffer with all registered templates and return the result.

If the search is successful, RET returns ERR_SUCCESS and returns the searched template ID and smart update results in the DATA field. Otherwise, RET returns ERR_IDENTIFY.

COMMAND AND RESPONSE

PREFIX	0xAA55	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0063	
LEN	6	
DATA	2 bytes	RamBuffer ID
	2 bytes	The starting Template ID to be searched
	2 bytes	The ending Template ID to be searched
PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0063	

LEN	Success: 5, Failure: 2	
RET	Result Code	
DATA	3 bytes	When successful: Template ID (2 bytes) + Smart update result (1 byte)

Example - Fingerprint comparison of the template temporarily stored in RamBuffer0 within the range of 1-200, comparison result ID=8

Host command: 55 AA 00 00 63 00 06 00 00 00 01 00 C8 00 00 00 00 00 00 00 00 00 00 00 00 00 31 02

Module response: AA 55 01 00 63 00 05 00 00 00 08 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 71 01

COMPARE SPECIFIED RAM BUFFER TEMPLATE WITH LIBRARY TEMPLATE BY ID (CMD_VERIFY 0X0064)

FUNCTION

Perform a 1:1 comparison between the template in the specified Ram Buffer and the template with the specified ID in the library, and return the result.

SEQUENCE

- If the specified Template ID is invalid, an error code ERR_INVALID_TMPL_NO is returned.
- If the specified Ram Buffer ID is invalid, an error code ERR_INVALID_BUFFER_ID is returned.
- If there is no Template registered with the specified ID, return error code ERR_TMPL_EMPTY.
- Perform a comparison between the template in the specified Ram Buffer and the template with the specified ID in the library, and return the result.

If the comparison is successful: then RET returns ERR_SUCCESS and DATA returns the Template number and smart update result. Otherwise: RET returns ERR_VERIFY.

COMMAND AND RESPONSE

PREFIX	0xAA55	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0064	
LEN	4	
DATA	2 bytes	Template ID to be compared
	2 bytes	RamBuffer ID
PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0064	
LEN	Success: 5, Failure: 2	
RET	Result Code	
DATA	3 bytes	When successful: Template ID (2 bytes) + Smart update result (1 byte)

Example - Perform a 1:1 verification of the fingerprint template in RamBuffer0 with the fingerprint numbered 8 in the database

Host command: 55 AA 00 00 64 00 04 00 08 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 6F 01

Module response: AA 55 01 00 64 00 05 00 00 00 08 00 01 00 00 00 00 00 00 00 00 00 00 00 72 01

SET MODULE SERIAL NUMBER (CMD_SET_MODULE_SN 0X0008)

FUNCTION

The module receives the module serial number (Module SN) sent from the host and saves it in the module. The Module SN is 16 bytes.

SEQUENCE

- Host sends a command packet to make the Target enter the waiting state for receiving the data (Module SN).

The DATA field in this command packet has already set the length of the next command packet to be sent.

-Target checks the accuracy of the received command packet.

If incorrect: Return the error code and end processing.

If the size of the data to be received is incorrect: Return ERR_INVALID_PARAM.

If correct: In order to notify that it has entered the data (Module SN) reception waiting state, it sends a response packet to the HOST, and enters the data (Module SN) reception waiting state.

- After receiving the response packet, the Host sets the Module SN in the command packet and sends it to the Target.

- After receiving the command packet, the Target sets the Module SN to the module and returns its result.

COMMAND AND RESPONSE

PREFIX	0xAA55
SID	Source Device ID
DID	Destination Device ID
CMD	0x0008
LEN	2
DATA	16 (Module SN block length)
PREFIX	0x55AA
SID	Source Device ID
DID	Destination Device ID
RCM	0x0008
LEN	2
RET	Result Code
DATA	No data
Command packet	
PREFIX	0xA55A
SID	Source Device ID
DID	Destination Device ID
CMD	0x0008
LEN	16 (Module SN block length)
DATA	Module SN (16 bytes)
PREFIX	0x5AA5
SID	Source Device ID
DID	Destination Device ID
RCM	0x0008
LEN	2

RET	Result Code
DATA	No data

For example:

None

GET MODULE SERIAL NUMBER (CMD_GET_MODULE_SN 0X0009)

FUNCTION

Send the Module SN saved in the module to the Host.

SEQUENCE

- In the form of a command response packet, specify the size of the Module SN to be received by the HOST as the response data and respond.
- The Module SN stored in the module is sent using the response packet.

COMMAND AND RESPONSE

PREFIX	0xAA55
SID	Source Device ID
DID	Destination Device ID
CMD	0x0009
LEN	0
DATA	No data
PREFIX	0x55AA
SID	Source Device ID
DID	Destination Device ID
RCM	0x0009
LEN	4
RET	Result Code
DATA	Success: Next data response packet length (Module SN Size(16)) Failure: Error code
When successful: Target sends response packet to HOST	
PREFIX	0x5AA5
SID	Source Device ID
DID	Destination Device ID
RCM	0x0009
LEN	Module SN Size(16)
RET	Result Code (ERR_SUCCESS)
DATA	Module SN (16 bytes)

For example:

None

GET REGISTERED ID LIST (CMD_GET_ENROLLED_ID_LIST 0X0049)

FUNCTION


```
A5 5A 01 00 49 00 43 00 00 00 FE FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF 07 00 00 00 55 3E
```

...

COMMAND AND RESPONSE

PREFIX	0x55AA
SID	Source Device ID
DID	Destination Device ID
RCM	0x00FF
LEN	2
RET	Result Code (ERR_SUCCESS)
DATA	-

PRECAUTIONS

1. The CMD_GENERATE command generates Template Data from ImageBuffer.
Therefore, before calling this command, you need to call the CMD_GET_IMAGE command in advance to save the image in the ImageBuffer.
2. Call the CMD_VERIFY, CMD_SEARCH, CMD_GENERATE, CMD_MERGE, CMD_MATCH commands, and the images saved in the ImageBuffer will be cleared.
3. Templates saved in Ram Buffer2 will be cleared when calling CMD_SEARCH, CMD_VERIFY, CMD_GENERATE, CMD_STORE_CHAR, CMD_DEL_CHAR, CMD_GET_EMPTY_ID, CMD_GET_STATUS, GET_BROKEN_ID, CMD_GETN_ENROLL_COUNT commands.

Therefore, do not use Ram Buffer2 except for registration.