



Unit-Fingerprint

Control Protocol

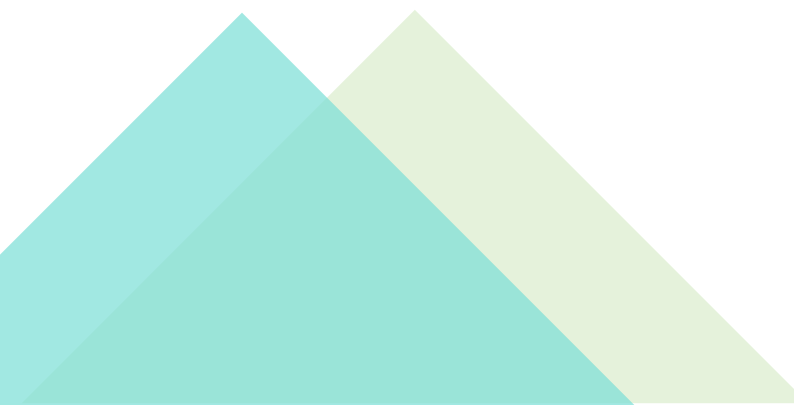


Table of contents

1、Communication Protocol Structure	4
1.1 Communication Protocol Parameters	4
1.1.1 Parameters between the internal MCU and the Fingerprint Module	4
1.1.2 Parameters of the External Communication Interface	4
1.2 Data Packet Format	4
1.2.1 Command Packet	4
1.2.2 Data Packet	5
1.2.3 Response\Return Packet	5
2、Software Development Guide	5
2.1 Parameter Table (Internal Fingerprint Module)	5
2.2 ROM and Sensor Driver	10
2.3 Device Address	10
2.4 Unit-Fingerprint Operating Status	10
2.4.1 Timed Sleep Mode	10
2.4.2 Enabling Mode	10
2.5 Unit-Fingerprint Internal Fingerprint Module Off Status	11
3、Command Set Definition	11
3.1 General Command Set	11
3.1.1 Acquire Image for Verification--PS_GetImage	12
3.1.2 Acquire Image for Enrollmen--PS_GetEnrollImage	12
3.1.3 Generate Characteristics--PS_GenChar	13
3.1.4 Accurately Compare Two Fingerprint Features--PS_Match	13
3.1.5 Search Fingerprint--PS_Search	14
3.1.6 Merge Features (Generate Template)--PS_RegModel	14
3.1.7 Storage Template--PS_StoreChar	15
3.1.8 Reading Template--PS_LoadChar	15
3.1.9 Upload Image--PS_UpImage	16
3.1.10 Delete Template--PS_DeletChar	17
3.1.11 1 Clear Fingerprint Database--PS_Empty	17
3.1.12 Write System Register--PS_WriteReg	17
3.1.13 Read Basic System Parameters--PS_ReadSysPara	18
3.1.14 Sampling Random Numbers--PS_GetRandomCode	19
3.1.15 Read Flash Information Page PS_ReadINPage	19
3.1.16 Write Notepad--PS_WriteNotepad	20
3.1.17 Read Notepad--PS_ReadNotepad	20
3.1.18 Read the Number of Valid Templates--PS_ValidTemplateNum	21
3.1.19 Read Index Table--PS_ReadIndexTable	21
3.1.20 Get the Chip 's Unique Serial Number--PS_GetChipSN	22
3.1.21 Handshake Command--PS_HandShake	22
3.1.22 Check Sensor--PS_CheckSensor	22
3.1.23 LED Control Light Command--PS_ControlBLN	23
3.1.24 Get Image Information Command--PS_GetImageInfo	24
3.1.25 Search Current Fingerprint Instruction--PS_SearchNow	25
3.1.26 Special Upload Template--PS_UpTemplet	25
3.1.27 Special Download Template--PS_DownTemplet	26
3.1.28 Set sleep time--PS_SetSleepTime	26
3.1.29 Get sleep time--PS_GeSleepTime	27
3.1.30 Set working mode--PS_SetWorkMode	27
3.1.31 Get the working mode--PS_GetWorkMode	28
3.1.32 Activate internal fingerprint module--PS_ActivateFingerprintModule	28
3.1.33 Get fingerprint module working status--PS_GetFingerprintModuleStatus	29
3.1.34 Save configuration information to flash--PS_SaveConfigurationToFlash	29
3.1.35 Get STM32 firmware version--PS_Get FirmwareVersion	30
3.2 Modular Instruction Set	30

3.2.1 Cancel Command--PS_Cancel	30
3.2.2 Automatic registration template PS_AutoEnroll	31
3.2.3 Automatic fingerprint verification--PS_AutoIdentify	33
4. Functional Demonstration Example	35
4.1 UART Command Packet Processing Process	35
4.2 UART Data Packet Sending Process	35
4.3 UART Data Packet Reception Process	36
4.4 Fingerprint Registration Process	37
4.5 Fingerprint Search Process	39
4.6 Master Controller Loads a Fingerprint Feature or Template for Accurate Comparison	40
4.7 Special Upload Template Process	41
4.8 Special Download Template Process	42
5. Return type Table	43
6. RC Calculation Method Example (C/C++)	45
6.1 Verification	45
6.2 Calculation	45

1、Communication Protocol Structure

1.1 Communication Protocol Parameters

1.1.1 Parameters between the internal MCU and the Fingerprint Module

Half-duplex asynchronous serial communication is adopted.

The default baud rate is 57600bps. (Modification is prohibited)

Data format: 8 data bits (LSB first) and 2 stop bits, with no parity bit.

Communication interface between the internal fingerprint module and the STM32 chip.

1.1.2 Parameters of the External Communication Interface

Half-duplex asynchronous serial communication is adopted.

The baud rate is 115200bps.

Data format: 8 data bits (LSB first) and 1 stop bit, with no parity bit.

The external communication interface of the ST chip, i.e., the user control interface.

1.2 Data Packet Format

1.2.1 Command Packet

Header	Chip Address	Identifier	Length	Command	Para 1	...	Para N	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	N bytes		N bytes	2 bytes
Parameter Description	Header	Packet header: each packet of data starts with 0xEF01						
	Chip Address	Device address: 0xFFFFFFFF, All data packets must contain this address.						
	Identifier	Represent different data packets: 0x01: Indicates a command packet. 0x02: Indicates a data packet, with subsequent data packets to follow. 0x08: Indicates a data packet, with no subsequent packets. 0x07: Indicates a response packet.						
	Length	Packet length = total number of bytes from the packet length to the checksum (commands, parameters, or data), including the checksum but excluding the bytes of the packet length itself.						
	Command	Command Parameter						
	Parameter	Different commands have different parameters.						
	Sum	The checksum is the sum of all bytes from the packet identifier to the checksum, including the packet identifier but excluding the bytes of the checksum itself. Any carry beyond 2 bytes is ignored.						

1.2.2 Data Packet

Header	Chip Address	Identifier	Length	Data	Sum
2 bytes	4 bytes	1 byte	2 bytes	N bytes	2 bytes
Parameter Description	Header	Packet header: each packet of data starts with 0xEF01			
	Chip Address	Device address: 0xFFFFFFFF, All data packets must contain this address.			
	Identifier	Represent different data packets: 0x01: Indicates a command packet. 0x02: Indicates a data packet, with subsequent data packets to follow. 0x08: Indicates a data packet, with no subsequent packets. 0x07: Indicates a response packet.			
	Length	Packet length = total number of bytes from the packet length to the checksum (commands, parameters, or data), including the checksum but excluding the bytes of the packet length itself.			
	Data	Transmitted Data			
	Sum	The checksum is the sum of all bytes from the packet identifier to the checksum, including the packet identifier but excluding the bytes of the checksum itself. Any carry beyond 2 bytes is ignored.			

1.2.3 Response\Return Packet

Header	Chip Address	Identifier	Length	Acknowledge Code	Return	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 bytes	N bytes	2 bytes
Parameter Description	Header	Packet header: each packet of data starts with 0xEF01				
	Chip Address	Device Address 0xFFFFFFFF, All data packets must contain this address.				
	Identifier	Represent different data packets: 0x01: Indicates a command packet. 0x02: Indicates a data packet, with subsequent data packets to follow. 0x08: Indicates a data packet, with no subsequent packets. 0x07: Indicates a response packet.				
	Length	Packet length = total number of bytes from the packet length to the checksum (commands, parameters, or data), including the checksum but excluding the bytes of the packet length itself.				
	Acknowledge Code	Feedback the status of command execution				
	Return	Feedback the parameters of command execution				
	Sum	The checksum is the sum of all bytes from the packet identifier to the checksum, including the packet identifier but excluding the bytes of the checksum itself. Any carry beyond 2 bytes is ignored.				

Notes:

1. Data packets cannot enter the execution process independently; they must follow a command packet or a response packet.
2. The data packets used for download (from the host computer to the module) or upload (from the module to the host computer) have the same format.
3. For multi-byte data, the high byte is placed first and the low byte last (e.g., the 2-byte sequence "00 06" represents the value 0006, not 0600).
4. Commands can only be sent from the host computer to the module, and the module responds to the host computer accordingly.

2、 Software Development Guide

2.1 Parameter Table (Internal Fingerprint Module)

- The content of the Parameter Table consists of the basic parameters for protocol and algorithm operation. Since the entire software system relies on the content of the Parameter Table, understanding and properly configuring the Parameter Table is crucial for the correct use of the chip.
- The Parameter Table is configured by the DSP initialization program during the first power-on and stored in the

system parameter storage area of FLASH. For each subsequent power-on, the SOC initialization program must first load the Parameter Table into RAM and initialize the system registers based on the content of the Parameter Table; the length of the Parameter Table is 64 words (128 bytes).

- The structure of the Parameter Table is as shown in the table below:

The initial content of the Parameter Table is configured by the ROM-resident program or user program during the first power-on of the system.

Type	Num	Name	Length (word)	Content and Default Values	Notes
PART1	1	SSR	1	0	
	2	SensorType	1		
	3	DataBaseSize	1	Auto-identification based on FLASH type	
PART2	4	SecurLevel	1	3	5 Levels
	5	DeviceAddress	2	0xffffffff	
	6	CFG_PktSize	1	1	These 8 registers are the System Configuration Table.
	7	CFG_BaudRate	1	6	
	8	CFG_VID	1		
	9	CFG_PID	1		
	10		1		
	11		1		
	12		1		
	13		1		
	14	ProductSN	4	ASCII	Device Descriptor
	15	SoftwareVersion	4	ASCII	
	16	Manufacturer	4	ASCII	
	17	SensorName	4	ASCII	
	18	PassWord	2	00000000H	Not Enabled
	19	JtagLockFlag	2	00000000H	
	20	SensorInitEntry	1	Entry Address	
	21	SensorGetImageEntry	1	Entry Address	
	22		27		
PART3	23	ParaTableFlag	1	0x1234	

- The Parameter Table is located in Page 1 of the system parameter storage area;
- During chip power-on, the Parameter Table is loaded from flash to RAM, and its structure and order remain unchanged;
- Detailed Explanation of the Parameter Table:

1) Status Register

SSR

Reset Value: 0x0000

Length: 1 word

Attribute: Read-Only

Purpose: System status indication

Read Command: PS_ReadSysPara (see Instruction Description for details)

Format: See following table for details

[15:4]	[3]	[2]	[1]	[0]
Reserved	ImgBufStat	PWD	Pass	Busy

Notes:

- Busy: 1 Bit, Set to '1' indicates the system is executing a command; '0' indicates the system is idle;
- Pass: 1 Bit, Set to '1' indicates fingerprint verification passed;
- PWD: 1 Bit, Set to '1' indicates the device handshake password verification has passed;
- ImgBufStat: 1 Bit, Set to '1' indicates that the fingerprint image buffer contains a valid fingerprint image.

- 2) Sensor Type SensorType**
- Reset Value: 0x0000
- Length: 1 word
- Attribute: Read-Only
- Purpose: Indicates the sensor driver type
- Read Command: PS_ReadSysPara, see Instruction description for details
- 3) Database Size DataBaseSize**
- Reset Value: According to FLASH
- Length: 1 word
- Attribute: Read-Only
- Purpose: Fingerprint database capacity indication
- Read Command: PS_ReadSysPara, see Instruction description for details
- 4) Security Level SecurLeve**
- Reset Value: 0x0003
- Length: 1 word
- Attribute: Read-Only
- Purpose: Security Level Indication
- The system sets the comparison threshold based on this value
- Read Command: PS_ReadSysPara, see Instruction description for details
- Setting Command: PS_WriteRegsee Instruction description for details
- Five Levels:
- | | |
|---------|---------|
| 1:Leve1 | Lowest |
| 2:Leve2 | |
| 3:Leve3 | |
| 4:Leve4 | |
| 5:Leve5 | Highest |
- 5) Device Address DeviceAddress (Modification is disabled for STM32)**
- Reset Value: 0xffffffff
- Length: 2 word
- Attribute: Read-Only
- Purpose: The system only receives address-matching command packets/data packets
- Read Command: PS_ReadSysPara, see Instruction description for details
- Setting Command: PS_SetChipAddr, see Instruction description for details
- 6) Data Packet Size CFG_PktSize**
- Reset Value: 0x0001
- Length: 1 word

Attribute:	Read-Write
Purpose:	During data transmission, the system sets the Length of a single data packet based on this value
Read Command:	PS_ReadSysPara, see Instruction description for details
Setting Command:	PS_WriteReg, see Instruction description for details
7) Baud Rate Coefficient	CFG_BaudRate (Modification is disabled for STM32)
Reset Value:	0x0006
Length:	1 word
Attribute:	Read-Only
Purpose:	Determine UART Baud Rate = this value multiplied by 9600
Read Command:	PS_ReadSysPara, see Instruction description for details
Setting Command:	PS_WriteReg, see Instruction description for details
8) USB ID	CFG_VID
Reset Value:	0x0453
Length:	1 word
Attribute:	Read-Only
Purpose:	USB Embedded ProtocoVID
Read Command:	PS_ReadINFpage, see Instruction description for details
9) USB ID	CFG_PID
Reset Value:	0x9005
Length:	1 word
Attribute:	Read-Only
Purpose:	USB Embedded ProtocoVID
Read Command:	PS_ReadINFpage, see Instruction description for details
10) Product SN	ProductSN
Reset Value:	First Power-On Initialization Value
Length:	4 words
Attribute:	Read-Only
Purpose:	Indicates the Product Model
Read Command:	PS_ReadINFpage, see Instruction description for details
11) Software Version	SoftwareVersion
Reset Value:	First Power-On Initialization Value
Length:	4 words
Attribute:	Read-Only
Purpose:	Indicates Software Version
Read Command:	PS_ReadINFpage, see Instruction description for details
12) Manufacturer Name	Manufacturer
Reset Value:	First Power-On Initialization Value
Length:	4 words
Attribute:	Read-Only

Purpose:	Indicates Manufacturer Name
Read Command:	PS_ReadINFpage, see Instruction description for details
13) Sensor Name	SensorName
Reset Value:	First Power-On Initialization Value
Length:	4 words
Attribute:	Read-Only
Purpose:	Indicates Sensor Name
Read Command:	PS_ReadINFpage, see Instruction description for details
14) Password	PassWord
Reset Value:	0x00000000
Length:	2 words
Attribute:	Read-Write
Purpose:	Handshake Password, the system responds only if the password is valid
Read Command:	PS_ReadINFpage, see Instruction description for details
Setting Command:	PS_SetPwd, see Instruction description for details
15) JTAG Lockout Flag	JtagLockFlag
Reset Value:	0x00000000
Length:	2 words
Attribute:	Read-Only
Purpose:	Writing a specific value during the first power-on will disable the JTAG port
Read Command:	PS_ReadINFpage, see Instruction description for details
16) Sensor Initialization Entry	SensorInitEntry
Reset Value:	Reserved
Length:	1 word
Attribute:	Read-Only
Purpose:	The system invokes the sensor initialization program based on this value, Reserved
Read Command:	PS_ReadINFpage, see Instruction description for details
17) Sensor Image Acquisition Entry	SensorGetImageEntry
Reset Value:	Reserved
Length:	1 word
Attribute:	Read-Only
Purpose:	The system invokes the sensor image acquisition based on this value, Reserved
Read Command:	PS_ReadINFpage, see Instruction description for details
18) Parameter Table Valid Flag	ParaTableFlag
Reset Value:	0x1234
Length:	1 word
Attribute:	Read-Only
Purpose:	If the value of this field is 0x1234, it indicates that the Parameter Table has been initialized; if the value of this field is 0x0204, it indicates that the system only initializes the PART1 section of the Parameter Table; if the value of this field is any other value, the system will initialize the Parameter Table.

2.2 ROM and Sensor Driver

ROM embeds a complete fingerprint recognition algorithm.

2.3 Device Address

The default address of the chip is 0xFFFFFFFF and cannot be modified. The address field of the data packet must match this address for the command packet/data packet to be received by the system.

2.4 Unit-Fingerprint Operating Status

2.4.1 Timed Sleep Mode

(1) Wakeup

- 1、Sleep Status, activated by finger press.
- 2、Awakened via command (see command 3.1.32 for details)

(2) Sleep

1. Timer Function and Configuration

The STM32 uses an internal timer for periodic timing, and the timing duration can be configured via command 3.1.28. When the UART receives data, the timer will be reset and restart timing.

2. Automatic Polling Mechanism After Timing Expires

If the timing duration elapses without receiving UART data during this period, the STM32 will actively query the fingerprint module for the finger press status, up to three times:

If all three queries return "no finger pressed", the fingerprint module will be turned off;

If any query returns "finger press detected", the query will be immediately terminated, and the timer will be reset to restart timing.

3. Command Processing Restrictions

During the STM32's finger status query period, only commands with command codes in the range of 0xD0 ~ 0xD7 are allowed to be processed. All other commands will be rejected, and an error code of 0xFB will be returned (see the "Return Type Table" for details).

4. Processing and Response to Abnormal Commands

During the STM32's automatic finger status query period, if a command not within the allowed range (0xD0 ~ 0xD7) is received, the STM32 will first return an error code of 0xFB (see the "Return Type Table" for details), then exit the current query process, and reset the timer to restart timing.

2.4.2 Enabling Mode

Under this mode, the device remains in the enabled state at all times.

Notes:

In Timed Sleep Mode, a return packet is sent after successful activation. The format corresponds to the response packet in 1.2.3, with the confirmation code being 0xFF (the return data packet is 0xEF 0x01 0xFF 0xFF 0xFF 0xFF 0x07 0x00 0x03 0xFF 0x01 0x09).

2.5 Unit-Fingerprint Internal Fingerprint Module Off Status

Under this status, commands with command codes in the range of 0xD0~0xD7 can still be processed normally. To make the internal fingerprint module enter the operating state, its internal fingerprint module must first be activated through passive or active means.

3、Command Set Definition

After connecting the necessary peripheral circuits to the Fingerprint Module SOC, a complete fingerprint recognition module can be formed. The module remains in a subordinate position (Slave mode) at all times, and the Host (master controller) needs to make the module perform various functions through different commands. The Host's commands, the module's responses, and data exchange are all carried out via data packets in a specified format. The Host must encapsulate the commands or data to be sent according to the following format, and must also parse the received data packets in the following format.

Notes: If the confirmation code returned by a command is 0xFE, it indicates that the internal fingerprint module is not activated. The fingerprint module must be activated first, and the specific activation steps can be referred to in Section 2.4.

3.1 General Command Set

Num	Command	Name	Description
1	0x01	PS_GetImage	Acquire Image for Verification
2	0x29	PS_GetEnrollImage	Acquire Image for Enrollment
3	0x02	PS_GenChar	Generate a fingerprint feature file from the original image and store it in the template buffer
4	0x03	PS_Match	Accurately compare the feature files or templates in the template buffer
5	0x04	PS_Search	Use the feature files in the template buffer to search the entire or partial fingerprint database
6	0x05	PS_RegModel	Fuse the feature files and generate a template
7	0x06	PS_StoreChar	Store the template files in the template buffer into the flash fingerprint database
8	0x07	PS_LoadChar	Read a template from the flash fingerprint database into the template buffer
9	0x0A	PS_UplImage	Upload original image
10	0x0C	PS_DeletChar	Delete a template file from the Flash Fingerprint database
11	0x0D	PS_Empty	Erase the Flash Fingerprint database
12	0x0E	PS_WriteReg	Write to the SOC system registers
13	0x0F	PS_ReadSysPara	Read the system basic parameters
14	0x14	PS_GetRandomCode	Sample random numbers
15	0x16	PS_ReadINFpage	Read the Content of the Flash Information Page
16	0x18	PS_WriteNotepad	Write notebook
17	0x19	PS_ReadNotepad	Read notebook
18	0x1D	PS_ValidTemplateNum	Read the valid template count
19	0x1F	PS_ReadIndexTable	Read the index table
20	0x34	PS_GetChipSN	Obtain the chip's unique serial number
21	0x35	PS_HandShake	Handshake command
22	0x36	PS_CheckSensor	Calibrate the sensor
23	0x3C	PS_ControlBLN	Breathing light command
24	0x3D	PS_GetImageInfo	Acquire image information

Num	Command	Name	Description
25	0x3E	PS_SearchNow	Search current fingerprint
26	0x7A	PS_UpTemplet	Upload the special template
27	0x7B	PS_DownTemplet	Download the special template
28	0xD0	PS_SetSleepTime	Set sleep time
29	0xD1	PS_GetSleepTime	Acquire sleep time
30	0xD2	PS_SetWorkMode	Set operating mode
31	0xD3	PS_GetWorkMode	Acquire operating mode
32	0xD4	PS_ActivateModule	Activate fingerprint module
33	0xD5	PS_GetFigurationModuleStatus	Acquire working status of the fingerprint module
34	0xD6	PS_SaveConfigurationToFlash	Save the configuration information to flash
35	0xD7	PS_GetFirmwareVersion	Acquire the STM32 firmware version

3.1.1 Acquire Image for Verification--PS_GetImage

- Function Description:

When verifying a fingerprint, detect the finger. If a finger is detected, acquire the fingerprint image and store it in the image buffer. Return a Confirmation Code indicating statuses such as "successful enrollment" or "no finger detected".

- Input Parameters: none

- Return Parameters: Confirmation Word

- Command Code: 01H

- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	01H	0005H

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

Notes: Confirmation Code=00H indicates successful image acquisition;

Confirmation Code=01H indicates an error in packet reception;

Confirmation Code=02H indicates no finger on the sensor;

Confirmation Code=03H indicates unsuccessful image acquisition;

sum refer to Checksum.

3.1.2 Acquire Image for Enrollment--PS_GetEnrollImage

- Function Description:

When enrolling a fingerprint, detect the finger. If a finger is detected, acquire the fingerprint image and store it in the image buffer. Return a Confirmation Code indicating statuses such as "successful enrollment" or "no finger detected".

- Input Parameters: none

- Return Parameters: Confirmation Word

- Command Code: 29H

- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	29H	002DH

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

Notes: Confirmation Code=00H indicates successful image acquisition;
 Confirmation Code=01H indicates an error in packet reception;
 Confirmation Code=02H indicates no finger on the sensor;
 Confirmation Code=03H indicates unsuccessful image acquisition;
 sum refer to Checksum.

3.1.3 Generate Characteristics--PS_GenChar

- Function Description: Generates a fingerprint feature file from the original image in the image buffer and stores it in the template buffer.
- Input Parameters: BufferID (positive integer, i.e. 1, 2, ...)
- Return Parameters: Confirmation Word
- Command Code: 02H
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Buffer ID	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes
0xEF01	xxxx	01H	0004H	02H	BufferID	sum

Notes: During registration , BufferID indicates the number of times the finger is pressed; in other cases , BufferID has a corresponding default value.

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

Notes: Confirmation Code = 00H means the feature is generated successfully;
 Confirmation Code = 01H means there is an error in receiving the packet;
 Confirmation Code = 06H means that the image is too cluttered to generate features;
 Confirmation Code = 07H means the fingerprint image is normal, but there are too few feature points to generate a feature;
 Confirmation Code = 08H means that during the registration process, you are required to enter a similar finger area each time, and the current feature is not similar to the previous feature; (This function is disabled by default)
 Confirmation Code = 0aH indicates feature merging failed;
 Confirmation Code = 15 H means that there is no valid original image in the image buffer and the image cannot be generated ;
 Confirmation Code = 28H means that during the registration process, you are required to enter a different finger area each time. If the area of the current feature overlaps too much with the previous feature, (this function is disabled by default)
 Sum refer to Checksum.

3.1.4 Accurately Compare Two Fingerprint Features--PS_Match

- Function Description: Accurately compare the feature files in the template buffer.
- Input Parameters: none
- Return Parameters: Confirmation Word, Comparison Score

- Command Code: 03H
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	03H	0007H

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Score	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes	2 bytes
0xEF01	xxxx	07H	0005H	xxH	xxxxH	sum

Notes: Confirmation Code = 00H indicates fingerprint matching;

Confirmation Code = 01H means there is an error in receiving the packet;

Confirmation Code=08H means the fingerprint does not match;

Sum refer to Checksum.

3.1.5 Search Fingerprint--PS_Search

- Function Description: Searches the entire or partial fingerprint database using the signature file in the template buffer. If found, returns the page number. This function is supported when the encryption level is set to 0 or 1.
- Input Parameters: BufferID (default is 1),
StartPage (starting page),
PageNum (page number)
- Return Parameters: Confirmation Word,
Page Number (matching fingerprint template),
Score (MatchScore)
- Command Code: 04H
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Buffer ID	Para	Para	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes	2 bytes	2 bytes
0xEF01	xxxx	01H	0008H	04H	BufferID	StartPage	PageNum	sum

Notes: BufferID defaults to 1, and the fingerprint template in the template buffer is used to search the entire or partial fingerprint library.

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Page	Score	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes	2 bytes	2 bytes
0xEF01	xxxx	07H	07H	xxH	PageID	MatchScore	sum

Notes: Confirmation Code = 00H means searched;

Confirmation Code = 01H means there is an error in receiving the packet;

Confirmation Code = 09H means no search results found; the page number and score are 0.

Confirmation Code = 17H means there is residual fingerprint or the finger has not moved between two captures;

Sum refer to Checksum.

3.1.6 Merge Features (Generate Template)--PS_RegModel

- Function Description: Generate a template by fusing feature files.
- Input Parameters: none

- Return Parameters: Confirmation Word
- Command Code: 05H
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	05H	0009H

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

Notes: Confirmation Code = 00H indicates a successful merge;

Confirmation Code = 01H means there is an error in receiving the packet;

Confirmation Code = 0 a H means the merge failed ;

Confirmation Code = 3bH means the template quality is poor and the merge fails;

Sum refer to Checksum.

3.1.7 Storage Template--PS_StoreChar

- Function Description: Save the template file in the template buffer to the flash database location of PageID number. This function is supported when the encryption level is set to 0 or 1.
- Input Parameters: BufferID (default is 1), PageID (fingerprint library location number)
- Return Parameters: Confirmation Word
- Command Code: 06H
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Buffer ID	Position number	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes	2 bytes
0xEF01	xxxx	01H	0006H	06H	BufferID	PageID	sum

Notes: BufferID defaults to 1.

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

Notes: Confirmation Code = 00H means the storage is successful;

Confirmation Code = 01H means there is an error in receiving the packet;

Confirmation Code = 0bH means PageID exceeds the range of fingerprint database;

Confirmation Code = 18H means an error occurred while writing to FLASH;

Confirmation Code=35H indicates illegal data;

Sum refer to Checksum.

3.1.8 Reading Template--PS_LoadChar

- Function Description: Read the fingerprint template with the specified ID number in the flash database into the template buffer.
- Input Parameters: BufferID (default is 2), PageID (fingerprint library template number)
- Return Parameters: Confirmation Word

- Command Code: 07H
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Buffer ID	Page Number	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes	2 bytes
0xEF01	xxxx	01H	0006H	07H	BufferID	PageID	sum

Notes: BufferID defaults to 2.

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

Notes: Confirmation Code = 00H means the reading is successful;

Confirmation Code = 01H means there is an error in receiving the packet;

Confirmation Code = 0 b H means PageID is out of the range of fingerprint database ;

Confirmation Code = 01H means that the reading is wrong or the template is invalid ;

Sum refer to Checksum.

3.1.9 Upload Image--PS_UpImage

- Function Description: Upload the data in the image buffer to the main control.
- Input Parameters: none
- Return Parameters: Confirmation Word
- Command Code: 0AH
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	0AH	000EH

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

Notes: Confirmation Code = 00H means that the subsequent data packets will be sent;

Confirmation Code = 01H means there is an error in receiving the packet;

Confirmation Code = 0 f H means that subsequent data packets cannot be sent ;

sum refer to Checksum.

- Subsequent data packets are sent after the response.

Header	Device Address	Identifier	Length	data	Sum
2 bytes	4 bytes	1 byte	2 bytes	N bytes	2 bytes
0xEF01	xxxx	xxH	xxxxH	xx	sum

Notes: Identifier = 02 : Data packet with subsequent packets.

Identifier = 08 : The last data packet , i.e. the end packet.

When uploading image data packets via UART , they are divided into packets according to the preset length .

- One byte contains two pixels , each pixel occupies 4 bits .

3.1.10 Delete Template--PS_DeletChar

- Function Description: Delete flash Specified in the database ID Starting with N fingerprint templates.
- Input Parameters: fingerprint library template number (PageID), number of templates to be deleted (N).
- Return Parameters: Confirmation Word
- Command Code: 0CH
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Page Number	Number of deletions	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes	2 bytes	2 bytes
0xEF01	xxxx	01H	0007H	0CH	PageID	N	sum

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

Notes: Confirmation Code = 00H means the template is deleted successfully;

Confirmation Code = 01H means there is an error in receiving the packet;

Confirmation Code = 10H means template deletion failed;

Sum refer to Checksum.

3.1.11 Clear Fingerprint Database--PS_Empty

- Function Description: Delete flash All fingerprint templates in the database.
- Input Parameters: none
- Return Parameters: Confirmation Word
- Command Code: 0DH
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	0DH	0011H

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

Notes: Confirmation Code = 00H means clearing is successful;

Confirmation Code = 01H means there is an error in receiving the packet;

Confirmation Code = 11H means clearing failed;

Sum refer to Checksum.

3.1.12 Write System Register--PS_WriteReg

- Function Description: Write module registers.
- Input Parameters: Register number, content
- Return Parameters: Confirmation Word
- Command Code: 0EH
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Register Number	Content	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	1 byte	2 bytes
0xEF01	xxxx	01H	0005H	0EH	0~13	xx	sum

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

Notes 1: Confirmation Code =00H means OK;

Confirmation Code = 01H means there is an error in receiving the packet;

Confirmation Code = 18H means an error occurred in reading or writing FLASH;

Confirmation Code =1aH means the register number is incorrect;

Confirmation Code =1bH indicates the register setting content error number;

Sum refer to Checksum.

Notes 2 : When the write system register (PS_WriteReg) instruction is executed , it first responds according to the original configuration , then modifies the system settings and records the configuration in FLASH , and then works according to the new configuration.

Register number	Register Name	Content Description
6	Packet Size Register	0 : 32 bytes 1 : 64 bytes 2 : 128 bytes 3 : 256 bytes

3.1.13 Read Basic System Parameters--PS_ReadSysPara

- Function Description : Read the basic parameters of the module (baud rate , packet size, etc.). 16 bytes store the basic communication and configuration information of the module , which is called the basic parameters of the module.
- Input Parameters: none
- Return Parameters: Confirmation Word, basic parameters (16 bytes)
- Command Code: 0FH
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	0FH	0013H

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Basic Parameter Column Table	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	16 bytes	2 bytes
0xEF01	xxxx	07H	0013H	xxH	See folowing table for structure	sum

Notes: Confirmation Code =00H means OK;

Confirmation Code = 01H means there is an error in receiving the packet;

Sum refer to Checksum.

Name	Content Description	Offset (bytes)	Size (bytes)
Status Register	System status register contents	0	2
Sensor Type	Sensor type code.	2	2

Name	Content Description	Offset (bytes)	Size (bytes)
Fingerprint database size	Fingerprint library capacity	4	2
Security Level	Score level code (1/2/3/4/5)	6	2
Device Address	32 -bit Device Address	8	4
Packet size	Packet size code: 0 : 32 bytes 1 : 62 bytes 2 : 128 bytes 3 : 256 bytes	12	2
Baud rate setting	N (Baud rate 9600*N bps)	14	2

3.1.14 Sampling Random Numbers--PS_GetRandomCode

- Function Description: Enables the chip to generate a random number and return it to the main controller.
- Input Parameters: none
- Return Parameters: Confirmation Word, Random Number
- Command Code: 14H
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	14H	0018H

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Random Numbers	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	4 bytes	2 bytes
0xEF01	xxxx	07H	0007H	xxH	xxxx	sum

Notes: Confirmation Code =00H indicates successful generation;

Confirmation Code = 01H means there is an error in receiving the packet;

Confirmation Code = 19 H means random number generation failed ;

Sum refer to Checksum.

3.1. 15 Read Flash Information Page PS_ReadINFpage

- Function Description: Read FLASH Information Page The information page (512 bytes).
- Input Parameters: none
- Return Parameters: Confirmation Word
- Command Code: 16H
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	16H	001AH

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

Notes: Confirmation Code = 00H means that the confirmation package will be sent later ;

Confirmation Code = 01H means there is an error in receiving the packet;

Sum refer to Checksum.

- Subsequent data packets are sent after the response.

Header	Chip address	Identifier	Length	data	Sum
2 bytes	4 bytes	1 byte	2 bytes	N bytes	2 bytes
0xEF01	xxxx	xxH	xxxxH	xx	sum

Notes: Identifier = 02 : Data packet with subsequent packets.

Identifier = 08 : The last data packet , i.e. the end packet.

When the UART reads the flash information page data packet , it is divided into packets and sent according to the preset length .

3.1.16 Write Notepad--PS_WriteNotepad

- Function Description: The module opens up a 256 bytes FLASH. The space is used to store user data . This storage space is called user User Notebook , which is logically divided into 8 The write notepad command is used to write the user's 32 bytes number to the specified notebook page.
- Input Parameters: Page number (0 to 7), User information (content)
- Return Parameters: Confirmation Word
- Command Code: 18H
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Page Number	User Information	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	32 bytes	2 bytes
0xEF01	xxxx	01H	0024H	18H	0~7	User content	sum

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

Notes: Confirmation Code =00H means OK;

Confirmation Code = 01H means there is an error in receiving the packet;

Confirmation Code = 1cH means that the page number of the notepad is specified incorrectly;

Confirmation Code = 18H means error in reading or writing FLASH

Sum refer to Checksum.

3.1.17 Read Notepad--PS_ReadNotepad

- Function Description: Read FLASH User Area 256 bytes data.
- Input Parameters: Page number (0-7)
- Return Parameters: Confirmation Word, User Information (User content)
- Command Code: 19H
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Page number	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes
0xEF01	xxxx	01H	0004H	19H	0~7	sum

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	User Information	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	32 bytes	2 bytes
0xEF01	xxxx	07H	0023H	xxH	User content	sum

Notes: Confirmation Code = 00H means OK;

Confirmation Code = 01H means there is an error in receiving the packet;

Confirmation Code = 1cH means that the page number of the notepad is specified incorrectly;

Sum refer to Checksum.

3.1.18 Read the Number of Valid Templates--PS_ValidTemplateNum

- Function Description: Read the number of valid templates.
- Input Parameters: none
- Return Parameters: Confirmation Word, Number of valid templates (N)
- Command Code: 1 DH
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	1DH	0021H

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Number of valid templates	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes	2 bytes
0xEF01	xxxx	07H	0005H	xxH	ValidN	sum

Notes: Confirmation Code = 00H means the reading is successful;

Confirmation Code = 01H means there is an error in receiving the packet;

Sum refers to Checksum.

3.1.19 Read Index Table--PS_ReadIndexTable

- Function Description: Read the index table of the input template .
- Input Parameters: Index Table page number , page number 0 , 1 respectively correspond to the template from 0-256 , 256-512 index , each 1 bit represents a table Version , 1 means the template of the corresponding storage area has been entered , 0 means it has not been entered.
- Return Parameters: Confirmation Word, Index Table Information (Index)
- Command Code: 1 FH
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Page Number	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes
0xEF01	xxxx	01H	0004H	1FH	0~1	sum

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Index Information	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	32 bytes	2 bytes
0xEF01	xxxx	07H	0023H	xxH	Index	sum

Notes: Confirmation Code =00H means OK;

Confirmation Code = 01H means there is an error in receiving the packet;

Confirmation Code = 0bH means that the address number when accessing the fingerprint library exceeds the fingerprint library range;

Sum refer to Checksum.

3.1.20 Get the Chip 's Unique Serial Number--PS_GetChipSN

- Function Description: Get the unique serial number of the chip.
- Input Parameters: Reserve
- Return Parameters: Confirmation Word, unique serial number (SN)
- Command Code: 34H
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Parameter	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes
0xEF01	xxxx	01H	0004H	34H	0	0039H

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Unique Serial Number	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	32 bytes	2 bytes
0xEF01	xxxx	07H	0023H	xxH	SN	sum

Notes: Confirmation Code =00H means OK;

Confirmation Code = 01H means there is an error in receiving the packet;

Sum refer to Checksum.

3.1.21 Handshake Command--PS_HandShake

- Function Description: Check whether the module is working properly.
- Input Parameters: none
- Return Parameters: Confirmation Word
- Command Code: 35H
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	35H	0039H

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

Notes: Confirmation Code =00H means OK;

Confirmation Code = 01H means there is an error in receiving the packet;

Sum refer to Checksum.

3.1. 22 Check Sensor--PS_CheckSensor

- Function Description: Check whether the sensor is working properly.

- Input Parameters: none
- Return Parameters: Confirmation Word
- Command Code: 36H
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	36H	003AH

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

Notes: Confirmation Code =00H means OK;

Confirmation Code = 01H means there is an error in receiving the packet;

Confirmation Code = 29H means the calibration sensor is wrong;

Sum refer to Checksum.

3.1. 23 LED Control Light Command--PS_ControlBLN

- Function Description: Control light commands are mainly divided into two categories : general indicator lights and colorful programmable breathing lights.
- Input Parameters: Function code , start color , end color , number of cycles
- Return Parameters: Confirmation Word
- Command Code: 3CH
- Command Packet Format:

Header	Device Address	Identifier	Length	Com Code	Function code	Start color	End Color	Cycles Times	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	1 byte	1 byte	1 byte	2 bytes
0xEF01	xxxx	01H	0007H	3CH	xxH	xxH	xxH	xxH	sum

- Auxiliary instructions :

Function code : LED Light mode control bit , 1- normal breathing light , 2- flash light , 3- normally open light , 4- normally closed light , 5- gradually open light , 6 Gradually turn off the lights , Other function codes are not applicable to this Command Packet Format:

Starting color : When set to normal breathing light , The color from off to bright is limited to ordinary breathing light (function code 01) Function , other functions When it is enabled , it is consistent with the end color , where bit 0 is the blue light control bit , bit 1 is the green light control bit , bit 2 is the red light control bit , and the 1 light Bright , set 0 light is off. 0x01 blue light on , 0x02 green light on , 0x03 cyan light on , 0x04 red light on , 0x05 purple light on , 0x06 Yellow light on , 0x07 The white light is on and 0x00 is off.

End color : When set to normal breathing light , the color from bright to off is limited to normal breathing light (function code 0x01) . For other functions , it is consistent with the starting color and the setting method is the same as the starting color.

Cycle times : Indicates the number of breathing or flashing lights . When set to 0 , it means infinite loop . When set to other values , it means a limited number of breathing . The cycle times are applicable to breathing and flashing functions , but are invalid in other functions , such as normally open , normally closed , gradual opening and gradual closing.

Header	Chip address	Identifier	Length	Command Code	Fun code	Time position	color 1	...	color5	Cycles Times	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	1 byte	1 byte	...	1 byte	1 byte	2 bytes
0xEF01	xxxx	01H	000 BH	3 CH	xxH	xxH	xxH	...	xxH	xxH	sum

- Auxiliary instructions :

Function code : 7- color programmable breathing light , other function codes are not applicable to this Command Packet Format:

Time bit : used to control the time for the light to breathe once , that is , the time from off to on and then to off . The time range of a single breath is 0. 1s~ 10.0s Left and right , use The number between 1-100 indicates that the number outside this range is invalid. 1 corresponding 0.1s The time bit is set to 100 correspond 10.0s The recommended time setting is 36 , breathing time and ordinary breathing light (function code 0x01) Same , probably About 3.6s .

Color code : 5 bytes , as shown in the following table , the color code of each byte is divided into 2 units , each unit has 4 , from high Starts with a valid bit , and 3 color control bits , each unit controls the process of a certain color light from off to on and then off. In addition , Programmed breathing light The lighting order of one cycle is from color 1 Unit 1 starts , then the color 1 Unit 2 , followed by color 2 Single Yuan 1 , and so on.

Color (1 byte)							
Unit1				Unit2			
Valid bits	Red light	Green light position	Blue light position	Valid bits	Red light	Green light position	Blue light position
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0

Valid bit : 0-this unit and all units after this unit are invalid , 1-this unit is valid;

Red light position : 0-red light off , 1-red light on;

Green light position : 0-green light off , 1-green light on;

Blue light position : 0-blue light off , 1-blue light on;

Cycle times : Indicates the number of times the breathing light is on . When set to 0 , it means infinite cycle . When set to other values , it means a limited number of breathing times.

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

Notes: Confirmation Code = 00H means the address is generated successfully;

Confirmation Code = 01H means there is an error in receiving the packet;

Confirmation Code =35H indicates illegal data;

Sum refer to Checksum.

3.1.24 Get Image Information Command--PS_GetImageInfo

- Function Description: After detection, the fingerprint image is recorded and stored in the image buffer , and the image information is returned.
- Input Parameters: none
- Return Parameters: Confirmation Word , Image Area (Percentage) , Image Quality (0 : Pass; Others : Fail)
- Command Code: 3DH
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	3DH	0041H

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Image area	Image quality	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	1 byte	2 bytes
0xEF01	xxxx	07H	0005H	xxH	xxH	xxH	sum

Notes: Confirmation Code = 00H means the image is acquired successfully;

Confirmation Code = 01H means there is an error in receiving the packet;

Confirmation Code = 02H means there is no finger on the sensor;

Confirmation Code = 03H means fingerprint image entry failed;

Sum refer to Checksum.

3.1. 25 Search Current Fingerprint Instruction--PS_SearchNow

- Function Description: Search the entire or part of the fingerprint library with the most recently extracted feature file in the template buffer. If found , return to the page Code. Such as Table 3- 1 The encryption level is set to 0 and 1 This function is supported.
- Input Parameters: Start Page (StartPage) , Page Number (PageNum)
- Return Parameters: Confirmation Word, Page Number (matching fingerprint template) , Score (MatchScore)
- Command Code: 3EH
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Start Page	Pages	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes	2 bytes	2 bytes
0xEF01	xxxx	01H	0007H	3EH	StartPage	PageNum	sum

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Image area	Image quality	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes	2 bytes	2 bytes
0xEF01	xxxx	07H	0007H	xxH	PageID	MatchScore	sum

Notes: Confirmation Code = 00H means searched;

Confirmation Code = 01H means there is an error in receiving the packet;

Confirmation Code = 09H means no search results found; at this time the page number and score are 0 ;

Confirmation Code = 31H means the function does not match the encryption level;

Sum refer to Checksum.

3.1. 26 Special Upload Template--PS_UpTemplet

- Function Description: Upload the template.
- Input Parameters: Template offset address , uploaded template size
- Return Parameters: Confirmation Word, Upload Template Size , Template Data
- Command Code: 7AH
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Template bias Move address	Upload template Board size	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes	2 bytes	2 bytes
0xEF01	xxxx	01H	0007H	7AH	xxxx	xxxx	sum

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Upload template Board size	Template data	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes	N bytes	2 bytes
0xEF01	xxxx	07H	xxxx	xxH	xxxx	xxxx	sum

Notes 1 :Confirmation Code = 00H means OK ;

Confirmation Code = 01H means there is an error in receiving the packet; sum refer to Checksum.

Notes 2 : When the uploaded template data is larger than the actual remaining template data , upload according to the actual remaining template data;

When the specified template offset address is greater than the template length , the returned uploaded template size is 0 .

3.1.27 Special Download Template--PS_DownTemplet

- Function Description: Download the template.
- Input Parameters: template offset address , downloaded template size , template data
- Return Parameters: Confirmation Word
- Command Code: 7BH
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Template bias Move address	Download template Board size	Template data	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes	2 bytes	N bytes	2 bytes
0xEF01	xxxx	01H	0007H	7BH	xxxx	xxxx	xxxx	sum

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

Notes 1 :Confirmation Code =00H means OK ;

Confirmation Code = 01H means there is an error in receiving the packet;

Sum refer to Checksum.

3.1. 28 Set sleep time--PS_SetSleepTime

- Function Description: Set the sleep time of the fingerprint module (the system default is 10 seconds)
- Input Parameters: N (time scale is seconds, range 10~254)
- Return Parameters: Confirmation Word
- Command Code: D0H
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Sleep time	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 bytes	2 bytes
0xEF01	xxxx	01H	0004H	D0H	xxxx	sum

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xx xx	sum

Notes 1: Confirmation Code =00H means OK;

Confirmation Code = 01H means there is an error in receiving the packet;

Confirmation Code = FDH indicates parameter error ;

Sum refers to Checksum.

Note 2: This time parameter is valid only in "Timed Sleep Mode" and is used to determine when the fingerprint module enters sleep mode. If no command is sent to the fingerprint module within the set time period , the module will actively detect whether a fingerprint has been pressed. This detection process is repeated three times . If no fingerprint has been pressed in all three tests , the fingerprint module will automatically shut down and enter standby mode , waiting for a fingerprint press or external command to wake it up.

If fingerprints are detected during any of the three detection processes, the timer will reset and the timing cycle will restart. Additionally, during the set period, if control commands are sent to the fingerprint module or the timing parameters are modified, the timer will also reset and the timing will resume.

Note 3 : You can use PS_SaveConfigurationToFlash to save the configuration to the internal flash and it will not be lost when the power is off.

3.1.29 Get sleep time--PS_GeSleepTime

- Function Description: Get the sleep time of the fingerprint module
- Input Parameters: none
- Return Parameters: Confirmation Word , sleep time (time scale is seconds)
- Command Code: D1H
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	D1 H	00D5H

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Sleep time	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 bytes	2 bytes
0xEF01	xxxx	07H	0004H	xxH	xx H	sum

Notes: Confirmation Code =00H means OK;

Confirmation Code = 01H means there is an error in receiving the packet;

Sum refer to Checksum.

3.1.30 Set working mode--PS_SetWorkMode

- Function Description: Set the working mode
- Input Parameters: Working mode (0: timed sleep mode, 1: on mode)
- Return Parameters: Confirmation Word
- Command Code: D2H
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Working Mode	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 bytes	2 bytes
0xEF01	xxxx	01H	0004H	D2 H	xxH	sum

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

Notes 1 : Confirmation Code =00H means OK;

Confirmation Code = 01H means there is an error in receiving the packet;

Confirmation Code = FD H indicates parameter error ;

Sum refers to Checksum.

Note 2: "Timed Sleep Mode" means the fingerprint module periodically determines whether to enter sleep mode based on its current operating status . In this mode , the system periodically queries the module's status to determine whether to disable fingerprint recognition. (For information on when to enter sleep mode , see the instructions for setting the sleep time command .) "On Mode" means the fingerprint module is always active and does not automatically enter sleep mode.

Notes 3 : Can be saved to internal flash through PS_SaveConfigurationToFlash, and will not be lost when power is off.

3.1.31 Get the working mode--PS_GetWorkMode

- Function Description: Get working mode
- Input Parameters: none
- Return Parameters: Working Mode
- Command Code: D3H
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	D3H	00D7H

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Working Mode	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 bytes	2 bytes
0xEF01	xxxx	07H	0004H	xxH	xxxx	sum

Notes: Confirmation Code =00H means OK;

Confirmation Code = 01H means there is an error in receiving the packet;

Sum refer to Checksum.

3.1.32 Activate internal fingerprint module--PS_ActivateFingerprintModule

- Function Description: Actively activate the Unit-Fingerprint internal fingerprint module (if the fingerprint module is not turned on, the fingerprint module will be turned on)
- Input Parameters: none
- Return Parameters: Confirmation Word
- Command Code: D4H
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	D4 H	00D8H

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07 H	0003H	xxxx	sum

Notes: Confirmation Code =00H means OK;

Confirmation Code = 01H means there is an error in receiving the packet;

Sum refer to Checksum.

3.1.33 Get fingerprint module working status--PS_GetFingerprintModuleStatus

- Function Description: Get the fingerprint module working status
- Input Parameters: none
- Return Parameters: Confirmation Word , fingerprint module working status (0 means disabled, 1 means enabled)
- Command Code: D5H
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	D5H	00D9H

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Working status	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 bytes	2 bytes
0xEF01	xxxx	07H	000 4 H	xxH	xxxx	sum

Notes: Confirmation Code =00H means OK;

Confirmation Code = 01H means there is an error in receiving the packet;

Confirmation Code = FDH indicates parameter error ;

Sum refer to Checksum.

3.1.34 Save configuration information to flash--PS_SaveConfigurationToFlash

- Function Description: Save sleep time and working mode to internal flash. The data will not be lost when power is off. **This command will affect the service life of the device. Please do not use it frequently .**
- Input Parameters: Save configuration (0 saves sleep time, 1 saves working mode)
- Return Parameters: Confirmation Word , working status (0 means saving successfully, 1 means saving failed)
- Command Code: D6H
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Save Configuration	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes
0xEF01	xxxx	01H	000 4 H	D6 H	xxxx	xxxx

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

Notes 1 : Confirmation Code =00H means OK;

Confirmation Code =01H means Error ;

Sum refer to Checksum.

Notes 2 : When reading and writing to the ST chip's Flash memory , avoid frequent writes to extend its lifespan. Before writing to Flash, a page must be erased , a time -consuming process that takes approximately 30ms. If the value in memory is the same as the value to be saved , the erase and write operations are unnecessary , reducing unnecessary Flash memory wear.

3.1.35 Get STM32 firmware version--PS_Get FirmwareVersion

- Function Description: Get the STM32 firmware version
- Input Parameters: none
- Return Parameters: Confirmation Word , firmware version number
- Command Code: D7H
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	D7H	00DBH

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Firmware version	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 bytes	2 bytes
0xEF01	xxxx	07H	0004H	xxH	xxxx	sum

Notes: Confirmation Code =00H means OK;

Confirmation Code = 01H means there is an error in receiving the packet;

Sum refer to Checksum.

3.2 Modular Instruction Set

Serial Number	Command code	Name	Description
1	0x30	PS_Cancel	Cancel order
2	0x31	PS_AutoEnroll	Automatically register module instructions
3	0x32	PS_AutoIdentify	Automatic fingerprint verification command

For automatic registration (0x31) and automatic verification (0x32), if there is no finger press during the waiting period and the module has not returned a timeout, do not directly cut off the VDD power supply. Please send the "cancel command" PS cancel(0x30) first.

3.2.1 Cancel Command--PS_Cancel

- Function Description: Cancel automatic template registration and automatic fingerprint verification. The encryption level is set to 0 or 1 case supports this
- Function.
- Input Parameters: none
- Return Parameters: Confirmation Word
- Command Code: 30H
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	30H	xxxxH

- Response Packet Format:

Header	Device Address	Identifier	Length	Confirmation Code	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

Notes: Confirmation Code = 00H means the setting is canceled successfully;

Confirmation Code = 01H means the cancellation of setting failed;

Confirmation Code = 31H means the function does not match the encryption level;

Sum refer to Checksum.

3.2.2 Automatic registration template PS_AutoEnroll

- Function Description: One-stop fingerprint registration , including fingerprint collection, feature generation, template combination, template storage and other functions. 0 or 1 This function is supported.
- Input Parameters: ID Number, number of entries, parameters
- Return Parameters: Confirmation Word , Parameters
- Command Code: 31H
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	ID Number	Entry Times	Parameter	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes	1 byte	2 bytes	2 bytes
0xEF01	xxxx	01H	0008H	31H	xxxxH	xxH	xxH	sum

- Auxiliary instructions :

ID Number : high byte first , low byte last. For example, Fingerprint No. 1 is 0001H .

Number of entries : 1 byte , input 2 times , then 02H , input 4 times 04H .

Parameter : The lowest bit is bit 0 .

1) bit 0 : Reserved;

2) bit 1 : Reserved;

3) bit 2 : Whether the module is required to return the current status at key steps during the registration process ,
0- required return , 1- not required return;

4) bit 3 : Whether to allow overwriting ID No. , 0- not allowed , 1- allowed;

5) bit 4 : Control bit for allowing repeated fingerprint registration , 0- allowed , 1- not allowed;

6) bit 5 : During registration , during multiple fingerprint collections , is it required to remove the finger before entering the next fingerprint image collection ? 0- Request to leave; 1- Do not request to leave;

7) bit 6~ bit 15 : Reserved.

- Response Packet Format:

Header	Chip address	Identifier	Length	Confirmation Code	parameter2 bytes		Sum	Notes
					parameter 1	parameter2		
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	1 byte	2 bytes	
0xEF01	xxxx	07H	0005H	xxH	00H	00H	sum	Instruction legitimacy detection : legitimate / ..
0xEF01	xxxx	07H	0005H	xxH	01H	1	sum	Image collection results : Success / Timeout
0xEF01	xxxx	07H	0005H	xxH	02H	1	sum	Generate feature results : Success / Failure

Header	Chip address	Identifier	Length	Confirmation Code	parameter2 bytes		Sum	Notes
					parameter 1	parameter2		
0xEF01	xxxx	07H	0005H	xxH	03H	1	sum	Fingers leave , No. 1 successfuentry : Success / Timeout
.....								
0xEF01	xxxx	07H	0005H	xxH	01H	n	sum	Image collection results : Success / Timeout
0xEF01	xxxx	07H	0005H	xxH	02H	n	sum	Generate feature results : Success / Failure
0xEF01	xxxx	07H	0005H	xxH	04H	F0H	sum	Merge Templates
0xEF01	xxxx	07H	0005H	xxH	05H	F1H	sum	Registered detection
0xEF01	xxxx	07H	0005H	xxH	06H	F2H	sum	Template storage results

- Confirmation Code , Parameters 1 and parameters 2 The return value

Confirmation Code	Interpretation	Para 1	Interpretation	Para2	Interpretation
00H	success	00H	Fingerprint legitimacy detection	00H	Fingerprint legitimacy detection
01H	fail	01H	Get Image	F0H	Merge Templates
07H	Failed to create feature	02H	Production characteristics	F1H	Check if the finger is registered
08H	registration process, you are required to enter similar Refers to the area , when the current feature is different from the previous feature Similar (this function is disabled by default)	03H	Determine if the finger leaves	F2H	Storage Template
0aH	Merge failed	04H	Merge Templates	n	Current entryn times
0bH	ID number out of range	05H	Registration Inspection		
18H	Read and Write FLASH Error	06H	Storage Template		
1fH	The fingerprint database is full				
22H	Fingerprint template is not empty				
25H	The number of entries is set incorrectly				
26H	time out				
27H	Fingerprint registered				
28H	registration process, you are required to enter a different Refers to the area where the current feature overlaps with the previous feature Too much area (this function is disabled by default)				
31H	Functionality does not match encryption level				
35H	Illegal data				
3bH	Poor template quality				

- Instruction Description :

1) If the specified ID number is invalid , the Confirmation Code , Parameter 1 and Parameter 2 are returned

(hereinafter directly described as return) : 0b 00 00H. Validity check :

- If specified ID If the number is invalid , it returns : 0b 00 00H .
- If the number of entries is incorrect , it wilreturn 25 00 00H In the state of not covering fingerprints , if the fingerprint library is full, it wilreturn 1f 00 00H ;
- If specified ID If the template already exists, return 22 00 00H .
- If the command validity check is successfu, it wilreturn 00 00 00H , and enter the first fingerprint entry.

2) Wait for image acquisition to succeed (return 00 01 0nH).

3) Wait for the feature generation to succeed (00 02 0nH) . If the feature generation fails (07 02 0nH) , or if the feature merging fails (0a 02 0nH) , wait for the image acquisition to succeed again.

① If the registration logic mode is configured as 1 , that is, each time a different finger area is required to be entered , if the area of the current feature overlaps with the previous feature too much when generating the feature (return 28 02 0nH),wait for the image acquisition to succeed again;

② If the registration logic mode is configured as 2 , that is, it requires that similar finger areas be entered each time , if this feature is not similar to the previous feature when generating the feature (return to 08 02 0nH),Wait again for the image acquisition to succeed.

4) Wait for the finger to be released . The first entry is successful(00 03 0nH) . After the finger is released , the process jumps to step 2 and enters the next loop until n is the set number of entries. Note: If the finger is set not to be released during the entry process , the process directly returns to the first successful entry and jumps to step 2. The last fingerprint collection , if the finger is not released, is a successful entry response.

5) Synthesize the template and get the template quality . If successful, it returns 00 04 F0H . If the template quality is too poor, it returns 3B 04 F0H.

6) Fingerprint duplicate check , which means matching the newly entered finger with the stored finger (enable or disable this function by setting parameter bit4) . If there is an identical fingerprint , it returns 27 05 F1H and ends the process; if there is no identical fingerprint , it returns 00 05 F1H.

7) Register the template data . If storage fails, it returns 01 06 F2H and ends the process. If storage succeeds, it returns 00 06 F2H.

8) If the PS_Cancel command is received , the command is terminated and a response is returned.

3.2.3 Automatic fingerprint verification--PS_Autoidentify

- Function Description: Automatically collect fingerprints , search for target templates or entire fingerprint templates in the fingerprint library , and return the search results. If the score of the template comparison with the currently collected fingerprint is greater than the maximum threshold , and the target template is an incomplete feature, the collected fingerprint is used. The feature updates the blank area of the target template. One-stop search includes functions such as acquiring images , generating features , and searching fingerprints. The encryption level is set to 0 or 1 This function is supported.
- Input Parameters: Security level, ID No., parameters
- Return Parameters: Confirmation Word, page number (matching fingerprint template)
- Command Code: 32H
- Command Packet Format:

Header	Device Address	Identifier	Length	Command Code	Security Level	ID Number	Parameter	Sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes	2 bytes	2 bytes
0xEF01	xxxx	01H	0008H	32H	xxH	xxxxH	xxxxH	sum

- Auxiliary instructions :

ID Number : 2 bytes , big-endian mode. For example, Fingerprint No.1 is 00 01H ID, No. 0xFFFF, then proceed 1 : N search; No proceed 1:1 match.

Parameter : The lowest bit is bit 0 .

- 1) bit 0 : Reserved;

2) bit 1 : Reserved;

3) bit 2 : During the registration process , whether the module is required to return the current status at the key step , 0- required return , 1- not required return return;

4) bit 3~ bit 15 : Reserved.

● Response Packet Format:

Header	Chip address	Identifier	Length	Confirmation Code	Para	ID Number	Score	Sum	Notes
2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes	2 bytes	2 bytes	
0xEF01	xxxx	07H	0008H	xxH	00H	xxxxH	xxxxH	sum	Instruction legitimacy detection : legitimate / ..
0xEF01	xxxx	07H	0008H	xxH	01H	xxxxH	xxxxH	sum	Image collection results : Success / Timeout
0xEF01	xxxx	07H	0008H	xxH	05H	xxxxH	xxxxH	sum	Search results : Success / Failure

● I Confirmation Code , Parameters 1 and parameters 2 The return value

Confirmation Code	Interpretation	Parameter	Interpretation
00H	success	00H	Fingerprint legitimacy detection
01H	fail	01H	Get Image
07H	Failed to create feature	05H	Registered fingerprint comparison
08H	Fingerprint mismatch		
09H	No fingerprint found		
0bH	ID number out of range		
17H	Residual fingerprints		
23H	Fingerprint template is empty		
24H	The fingerprint database is empty		
26H	time out		
27H	Indicates that the fingerprint already exists		
31H	Functionality does not match encryption level		

● Instruction Description :

1) If the fingerprint database is empty , the confirmation code and parameters are returned (hereinafter directly described as return) : 24 00H If you specify If the ID number is invalid , it will return 0b 00H If the registered Template does not exist , it returns 23 00H .

2) The command legitimacy check is successful and returns 00 00H , and enter fingerprint entry.

3) If a complete fingerprint entry is not completed within the set timeout period , the 26 00H , end the process.

4) Check the correctness of the input fingerprint image. If it is incorrect , wait for the next image acquisition.

5) If the fingerprint is correct , it will return 00 01H , that is, the fingerprint is entered and the image is obtained successfully.

6) If the production feature fails , return 09 0 5H , end the process.

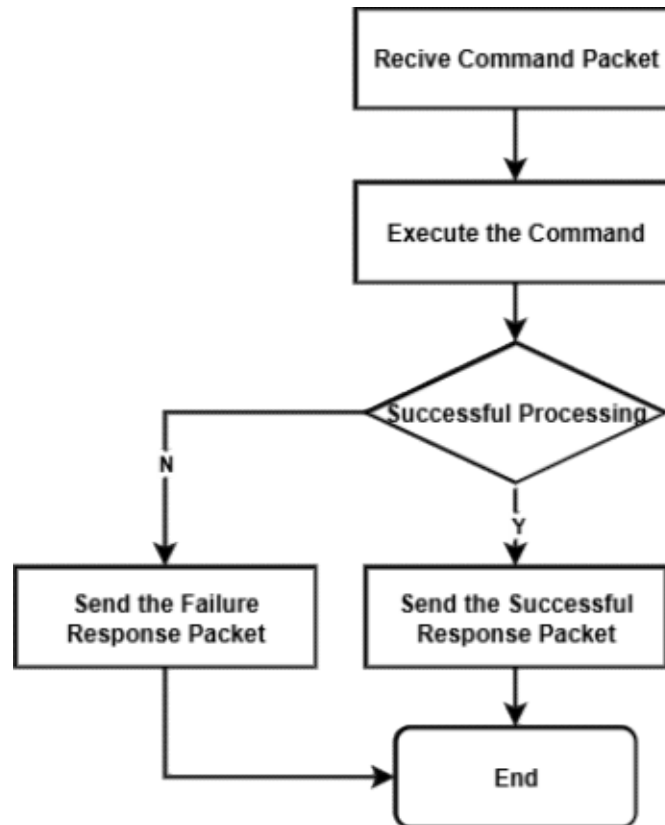
7) After the feature is generated successfully , the currently collected fingerprint template is compared with the registered fingerprint template and the result is returned. If the comparison fails , it returns 09 05H , end the process; if the comparison is successful, return 00 05H , and the correct ID Numbers and scores.

8) If you receive If the PS_Cancel command is executed , the command is terminated and a response is returned.

4. Functional Demonstration Example

This demonstration ensures that the internal fingerprint module is activated, and if password verification is required, ensures that the password verification is successful.

4.1 UART Command Packet Processing Process



Function Implementation Example 1 : UART Command Packet Processing

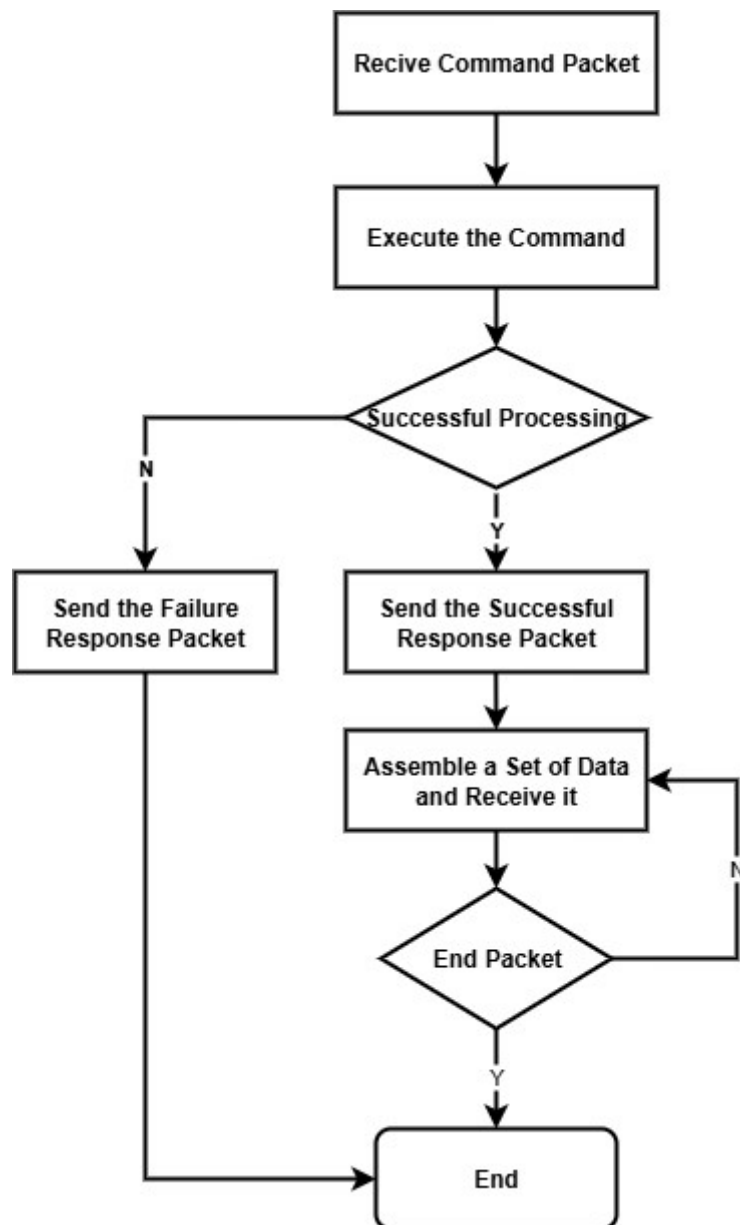
4.2 UART Data Packet Sending Process

UART Before transmitting a data packet , the device must first receive the instruction packet for transmitting the data packet , prepare for the transmission , and then send a success response packet. Finally, the data packet transmission begins. The data packet mainly includes : Header , Device Address , Identifier , Length , Data, and Sum .

There are two main types of data packet identifiers : 02H and 08H 02H : Data packet , and there are subsequent packets. 08H : The last data packet , that is, the end The data length is pre-set and is mainly divided into : 32 , 64 , 128 ,and 256 Four types .

For example , the data length to be transmitted is 1 K bytes , the data length pre-set in the data packet is 128 bytes , then we need to 1 K Bytes of data are divided into 8 data packets are transmitted. Each data packet contains : 2 bytes Header , 4 bytes Chip address, 1 byte Identifier , 2 bytes Bag Length , 128 bytes Data and 2 bytes Sum, each data length is 139 bytes In addition , among the 8 data packets , the first 7 data packets The report identification is 02H , the last end data packet report identifier is 08H . Finally , it is important to note that if the length of the end packet does not reach 139 bytes ,

The actual length is transmitted and will not be expanded in other ways. 139 bytes .



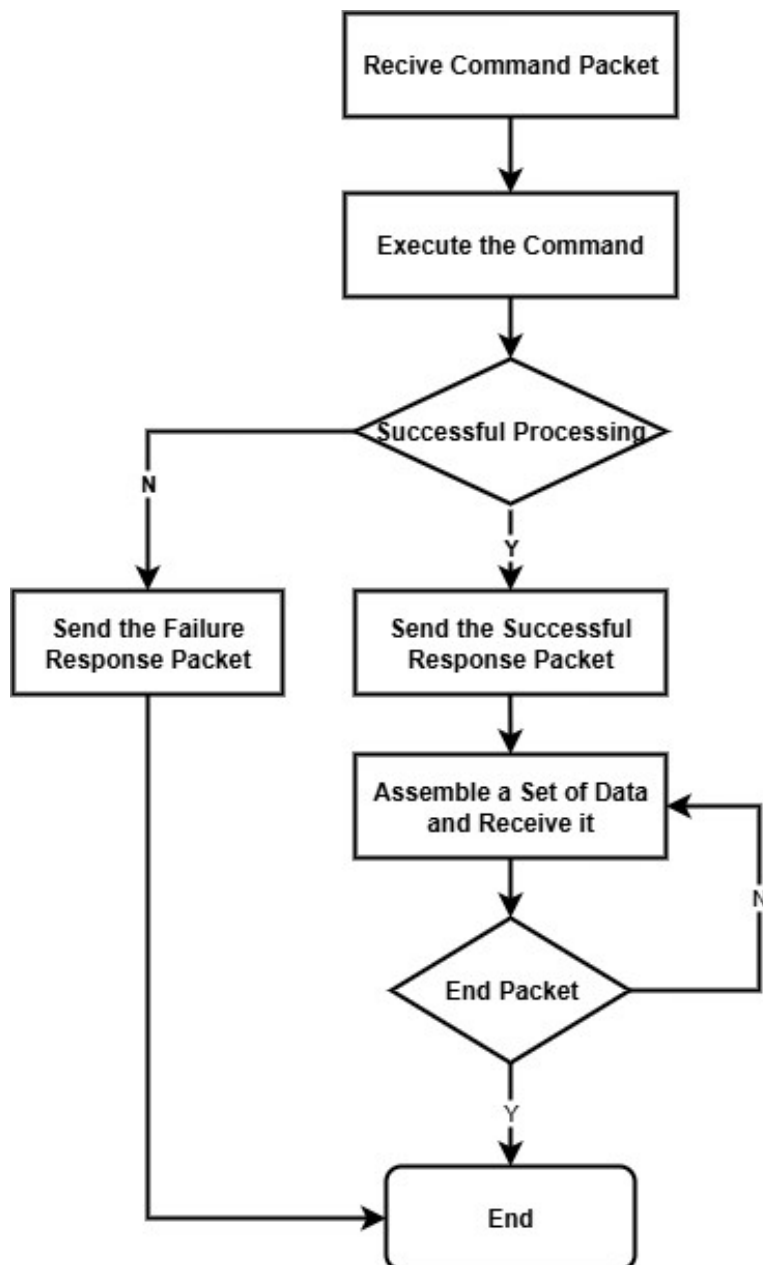
Function Implementation Example 2 : UART Data Packet Sending Process

4.3 UART Data Packet Reception Process

UART Before transmitting a data packet , you must first receive the instruction packet for transmitting the data packet , and then send a successful response packet after preparing for the transmission . Transmit data packets. Data packets mainly include : Header , Device Address , Identifier , Length , Data and Sum .

There are two main types of data packet identifiers : 02H and 08H 02H : Data packet , and there are subsequent packets. 08H : The last data packet , that is, the end The data length is pre-set and is mainly divided into : 32 , 64 , 128 ,and 256 Four types.

For example , the data length to be transmitted is 1 K bytes , the data length pre-set in the data packet is 128 bytes , then we need to 1 K Bytes of data are divided into 8 data packets are transmitted. Each data packet contains : 2 bytes Header , 4 bytes Chip address, 1 byte Identifier , 2 bytes Bag Length , 128 bytes Data and 2 bytes Sum, each data length is 139 bytes In addition , among the 8 data packets , the first 7 data packets The report identification is 02H , the last end data packet report identifier is 08H . Finally , it is important to note that if the length of the end packet does not reach 139 bytes ,The actual length is transmitted and will not be expanded in other ways. 139 bytes .



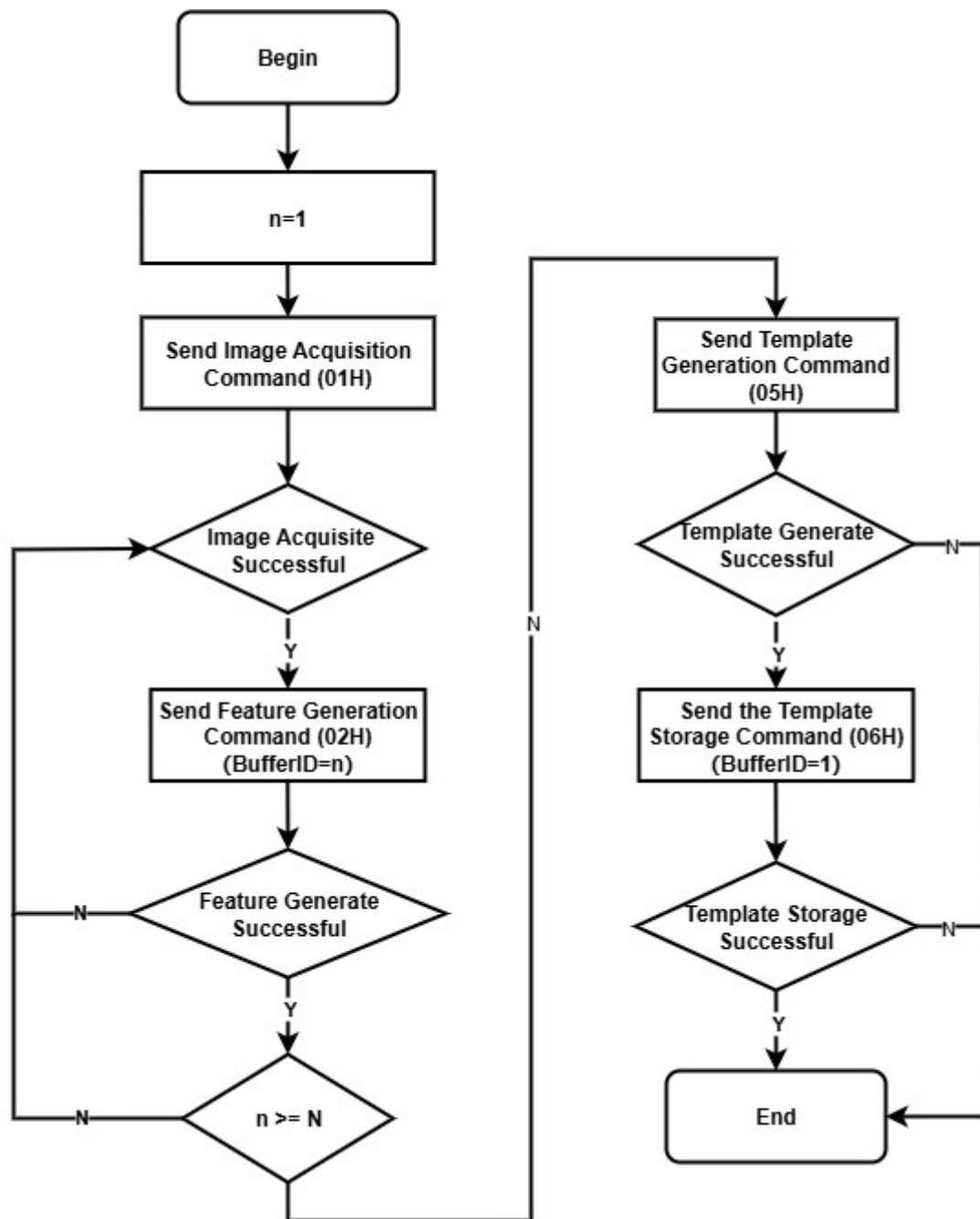
Function Implementation Example 3 : UART Data Packet Receiving Process

4.4 Fingerprint Registration Process

Such as Table in the registration process , if you want to controthe finger area of each entry , you can configure the registration logic mode by writing the system register instruction . (Registration logic mode defaults to 0 , the finger area is not controlled each time input).

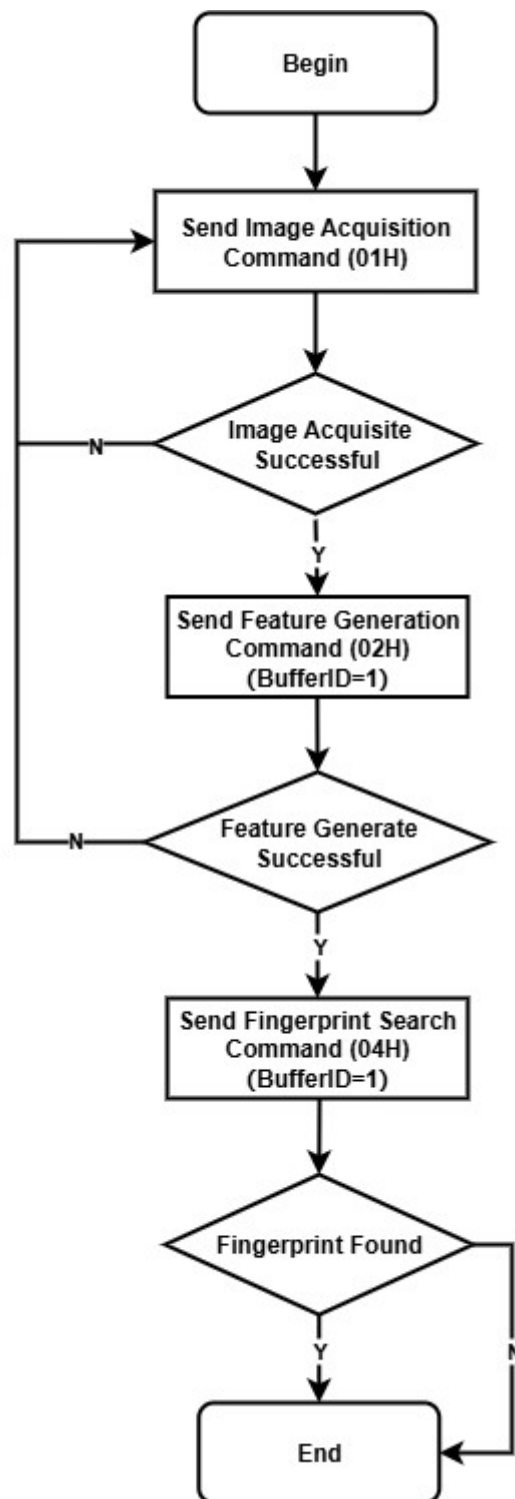
1. If you need to enter different finger areas each time , configure the registration logic mode to 1 , send the command EF 01 FF FF FF FF 01 00 05 0E 03 01 00 18 , when generating features, if the overlap area between the current and previous features is too large , the Confirmation Code in the module response package is 0x28 ;

2. If you need to enter similar finger areas each time , configure the registration logic mode to 2. Send the command EF 01 FF FF FF FF 01 00 05 0E 03 02 00 19 When generating a feature , if this feature is not similar to the previous feature , the Confirmation Code in the module response package is 0x08 .



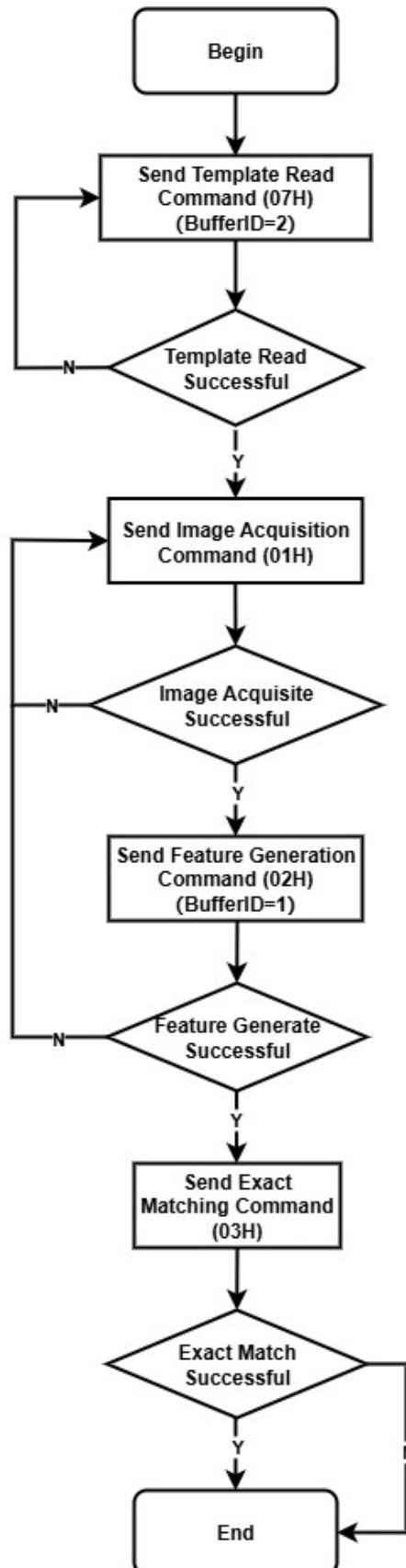
Function Implementation Example 4 : Fingerprint Registration Process

4.5 Fingerprint Search Process



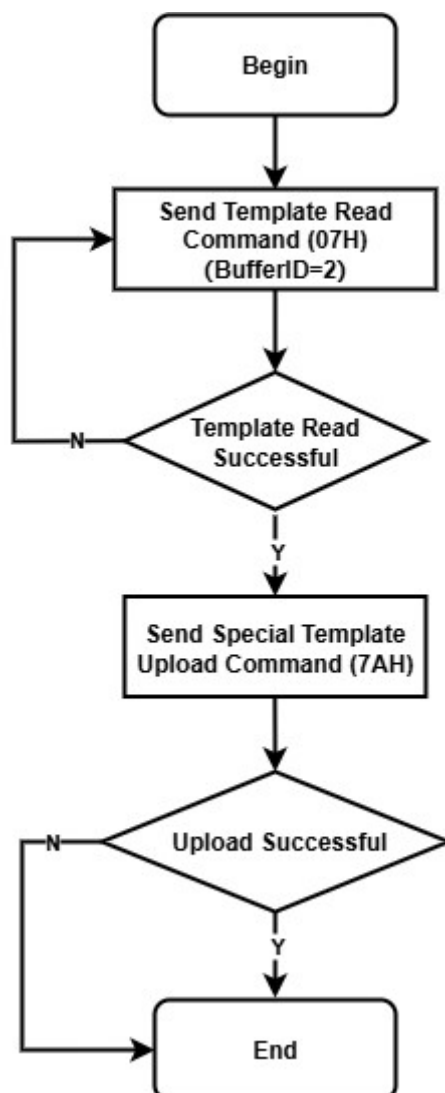
Function Implementation Example 5 : Fingerprint Search Process

4.6 Master Controller Loads a Fingerprint Feature or Template for Accurate Comparison



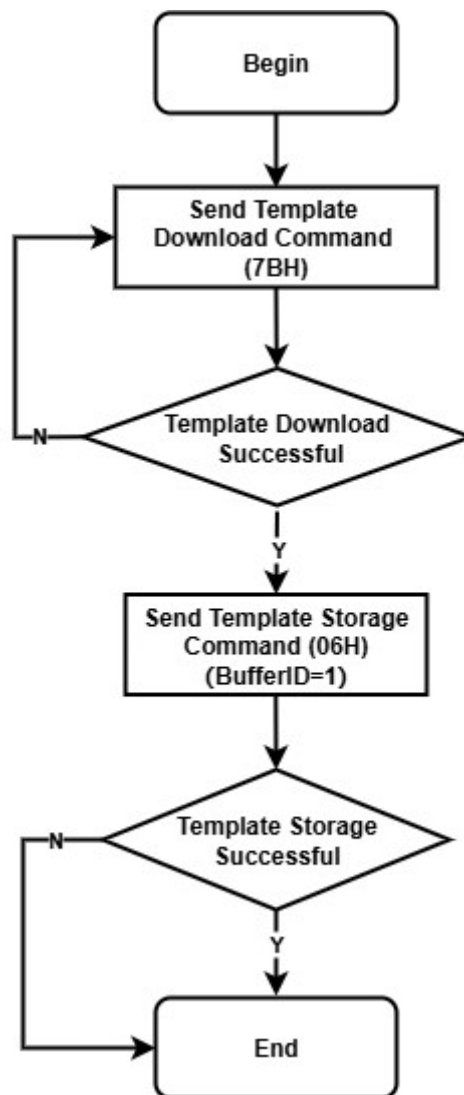
Function Implementation Example 6 : Master Loads a Fingerprint Feature or Template for Accurate Comparison

4.7 Special Upload Template Process



Function Example 7 : Speciaupload template process

4.8 Special Download Template Process



Function Example 8 : Speciadownload template process

5. Return type Table

Serial Number	Value	Meaning
1	0x00	The command is executed or OK
2	0x01	Packet reception error
3	0x02	No finger on the sensor
4	0x03	Failed to enter fingerprint image
5	0x04	The fingerprint image is too dry or too light to be a feature.
6	0x05	The fingerprint image is too wet or too blurry to generate features
7	0x06	Fingerprint images are too messy to generate features
8	0x07	The fingerprint image is normal, but there are too few feature points (or the area is too small)
9	0x08	Fingerprint mismatch
10	0x09	No fingerprint found
11	0x0A	Feature merging failed
12	0x0B	When accessing the fingerprint library, the address number exceeds the fingerprint library range.
13	0x0C	Error or invalid template read from fingerprint database
14	0x0D	Failed to upload features
15	0x0E	The module cannot receive subsequent data packets
16	0x0F	Image upload failed
17	0x10	Failed to delete template
18	0x11	Failed to clear the fingerprint database.
19	0x12	Unable to enter low power state
20	0x13	Incorrect password
21	0x14	System reset failed
22	0x15	There is no valid original image in the buffer and the image cannot be generated.
23	0x16	Online upgrade failed
24	0x17	Residual fingerprints or the finger has not moved between two captures
25	0x18	Error reading or writing FLASH
26	0x19	Random number generation failed
27	0x1A	Invalid register number
28	0x1B	Register setting content error number
29	0x1C	Notepad page number specified error
30	0x1D	Port operation failed
31	0x1E	Automatic enroll failed
32	0x1F	Fingerprint database full

Serial Number	Value	Meaning
33	0x20	Device Address Error
34	0x21	Incorrect password
35	0x22	Fingerprint template is not empty
36	0x23	Fingerprint template is empty
37	0x24	The fingerprint database is empty
38	0x25	The number of entries is set incorrectly
39	0x26	time out
40	0x27	Fingerprint already exists
41	0x28	Fingerprint features are associated
42	0x29	Sensor initialization failed
43	0x2A	Module information is not empty
44	0x2B	Module information is empty
45	0x2C	OTP operation failed
46	0x2D	Key generation failed
47	0x2E	Key does not exist
48	0x2F	Security algorithm execution failed
49	0x30	The encryption and decryption results of the security algorithm are incorrect
50	0x31	Functionality does not match encryption level
51	0x32	Key locked
52	0x33	Small image area
53	0x34	Static foreign objects in the image (Orange)
54	0x35	Illegal data
56	0x37	Static foreign matter in the feature (Orange)
57	0xFB	The module is busy (it is asking whether to shut down the module. After returning, the module exits the inquiry and the timer is reset)
58	0xFC	This operation is blocked
59	0xFD	Parameter error
60	0xFE	The fingerprint module is not turned on
61	0xFF	Passive activation

6. RC Calculation Method Example (C/C++)

buffer indicates the instruction packet cache, size indicates the instruction length.

6.1 Verification

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

6.2 Calculation

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