

Fingerprint Sensor Module User Manual

We are dedicated to developing sensing technology, and providing customers with an innovative and diverse range of sensor products.

Our sensors and state-of-the-art fingerprint recognition algorithm technologies provide advanced and convenient fingerprint acquisition and verification.

Table of Contents

1	Hardware Guide.....	5
1.1	UART	5
1.2	USB	5
1.3	Cooperation Between UART and USB.....	5
2	Software Developing Guide	6
2.1	Parameter Table	6
2.2	System Parameter Memory Structure	7
2.3	Notepad	8
2.4	Buffer and Fingerprint Database	8
2.5	Features and Templates.....	8
2.6	Feature file structure.....	9
2.7	ROM	9
3	Instruction Set.....	10
3.1	PS_GetImage	10
3.2	PS_GenChar	10
3.3	PS_Match	10
3.4	PS_Search	10
3.5	PS_RegModel	10
3.6	PS_StoreChar	10
3.7	PS_LoadChar	10
3.8	PS_UpChar	10
3.9	PS_DownChar.....	11
3.10	PS_UpImage	11
3.11	PS_DownImage.....	11
3.12	PS_DeletChar.....	11
3.13	PS_Empty.....	11
3.14	PS_WriteReg.....	11
3.15	PS_ReadSysPara.....	11
3.16	PS_SetPwd	11
3.17	PS_VfyPwd.....	11
3.18	PS_GetRandomCode.....	12
3.19	PS_SetChipAddr	12
3.20	PS_ReadINFpage	12
3.21	PS_WriteNotepad.....	12

3.22	PS_ReadNotepad.....	12
3.23	PS_ValidTemplateNum	12
3.24	PS_ReadIndexTable	12
3.25	PS_Cancel.....	12
4	Instruction Form specification.....	13
4.1	Data packet Form	13
4.2	Instruction ACK.....	14
4.3	Detail Instruction	17
4.3.1	PS_GetImage(GetImage).....	17
4.3.2	PS_GenChar(Generate Feature)	18
4.3.3	PS_Match(Pattern-Match Two Fingerprints Feature).....	19
4.3.4	PS_Search (Search Fingerprint)	20
4.3.5	PS_RegModel(Merge Features Generating Template).....	21
4.3.6	PS_StoreChar(Store Templates).....	22
4.3.7	PS_LoadChar(Readout Templates)	23
4.3.8	PS_UpChar(upload feature or templates)	24
4.3.9	PS_DownChar(Download Feature or Templates)	25
4.3.10	PS_UpImage(Upload Image)	26
4.3.11	PS_DownImage(Download Image)	27
4.3.12	PS_DeletChar>Delete Template).....	28
4.3.13	PS_Empty(Clear Fingerprint Database)	29
4.3.14	PS_WriteReg(Write System Registers).....	30
4.3.15	PS_ReadSysPara(Read System Basic Parameter)	31
4.3.16	PS_SetPwd(Set Password).....	33
4.3.17	PS_VfyPwd(Verify Password)	34
4.3.18	PS_GetRandomCode(Get Random Code)	35
4.3.19	PS_SetChipAddr(Set Chip Address)	36
4.3.20	PS_ReadINFpage(Read Flash Info Page)	37
4.3.21	PS_WriteNotepad(Write Notepad)	38
4.3.22	PS_ReadNotepad(Read Notepad)	39
4.3.23	PS_ValidTemplateNum(Read valid template number).....	40
4.3.24	PS_ReadIndexTable(Read IndexTable).....	41
4.3.25	PS_Cancel(Cancel command)	42
5	Function Implement Illustration	43
5.1	Enroll Flow (N times finger press for 1 Finger ID)	43
5.2	Search Database Flow	44

5.3	Match Flow	45
5.4	Capture one finger and upload Char to Host Flow	46
5.5	Host Download Char and search Flow	47
6	Revision history	48

1 Hardware Guide

1.1 UART

- UART default baud rate : 57600
- Data format: 8 bits of data, No parity, 2 stop bits
- UART baud can set by instruction, from 9600 to 115200bps.
- UART_TX, UART_RX can be linked to receive and transfer pin of a MCU directly(Vio=3.3V)
However if the host is PC, a RS232 level converter (3.3V) is needed for the right connection.

1.2 USB

- Standard USB interface, with USB protocol embedded
- Compatible with USB2.0 and able to work at Low Speed/Full Speed;
- Default VID = 0x2B41 , PID = 0x0101

1.3 Cooperation Between UART and USB

- The Host can communicates with Device via UART or USB
- Both of them use the same communication protocol and command
- The two share the same data buffer

2 Software Developing Guide

2.1 Parameter Table

- a) Parameter table contains the basic parameters for the running of protocol and algorithm. The whole software system will access this content. So understanding it and carefully setting the Parameter Table is vital for the correct use of the chip.
- b).The parameter table is configured by MCU initialization program (ROM-resident or user-defined) at the chip's first power on, with the result being stored in the parameter section of the serial FLASH. For the successive power up, the DSP initialization program will load the parameter table to RAM before initialize the system registers according to it. The length of parameter table is 64 words (128 bytes).
- c) Structure:

Type	Number	Name	Length (word)	Default/content	Description
PART1	1	SSR	1	0	State register
	2	SensorType	1	0--15	
	3	DataBaseSize	1	According to FLASH	Fingerprint Database size
PART2	4	SecurLevel	1	3	divided into 5 levels
	5	DeviceAddress	2	0xffffffff	Device address: Chip address, can be set by instruction
	6	CFG_PktSize	1	1	Data packet size
	7	CFG_BaudRate	1	6	Baudrate factor
	8	CFG_VID	1		
	9	CFG_PID	1		
	10	Reserved	1		The eight registers (6~13) are system configuration table
	11	Reserved	1		
	12	Reserved	1		
	13	Reserved	1		
	14	ProductSN	4	Product SN,ASCII code	Device descriptor
	15	SoftwareVersion	4	ASCII code	
	16	Manufacturer	4	ASCII code	

Type	Number	Name	Length (word)	Default/content	Description
	17	SensorName	4	ASCII code	
	18	Password	2	00000000H	Default 00000000H
	19	JtagLockFlag	2	00000000H	
	20	SensorInitEntry	1	Entry address	Sensor initialization routine entry
	21	SensorGetImageEntry	1	Entry address	Get-image program entry
	22	Reserved	27		
PART3	23	ParaTableFlag	1	0X1234	Parameter Table valid flag

d) The parameter table is located at the first page of the SPM (System Parameter Memory);

e) The parameter table is loaded from FLASH to RAM space (0x200~0x23F) after power on with its structure and order remain the same;

2.2 System Parameter Memory Structure

a) System Parameter Memory Structure

PageNum	Content	Comment
0	Reserved	
1	Parameter table	
2	User's Notepad	
3	Reserved	
4	Reserved	
5	Reserved	
6	Reserved	
7	Index of fingerprint	Can index as much as

b) System Parameter Memory is divided into 8 pages, 512 bytes per page.

2.3 Notepad

512-byte memory is set aside in flash for User's notepad. The notepad is divided into 16 pages logically, 32 bytes per page. The host can access any page by instruction PS_WriteNotepad or PS_ReadNotepad.

Note: when written, the whole page is taken as a whole and its former contents will be replaced.

2.4 Buffer and Fingerprint Database

There are a 72K-byte image buffer ImageBuffer and two 7-byte feature file buffers CharBuffer1 and CharBuffer2 in the chip. Users can read/write any of the above buffers by instructions. CharBuffer1 or CharBuffer2 can be used to store general feature file as well as template feature file. In order to cut down the communication time when uploading or downloading image through UART interface, only the upper 4 bits of the pixel byte are applied, i.e. to combine two pixels bytes into one byte in transmitting. No such tricks are made when transmitting through USB interface.

The capacity of fingerprint database changes with FLASH memory capacity, which is identified by the system automatically. Fingerprint templates are stored sequentially according to the SN, while the definition of SN is: $0 \sim (N-1)$ (N =fingerprint database capacity).

Note: Only through SN index will users access the fingerprint database.

2.5 Features and Templates

Fingerprint feature file occupies 425 bytes, including general information as well as minutiae information; Template file occupies 2129 bytes, sum of five features files of the same fingerprint.

2.6 Feature file structure

The minutiae number of a feature file is no more than 99. Of the total 2129 bytes (size of feature file is 425 bytes), the first 56 bytes is the file header used for general information; The latter 369 bytes are to store minutiae information, 4 bytes for each minutiae.

File Header Format

Name	Bytes	Description
Flag	0	Feature file flag. To distinguish the feature files generated by different sensors or algorithms. "0": the feature file is invalid or deleted. So no feature file can be stored to database when the flag is "0"
Type	1	Feature file type. 0: the file only contains file header; 1: the file contains file header and reduced minutiae information. 2: the file contains file header and complete minutiae information.
Quality	2	Quality of feature. Value range is 0~100 with the larger value indicating the higher feature quality.
Number	3	Minutiae number within the range of 5~50.
SN	4~5	Searching assistant.
Background table	6~39	Zippped information of background table
Singularity point coordinate	40~43	Includes (x, y) coordination information of the two center points.
--	44~55	Reserved

Feature Unit Structure (4 bytes)

31	23	22	13	4	1	0
x		y	Angle	Minutiae quality		Attribute

2.7 ROM

ROM contains complete fingerprint identification system, including communication protocol, command interpretation, algorithm, FLASH management and drivers for sensors HF105.

3 Instruction Set

3.1 PS_GetImage

Instruction Code: 01H

Function: Reading images from sensor and store them in the image buffer

3.2 PS_GenChar

Instruction Code: 02H

Function: Generating fingerprint features according to original images and store them in CharBuffer1

3.3 PS_Match

Instruction Code: 03H

Function: Pattern-matching the feature file in CharBuffer1 and CharBuffer2

3.4 PS_Search

Instruction Code: 04H

Function: Using the feature files in CharBuffer1 to search the whole or part of fingerprint database

3.5 PS_RegModel

Instruction Code: 05H

Function: Merging the feature files in CharBuffer1 and generate them into template to store in CharBuffer1

3.6 PS_StoreChar

Instruction Code: 06H

Function: Storing files in the feature buffer to FLASH fingerprint database

3.7 PS_LoadChar

Instruction Code: 07H

Function: Reading a template from FLASH fingerprint database to feature buffer

3.8 PS_UpChar

Instruction Code: 08H

Function: Uploading files in the feature buffer to the host

3.9 PS_DownChar

Instruction Code: 09H

Function: Downloading a feature file from the host to the feature buffer

3.10 PS_UpImage

Instruction Code: 0AH

Function: Uploading original image

3.11 PS_DownImage

Instruction Code: 0BH

Function: Downloading original image

3.12 PS_DeletChar

Instruction Code: 0CH

Function: Deleting a feature file of the FLASH fingerprint database

3.13 PS_Empty

Instruction Code: 0DH

Function: Clearing FLASH fingerprint database

3.14 PS_WriteReg

Instruction Code: 0EH

Function: Writing SOC system register

3.15 PS_ReadSysPara

Instruction Code: 0FH

Function: Reading system basic parameter

3.16 PS_SetPwd

Instruction Code: 12H

Function: Setting device handshake passwords

3.17 PS_VfyPwd

Instruction Code: 13H

Function: Verifying device handshake passwords

3.18 PS_GetRandomCode

Instruction Code: 14H

Function: Sampling random code

3.19 PS_SetChipAddr

Instruction Code: 15H

Function: Setting chip address

3.20 PS_ReadINFpage

Instruction Code: 16H

Function: Reading contents of FLASH Information Page

3.21 PS_WriteNotepad

Instruction Code: 18H

Function: Writing notepad

3.22 PS_ReadNotepad

Instruction Code: 19H

Function: Reading notepad

3.23 PS_ValidTemplateNum

Instruction Code: 1dH

Function: Reading valid template number

3.24 PS_ReadIndexTable

Instruction Code: 1fH

Function: Reading index table of template.

3.25 PS_Cancel

Instruction Code: 30H

Function: Cancel command

4 Instruction Form specification

MCU SOC can form complete fingerprint identification module with several necessary periphery circuit (sensor, flash, power supply, etc.). The module is in Slave mode all the time. The host can issue different instructions to the module, for various functions. The host instruction, modules ACK and data exchanges are all work according to given format data packet. The host should packet instructions and data which need transmitting as well as analyze received data packets based-on the following format. For multi-byte data, the high byte is in front, low byte retro, e.g. two-byte data 00 60 should be expressed as 0060, not 0600.

4.1 Data packet Form

Instruction /data packet altogether be classified into three categories:

Packet flag=01 Command packet

Packet flag=02 Data packet, and with continue packet

Packet flag=08 The last data packet,i.e. end packet

All data packets should be with packet header:0xEF01

01 Command packet format:

Name	Packet header	Chip address	Packet flag	Packet length	Instruction	Parameter 1	...	Parameter n	Check sum
Byte No.	2bytes	4 bytes	1 byte	2 bytes	1 byte				2 bytes
Content	0xEF01	xxxx	01	N=					

02 Data packet format:

Name	Packet header	Chip address	Packet flag	Packet length	Data	Check sum
Byte No.	2bytes	4bytes	1 byte	2 bytes	N bytes	2 bytes
Content	0xEF01	xxxx	02			

08 End packet format:

Name	Packet header	Chip address	Packet flag	Packet length	Data	Check sum
Byte No.	2bytes	4bytes	1 byte	2 bytes	N bytes	2 bytes
Content	0xEF01	xxxx	08			

- The data packet should not enter implement flow respectively, but behind the instruction packet or ACK packet;
- Downloaded or uploaded data packet are in the same format;
- Packet length= The total byte number from packet length to Sum (instruction, parameter or data), including Sum, but not the byte number of packet length itself;
- Sum is the total bytes from packet flag to Sum, the carry will be ignored if it exceed 2 bytes;
- The default chip address is 0xFFFFFFFF before its issue. Once the host issues chip address by instruction, all data packets should receive and transmit according to the address. Chip will reject packets with wrong address.

4.2 Instruction ACK

ACK is to report relevant command running condition and result to the host, the ACK packet contains parameter and can be with continue data packet. Only when the host received the ACK packet of SOC can it confirm the condition of SOC packet receiving and instruction implementing.

ACK packet format:

Name	Packet header	Chip address	Packet flag	Packet length	Confirm Code	Return parameter	Check sum
Byte No.	2 bytes	4bytes	1 byte	2 bytes	1 byte	N bytes	2 bytes
Content	0xEF01		07				

Definition of Confirm codes:

1. 00h: Indicates instruction implementing end or OK;
2. 01h: Indicates data packet receiving error;
3. 02h: Indicates no finger on the sensor;
4. 03h: Indicates getting fingerprint image failed;
5. 04h: Indicates the fingerprint image is too dry or too light to generate feature;
6. 05h: Indicates the fingerprint image is too humid or too blurry to generate feature;
7. 06h: Indicates the fingerprint image is too amorphous to generate feature;

8. 07h: Indicates the fingerprint image is in order, but with too little minutiae (or too small area) to generate feature;
9. 08h: Indicates the fingerprint unmatched;
10. 09h: Indicates no fingerprint searched;
11. 0ah: Indicates the feature merging failed;
12. 0bh: Indicates the address SN exceeding the range of fingerprint database when accessing to it;
13. 0ch: Indicates template reading error or invalid from the fingerprint database;
14. 0dh: Indicates feature uploading failed;
15. 0eh: Indicates the module cannot receive continue data packet;
16. 0fh: Indicates image uploading failed;
17. 10h: Indicates module deleting failed;
18. 11h: Indicates the fingerprint database clearing failed;
19. 12h: Indicates cannot be in low power consumption;
20. 13h: Indicates the password incorrect;
21. 14h: Indicates the system reset failed;
22. 15H: Indicates there is no valid original image in buffer to generate image;
23. 16H: Indicates on-line upgrading failed;
24. 17H: Indicates there are incomplete fingerprint or finger stay still between twice image capturing;
25. 18H: Indicates read-write FLASH error;
26. 0xf0: Existing instruction of continue data packet, ACK with 0xf0 after receiving correctly;
27. 0xf1: Existing instruction of continue data packet, the command packet ACK with 0xf1;
28. 0xf2: Indicates Sum error when burning internal FLASH;
29. 0xf3: Indicates packet flag error when burning internal FLASH;
30. 0xf4: Indicates packet length error when burning internal FLASH;
31. 0xf5: Indicates the code length too long when burning internal FLASH;
32. 0xf6: Indicates burning FLASH failed when burning internal FLASH;
33. 0x19: Non-defined error;

- 34. 0x1a: Invalid register number;
- 35. 0x1b: Register distributing content wrong number;
- 36. 0x1c: Notepad page number appointing error;
- 37. 0x1d: Port operation failed;
- 38. 0x1e: Automatic enroll failed;
- 39. 0x1f: Fingerprint database is full;
- 40. 0x20—0xefh: Reserved.
- 41. 0x21: Must verify password first.

Instructions can only be transmitted from the host to the module, and the module ACK to the host.

After the system power on reset, it will first check whether the default device handshake passwords have been modified or not. If not, the system will ensure the host no passwords verification, then SOC enters into normal work directly; if yes, should first verify the device handshake passwords, SOC enters into normal work after passing passwords.

4.3 Detail Instruction

4.3.1 PS_GetImage(GetImage)

- Function description: Detecting finger, then get the fingerprint image and store it in ImageBuffer. Returning to confirm code to show: getting success, no finger, etc.
- Input parameter: none
- Return value: Confirm words
- Instruction Code: 01H
- Instruction packet format:

Packet header	Chip address	Packet flag	Packet length	Instruction code	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	03H	01H	05H

- ACK packet format:

Packet header	Chip address	Packet flag	Packet length	Confirm Code	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	03H	xxH	sum

Comment:

Confirm Code=00H shows getting success;

Confirm Code=01H shows receiving packet error;

Confirm Code=02H shows no finger on the sensor;

Confirm Code=03H shows getting failed;

Sum=Check sum.

4.3.2 PS_GenChar(Generate Feature)

- Function description: Generating the original image in ImageBuffer to fingerprint feature file and store it in CharBuffer.
- Input parameter: BufferID (feature buffer number)
- Return value: Confirm words
- Instruction Code: 02H
- Instruction packet format:

Packet header	Chip address	Packet flag	Packet length	Instruction code	Buffer number	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes
0xEF01	xxxx	01H	04H	02H	BufferID	sum

Comment: In Enroll mode, the BufferID meas enroll times,the max enroll time setting to 5.

- ACK packet format:

Packet header	Chip address	Packet flag	Packet length	Confirm Code	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	03H	xxH	sum

Comment:

Confirm Code=00H shows generating success;

Confirm Code=01H shows receiving packet error;

Confirm Code=06H Shows the fingerprint image is too amorphous to generate feature;

Confirm Code=07H Shows the fingerprint image is in order, but with too little minutiaes to generate feature;

Confirm Code=15H Shows there is no valid original image in buffer to generate image;

Sum=Check sum.

4.3.3 PS_Match(Pattern-Match Two Fingerprints Feature)

- Function description: Pattern-matching the feature files in CharBuffer1 and CharBuffer2
- Input parameter: none
- Return value: Confirm words , match scores
- Instruction code:03H
- Instruction packet format:

Packet header	Chip address	Packet flag	Packet length	Instruction code	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	03H	03H	07H

- ACK packet format:

Packet header	Chip address	Packet flag	Packet length	Confirm code	Score	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes	2 bytes
0xEF01	xxxx	07H	05H	XxH	XxH	sum

Comment:

Confirm Code=00H shows fingerprint matched;

Confirm Code=01H shows receiving packet error;

Confirm Code=08H shows fingerprint unmatched;

Sum=Check sum.

4.3.4 PS_Search (Search Fingerprint)

- Function description: To search the whole or part of fingerprint database with feature files in CharBuffer1 or CharBuffer2. If get, jump to the original page.
- Input parameter: BufferID, StartPage, PageNum
- Return value: Confirm words, page number(Matched fingerprint template)
- Instruction code: 04H
- Instruction packet format:

Packet header	Chip address	Packet flag	Packet length	Instruction code	Buffer number	Parameter	Parameter	Check sum
2 bytes	4bytes	1 byte	2bytes	1 byte	1 byte	2 bytes	2 bytes	2bytes
0xEF01	xxxx	01H	08H	04H	BufferID	StartPage	pageNum	sum

Comment: The BufferID in CharBuffer1 are 01H.

- ACK packet format:

Packet header	Chip address	Packet flag	Packet length	Confirm code	Page number	Score	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes	2 bytes	2 bytes
0xEF01	xxxx	07H	07H	XxH	PageID	MatchScore	sum

Comment:

Confirm Code=00H shows searching success;

Confirm Code=01H shows receiving packet error;

Confirm Code=09H shows unsearched, here the page number and score are "0";

Sum=Check sum.

4.3.5 PS_RegModel(Merge Features Generating Template)

- Function description: Merging feature files in CharBuffer1 and CharBuffer2 to generate templates, store the result in CharBuffer1 and CharBuffer2.
- Input parameter: None
- Return value: Confirm words
- Instruction code: 05H
- Instruction packet format:

Packet header	Chip address	Packet flag	Packet length	Instruction code	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	03H	05H	09H

- ACK packet format:

Packet header	Chip address	Packet flag	Packet length	Confirm Code	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	03H	xxH	sum

Comment:

Confirm Code=00H shows merging success;

Confirm Code=01H shows receiving packet error;

Confirm Code=0aH shows merging failed (two fingerprints are not from the same finger)

Sum=Check sum.

4.3.6 PS_StoreChar(Store Templates)

- Function description: Storing the template files in CharBuffer1 to the location of PageIDNum flash database.
- Input parameter: BufferID (buffer number), PageID (fingerprint database location number)
- Return value: Confirm words
- Instruction code: 06H
- Instruction packet format:

Packet header	Chip address	Packet flag	Packet length	Instruction code	Buffer number	Location number	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes	2 bytes
0xEF01	xxxx	01H	06H	06H	BufferID	PageID	sum

Comment: The BufferID in CharBuffer1 are 1h

- ACK packet format:

Packet header	Chip address	Packet flag	Packet length	Confirm Code	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	03H	xxH	sum

Comment:

Confirm Code=00H shows storing success;

Confirm Code=01H shows receiving packet error;

Confirm Code=0bH shows PageID exceeded the range of fingerprint database;

Confirm Code=18H shows writing FLASH error;

Sum=Check sum.

4.3.7 PS_LoadChar(Readout Templates)

- Function description: Readin the fingerprint templates which appointed IDNum in flash database to template buffer CharBuffer1 or CharBuffer2
- Input parameter: BufferID (buffer number), PageID (fingerprint database template number)
- Return value: Confirm words
- Instruction code: 07H
- Instruction packet format:

Packet header	Chip address	Packet flag	Packet length	Instruction code	Buffer number	Page number	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes	2 bytes
0xEF01	xxxx	01H	06H	07H	BufferID	PageID	sum

Comment: The BufferID in CharBuffer1 and CharBuffer2 are 1h and 2h.

- ACK packet format:

Packet header	Chip address	Packet flag	Packet length	Confirm Code	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	03H	xxH	sum

Comment:

Confirm Code=00H shows reading success;

Confirm Code=01H shows receiving packet error;

Confirm Code=0cH shows reading error or template invalid;

Confirm Code=0bH shows PageID exceeded the range of fingerprint database;

Sum=Check sum.

4.3.8 PS_UpChar(upload feature or templates)

- Function description: Uploading the feature files in feature buffer to the host.
- Input parameter: BufferID (buffer number)
- Return value: Confirm words
- Instruction code: 08H
- Instruction packet format:

Packet header	Chip address	Packet flag	Packet length	Instruction code	Buffer number	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes
0xEF01	xxxx	01H	04H	08H	BufferID	sum

Comment: The BufferID in CharBuffer1 and CharBuffer2 are 1h and 2h.

- ACK packet format:

Packet header	Chip address	Packet flag	Packet length	Confirm Code	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	03H	xxH	sum

Comment :

Confirm Code=00H shows transmitting data packet later;

Confirm Code=01H shows receiving packet error;

Confirm Code=0dH shows instruction running error;

Sum=Check sum.

- Transmitting continue data packet after ACK

4.3.9 PS_DownChar(Download Feature or Templates)

- Function description: The host download feature files to one feature buffer of the module
- Input parameter: BufferID(buffer number)
- Return value: Confirm words
- Instruction code:09H
- Instruction packet format:

Packet header	Chip address	Packet flag	Packet length	Instruction code	Buffer number	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes
0xEF01	xxxx	01H	04H	09H	BufferID	sum

Comment : The BufferID in CharBuffer1 and CharBuffer2 are 1h and 2h.

- ACK packet format :

Packet header	Chip address	Packet flag	Packet length	Confirm Code	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	03H	xxH	sum

Comment:

Confirm Code=00H shows can receive continue data packet;

Confirm Code=01H shows receiving packet error;

Confirm Code=0eH shows cannot receive continue data packet;

Sum=Check sum.

- Receiving continue data packet after ACK

4.3.10 PS_UpImage(Upload Image)

- Function description: Uploading data in image buffer to the host
- Input parameter: none
- Return value: Confirm words
- Instruction code: 0aH
- Instruction packet format:

Packet header	Chip address	Packet flag	Packet length	Instruction code	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	03H	0aH	000eH

- ACK packet format:

Packet header	Chip address	Packet flag	Packet length	Confirm Code	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	03H	xxH	sum

Confirm:

Code=00H shows going on transmitting continue data packet;

Confirm Code=01H shows receiving packet error;

Confirm Code=0fH shows cannot transmit continue data packet;

Sum=Check sum.

- Transmitting continue data packet after ACK
- A byte contains two pixels, each pixel with 4bits

4.3.11 PS_DownloadImage(Download Image)

- Function description: The host download image data to the module
- Input parameter: none
- Return value: Confirm words
- Instruction code: 0bH
- Instruction packet format:

Packet header	Chip address	Packet flag	Packet length	Instruction code	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	03H	0bH	000fH

- ACK packet format:

Packet header	Chip address	Packet flag	Packet length	Confirm Code	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	03H	xxH	sum

Comment:

Confirm Code=00H shows can receive continue data packet;

Confirm Code=01H shows receiving packet error;

Confirm Code=0eH shows cannot receive continue data packet;

Sum=Check sum.

- Receiving continue data packet after ACK, the data packet length should be 64, 128 or 256
- A byte contains two pixels, each pixel with 4bits

4.3.12 PS_DeletChar(Delete Template)

- Function description: Deleting the beginning N fingerprint templates which appointed IDNum in flash database
- Input parameter: PageID(fingerprint database template number), N=deleted templates number
- Return value: Confirm words
- Instruction code: 0cH
- Instruction packet format:

Packet header	Chip address	Packet flag	Packet length	Instruction code	Page number	Delete number	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 byte	2 bytes	2 bytes
0xEF01	xxxx	01H	07H	0cH	PageID	N	sum

- ACK packet format:

Packet header	Chip address	Packet flag	Packet length	Confirm Code	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	03H	xxH	sum

Comment :

Confirm Code=00H shows deleting module success;

Confirm Code=01H shows receiving packet error;

Confirm Code=10H shows deleting module failed;

Sum=Check sum.

4.3.13 PS_Empty(Clear Fingerprint Database)

- Function description: Deleting all fingerprint modules in flash database
- Input parameter: none
- Return value: Confirm words
- Instruction code: 0dH
- Instruction packet format:

Packet header	Chip address	Packet flag	Packet length	Instruction code	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	03H	0dH	0011H

- ACK packet format:

Packet header	Chip address	Packet flag	Packet length	Confirm Code	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	03H	xxH	sum

Comment:

Confirm Code=00H shows clearing success;

Confirm Code=01H shows receiving packet error;

Confirm Code=11H shows clearing failed;

Sum=Check sum.

4.3.14 PS_WriteReg(Write System Registers)

- Function description: Writing module registers
- Input parameter: Register SN
- Return value: Confirm words
- Instruction code: 0eH
- Instruction packet format:

Packet header	Chip address	Packet flag	Packet length	Instruction code	SN of Registers	Content	Check sum
2 bytes	4bytes	1 byte	2bytes	1 byte	1 byte	1 bytes	2 bytes
0xEF01	xxxx	01H	05H	0eH	4/5/6	xx	sum

- ACK packet format:

Packet header	Chip address	Packet flag	Packet length	Confirm Code	Check sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	03H	xxH	sum

Register NO.	Register Name	Content description
4	Baud rate control register	9600 * N
5	Match threshold register	1: level1 2: level2 3: level3 4: level4 5: level5
6	Packet size register	0: 32bytes 1: 64bytes 2: 128bytes 3: 256bytes

Comment1:

Confirm code=00H shows OK;

Confirm Code=01H shows receiving packet error;

Confirm Code=1aH shows register SN error;

Sum=Check sum.

Comment2:

When write the instruction implementing of system register (PS_WriteReg), first ACK with the original configuration, after that modify the system configuration and record it to FLASH. At next power on, the system will work with new configuration.

4.3.15 PS_ReadSysPara(Read System Basic Parameter)

- Function description: Read the module's basic parameter(baudrate, packet size etc.).

The former 16 bytes of **Parameter Table** stores module's basic information of communication and configuration, which are module's basic parameters.

- Input parameter: none
- Return value: Confirm words + basic parameter(16bytes)
- Instruction code: 0fH
- Instruction packet format:

Packet header	Chip address	Packet flag	Packet length	Instruction code	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	03H	0fH	0013H

- ACK packet format:

Packet header	Chip address	Packet flag	Packet length	Confirm Code	Basic parameter table	Check sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	16 bytes	2 bytes
0xEF01	xxxx	07H	13H	xxH	Refer to the following table	sum

Comment:

Confirm code=00H shows OK;

Confirm Code=01H shows receiving packet error;

Sum=check sum.

Table. Basic **parameter table**

Name	Content description	Offset(word)	Size(word)
State register	Content of system's state registers	0	1
Sensor type	Sensor type code	1	1
Fingerprint database size	Fingerprint database capacity	2	1
Security rank	Security level code (1/2/3/4/5)	3	1
Device address	32 bits device address	4	2
Data packet size	Data packet size code: 0: 32 bytes 1: 64 bytes 2: 128 bytes 3: 256 bytes	6	1
Baudrate configuration	N (Baudrate=9600*N bps)	7	1

4.3.16 PS_SetPwd(Set Password)

- Function description: Setting module handshake password
- Input parameter: Password
- Return value: Confirm words
- Instruction code: 12H
- Instruction packet format:

Packet header	Chip address	Packet flag	Packet length	Instruction code	Password	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	4 byte	2 bytes
0xEF01	Xxxx	01H	07H	12H	Password	sum

Comment : The default value of module address is “0”

- ACK packet format:

Packet header	Chip address	Packet flag	Packet length	Confirm Code	Check sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	03H	xxH	sum

Comment:

Confirm code=00H shows OK;

Confirm Code=01H shows receiving packet error;

Sum=check sum.

4.3.17 PS_VfyPwd(Verify Password)

- Function description: Verifying module handshake passwords
- Input parameter: Password
- Return value: Confirm words
- Instruction code: 13H
- Instruction packet format:

Packet header	Chip address	Packet flag	Packet length	Instruction code	Password	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	4 byte	2 bytes
0xEF01	xxxx	01H	07H	13H	Password	sum

- ACK packet format:

Packet header	Chip address	Packet flag	Packet length	Confirm Code	Check sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	03H	xxH	sum

Comment:

Confirm code=00H shows OK;

Confirm Code=01H shows receiving packet error;

Confirm Code=13H shows password incorrect;

Confirm Code=21H shows Must verify password first;

Sum=check sum.

4.3.18 PS_GetRandomCode(Get Random Code)

- Function description: Making chip generate a random code and return to the host
- Input parameter: none
- Return value: Confirm words
- Instruction code: 14H
- Instruction packet format:

Packet header	Chip address	Packet flag	Packet length	Instruction code	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	03H	14H	0018H

- ACK packet format:

Packet header	Chip address	Packet flag	Packet length	Confirm Code	Random code	Check sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	4 bytes	2 bytes
0xEF01	xxxx	07H	03H	xxH	xxxx	sum

Comment:

Confirm code=00H shows generating success;

Confirm Code=01H shows receiving packet error;

Sum=check sum.

4.3.19 PS_SetChipAddr(Set Chip Address)

- Function description: Setting chip address
- Input parameter: none
- Return value: Confirm words
- Instruction code: 15H
- Instruction packet format:

Packet header	Chip address	Packet flag	Packet length	Instruction code	Chip address	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	4 bytes	2 bytes
0xEF01	xxxx	01H	03H	15H	xxxx	sum

- ACK packet format:

Packet header	Chip address	Packet flag	Packet length	Confirm Code	Check sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	07H	xxH	sum

Comment:

Confirm code=00H shows generating success;

Confirm Code=01H shows receiving packet error;

Sum=check sum.

- When the host download instruction packet, the chip address adopts default address:
0xffffffff; the ACK packet address space adopts the newly generated address;
- After running the instruction, the chip address be fixed, remains no change.
Only to clear FLASH can you change it;
- After running the instruction,all data packets should apply the generated address.

4.3.20 PS_ReadINFpage(Read Flash Info Page)

- Function description: Reading the information page in FLASH (512bytes)
- Input parameter: none
- Return value: Confirm words
- Instruction code: 16H
- Instruction packet format:

Packet header	Chip address	Packet flag	Packet length	Instruction code	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	03H	16H	001aH

- ACK packet format:

Packet header	Chip address	Packet flag	Packet length	Confirm Code	Check sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	03H	xxH	sum

Comment:

Confirm code=00H shows generating success;

Confirm Code=01H shows receiving packet error;

Confirm Code=0dH shows instruction running failed;

Sum=check sum.

- Transmitting continue data packet after ACK.

4.3.21 PS_WriteNotepad(Write Notepad)

- Function description: Distributing a-256-bytes FLASH space for storing user's data in the module, named as user notepad. The pad is divided into 16 pages logically. To write the notepad command --- for read-in user's 32 bytes data to appointed notepad page
- Input parameter: NotePageNum, user content
- Return value: Confirm words
- Instruction code: 18H
- Instruction packet format:

Packet header	Chip address	Packet flag	Packet length	Instruction code	Page number	User information	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	1 byte	32 bytes	2 bytes
0xEF01	xxxx	01H	24H	18H	0~15	content	sum

- ACK packet format:

Packet header	Chip address	Packet flag	Packet length	Confirm Code	Check sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	03H	xxH	sum

Comment:

Confirm code=00H shows generating success;

Confirm Code=01H shows receiving packet error;

Sum=check sum.

4.3.22 PS_ReadNotepad(Read Notepad)

- Function description: Reading 128bytes data in FLASH user field
- Input parameter: none
- Return value: Confirm words + User information
- Instruction code: 19H
- Instruction packet format:

Packet header	Chip address	Packet flag	Packet length	Instruction code	Page number	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes
0xEF01	xxxx	01H	04H	19H	0~15	sum

- ACK packet format:

Packet header	Chip address	Packet flag	Packet length	Confirm Code	User information	Check sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	32 bytes	2 bytes
0xEF01	xxxx	07H	23H	xxH	content	sum

Comment:

Confirm code=00H shows generating success;

Confirm Code=01H shows receiving packet error;

Sum=check sum.

4.3.23 PS_ValidTemplateNum(Read valid template number)

- Function description: Reading valid template number
- Input parameter: none
- Return value: Confirm words, valid template number (ValidN)
- Instruction code: 1dH
- Instruction packet format:

Packet header	Chip address	Packet flag	Packet length	Instruction code	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	03H	1dH	21H

- ACK packet format:

Packet header	Chip address	Packet flag	Packet length	Confirm Code	valid template number	Check sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes	2 bytes
0xEF01	xxxx	07H	05H	xxH	ValidN	sum

Comment:

Confirm code=00H shows generating success;

Confirm Code=01H shows receiving packet error;

Sum=check sum.

4.3.24 PS_ReadIndexTable(Read IndexTable)

- Function description: Reading index table of template.
- Input parameter: Index page. Page 0: 0~255. Page 1: 256-511.
Each bit means one template, 1: means it was used, 0: means empty.
- Return value: Confirm words, index information
- Instruction code: 1fH
- Instruction packet format:

Packet header	Chip address	Packet flag	Packet length	Instruction code	Index page	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes
0xEF01	xxxx	01H	04H	1fH	0~1	sum

- ACK packet format:

Packet header	Chip address	Packet flag	Packet length	Confirm Code	Index information	Check sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	32 bytes	2 bytes
0xEF01	xxxx	07H	23H	xxH	Index	sum

Comment:

Confirm code=00H shows generating success;

Confirm Code=01H shows receiving packet error;

Sum=check sum.

4.3.25 PS_Cancel(Cancel command)

- Function description: Cancel command.
- Input parameter: none.
- Return value: Confirm words
- Instruction code: 30H
- Instruction packet format:

Packet header	Chip address	Packet flag	Packet length	Instruction code	Check sum
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	03H	30H	34H

- ACK packet format:

Packet header	Chip address	Packet flag	Packet length	Confirm Code	Check sum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	03H	xxH	sum

Comment:

Confirm code=00H shows cancel command success;

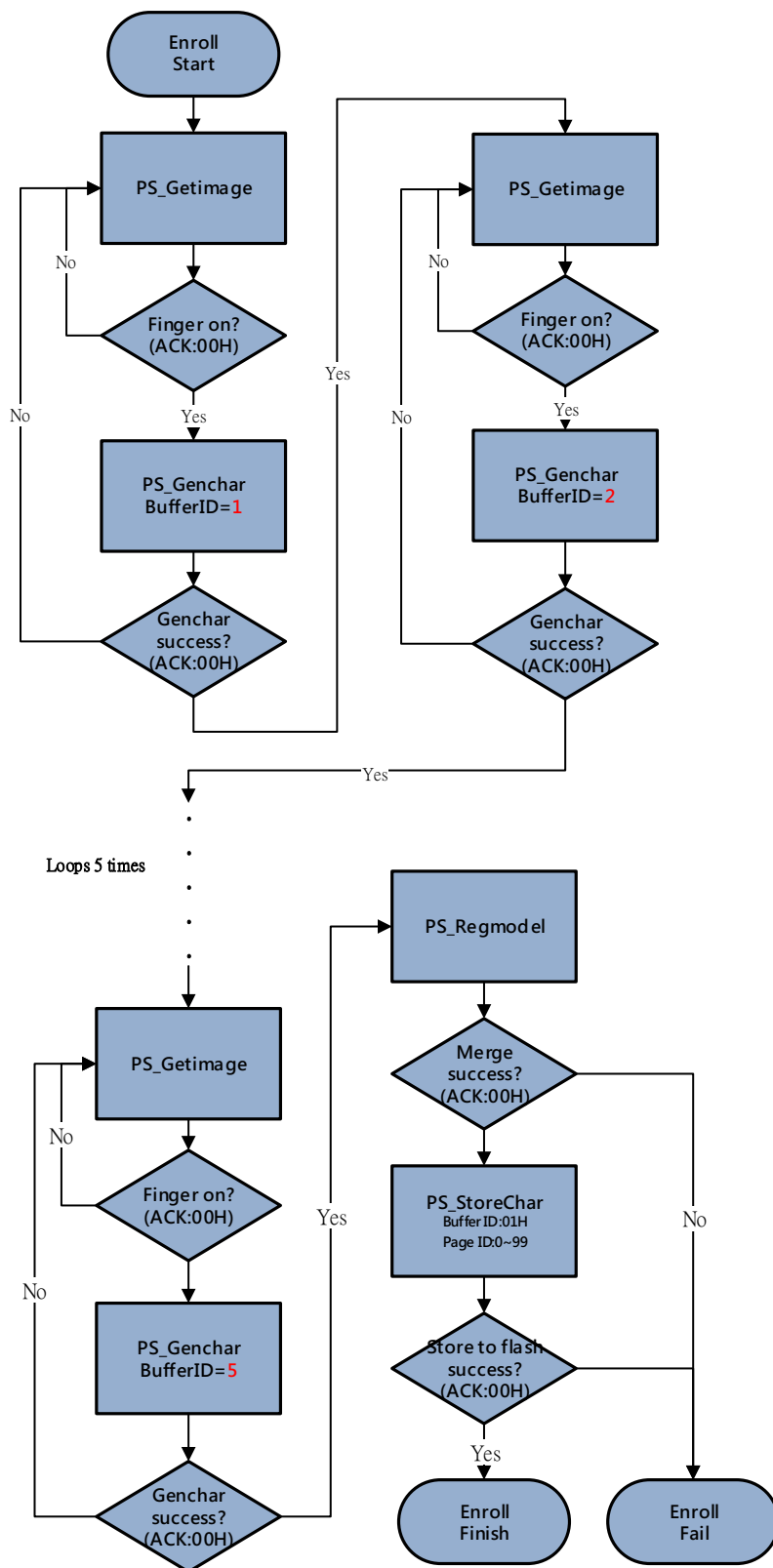
Confirm Code=01H shows cancel command fail;

Sum=check sum.

5 Function Implement Illustration

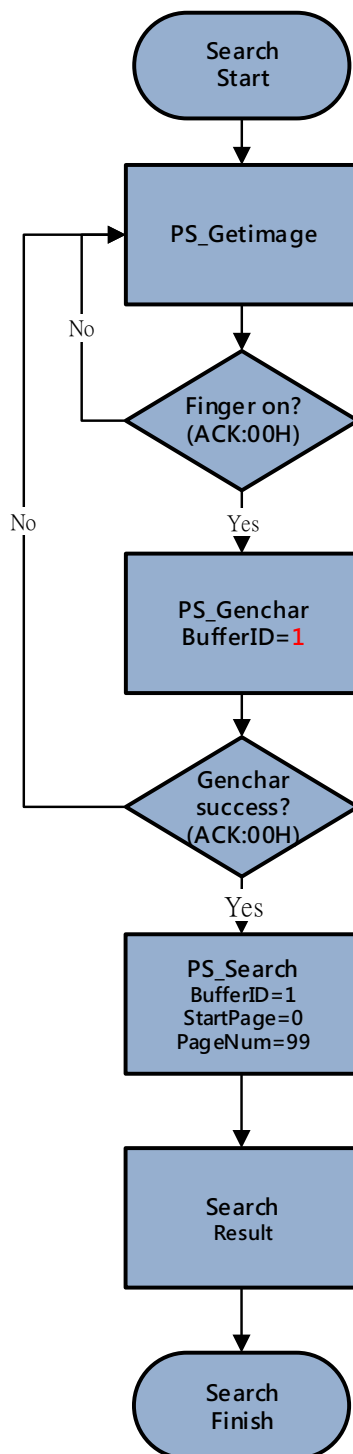
5.1 Enroll Flow (N times finger press for 1 Finger ID)

Pressing fingerprint n times (e.g. 5 times) to record a template and store in flash fingerprint database;



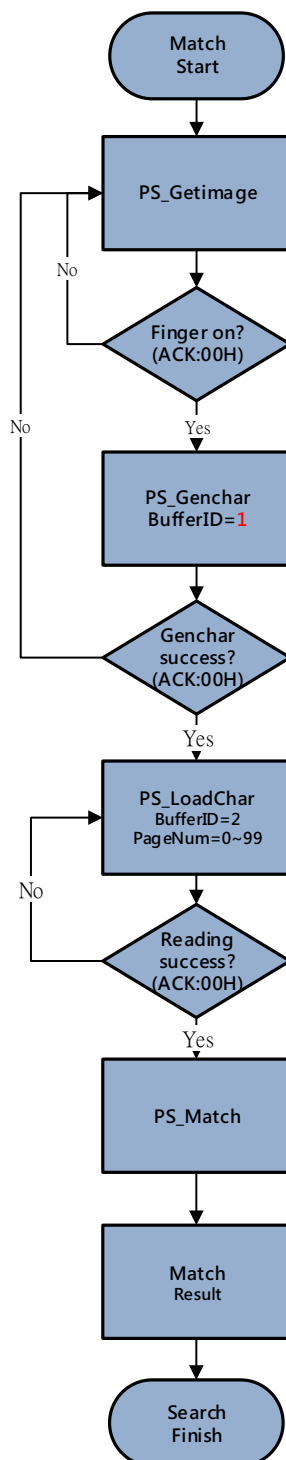
5.2 Search Database Flow

Reading an image from sensor and searching the database field from 0 – 99



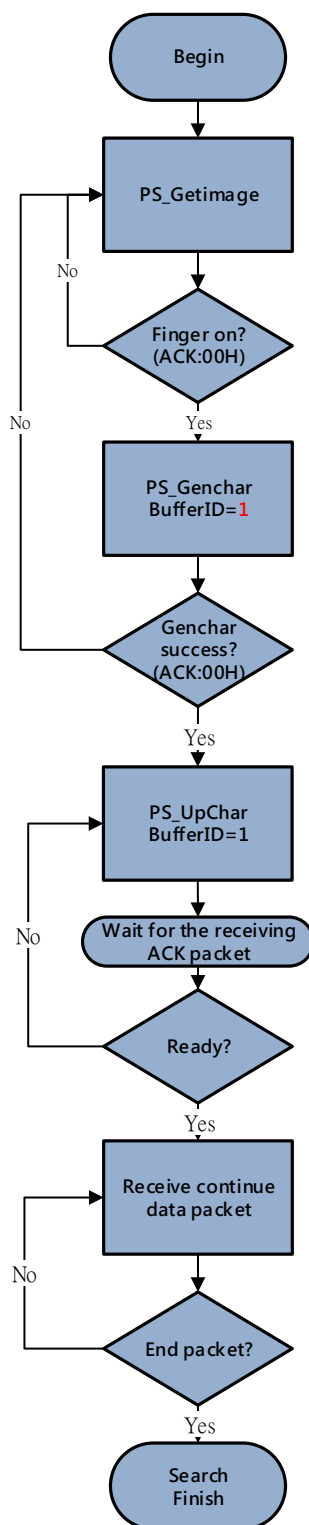
5.3 Match Flow

Reading an image from sensor and match appointed enrolled ID (0 – 99) from fingerprint database



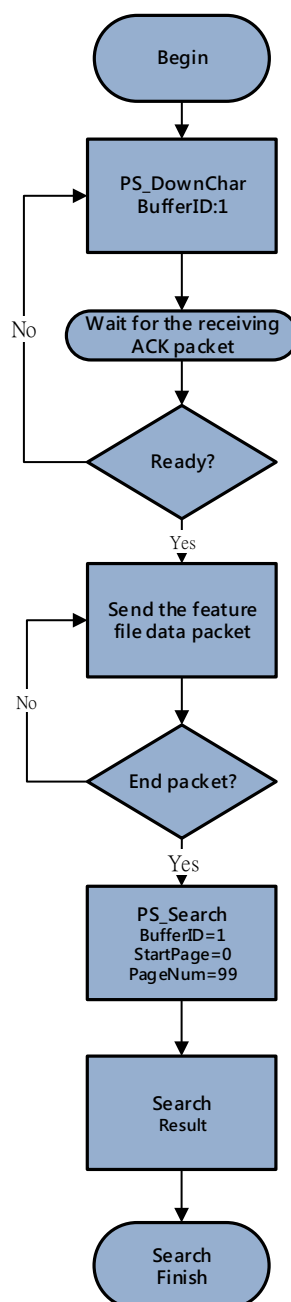
5.4 Capture one finger and upload Char to Host Flow

Capture a fingerprint image from sensor and generating the feature file, then upload to host.



5.5 Host Download Char and search Flow

Downloading a fingerprint feature from the host and searching fingerprint database with it.



6 Revision history

Version	Date	Changes	Approved	Checked	Author
V1.0	2019/01/21	Initial draft	Albert ke	Hyde Wu	Hyde Wu