

OP-69

Fingerprint Integrated Module

User's Manual

- 1 Summary..... 1**
 - 1.1 Main Functions..... 1
 - 1.2 Applications 1
- 2 Module Hardware Description 2**
 - 2.1 Hardware Features 2
 - 2.2 Configuration and Connector 2
 - 2.2.1 Configuration 2
 - 2.2.2 Connector Signal Description 3
- 3 Module Technical Parameters..... 4**
 - 3.1 Parameters..... 4
 - 3.2 Default Factory Settings..... 4
- 4 Command Mode user’s guide 5**
 - 4.1 The process of communication..... 5
 - 4.2 Classify of communication packet..... 5
 - 4.2.1 Command packet..... 5
 - 4.2.2 Response packet..... 5
 - 4.2.3 Data Packet..... 6
 - 4.3 Packet Structure..... 6
 - 4.3.1 Packet Identify Code 6
 - 4.3.2 Structure of Command Packet 6
 - 4.3.3 Response packet..... 6
 - 4.3.4 Command Data Packet 7
 - 4.3.5 Response Data Packet 7
- 5 Packet Command Detail Description..... 8**
 - 5.1 Structure of Fingerprint Template Data 8
 - 5.2 Command List 9
 - 5.3 Packet Description..... 11
 - 5.3.1 Verify 11
 - 5.3.2 Identify 12
 - 5.3.3 Enroll..... 13
 - 5.3.4 Enroll One Time 14
 - 5.3.5 Clear Template..... 15
 - 5.3.6 Clear All Template..... 16
 - 5.3.7 Get Empty ID..... 17
 - 5.3.8 Get Template Status 18
 - 5.3.9 Get Broken Template 19
 - 5.3.10 Read Template 20
 - 5.3.11 Write Template 21
 - 5.3.12 Set Security Level 23
 - 5.3.13 Get Security Level..... 24

5.3.14 Set Finger Time Out	25
5.3.15 Get Finger Time Out	26
5.3.16 Set Device ID	26
5.3.17 Get Device ID	27
5.3.18 Get F/W Version	28
5.3.19 Finger Detect.....	29
5.3.20 Set Baudrate.....	30
5.3.21 Set Duplication Check	31
5.3.22 Get Duplication Check	32
5.3.23 Enter Standby Mode	33
5.3.24 Enroll And Store in RAM.....	34
5.3.25 Get Enroll Data	35
5.3.26 Get Feature Data of Captured FP	36
5.3.27 Verify Downloaded Feature with Captured FP	37
5.3.28 Identify Downloaded Feature with Captured FP	38
5.3.29 Get Device Name	39
5.3.30 Sensor LED Control	40
5.3.31 Identify Free.....	41
5.3.32 Set Device Password	42
5.3.33 Verify Device Password	42
5.3.34 Get Enroll Count	43
5.3.35 FP Cancel.....	44
5.3.36 Test Connection	45
5.3.37 Incorrect Command.....	46
5.4 Attentions	47
6 Appendix.....	48
6.1 Acknowledge and Error Code List.....	48
6.2 The Flow of Enroll and Identify	50
6.2.1 The Flow of Enroll.....	50
6.2.2 The Flow of Verify & Identify.....	51

1 Summary

OP-69 are the fingerprint module for secondary development which has integrated fingerprint Collecting and single chip processor together. It features small size, low power consumption, simple ports, high reliability, small fingerprint template (512bytes), large fingerprint capacity, etc. It is convenient to be embedded to user system for realizing clients required fingerprint verification products.

OP-69 outstandingly features self-learning function. During the fingerprint verification process, the latest collected fingerprint features would be integrated into the fingerprint database automatically so that the users would obtain better and better fingerprint verification result.

SM Series module is UART communication interface with adjustable safety level function, fingerprint data reading & writing function, 1:N and 1:1 verification function.

1.1 Main Functions

- ◆ Communication interface :UART
- ◆ Optic sensor is reliable and Low-cost, High ESD Protection
- ◆ 1:N Identification (One-to-Many)
- ◆ 1:1 Verification (One-to-One)
- ◆ High speed fingerprint identification algorithm engine
- ◆ Self study function
- ◆ Fingerprint template data read from /write to FLASH memory function
- ◆ Get Feature Data of Captured fingerprint and Verify/Identify Downloaded Feature with Captured fingerprint(Specially designed for fingerprint stored in IC card)
- ◆ Identify Downloaded Feature with Captured fingerprint
- ◆ Security Level setting
- ◆ Able to set BaudRate/ Device ID/Device Password

1.2 Applications

- ◆ Access control systems
- ◆ Time & Attendance
- ◆ Locks, safes
- ◆ POS, handheld terminals

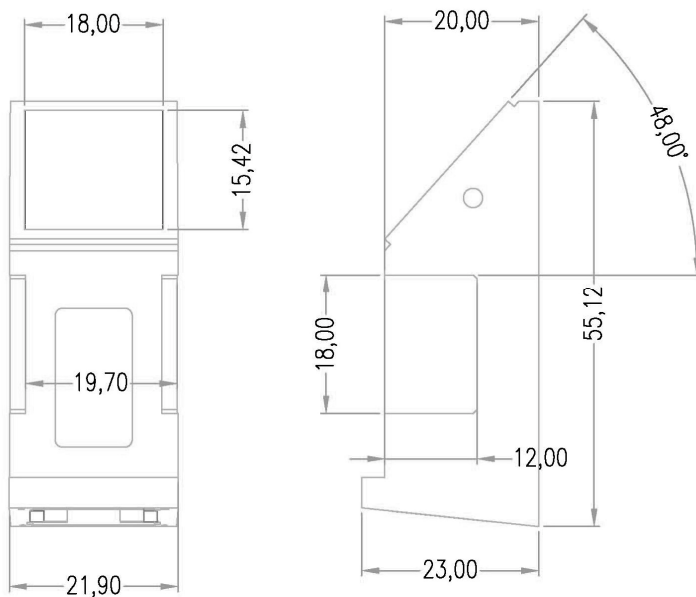
2 Module Hardware Description

2.1 Hardware Features

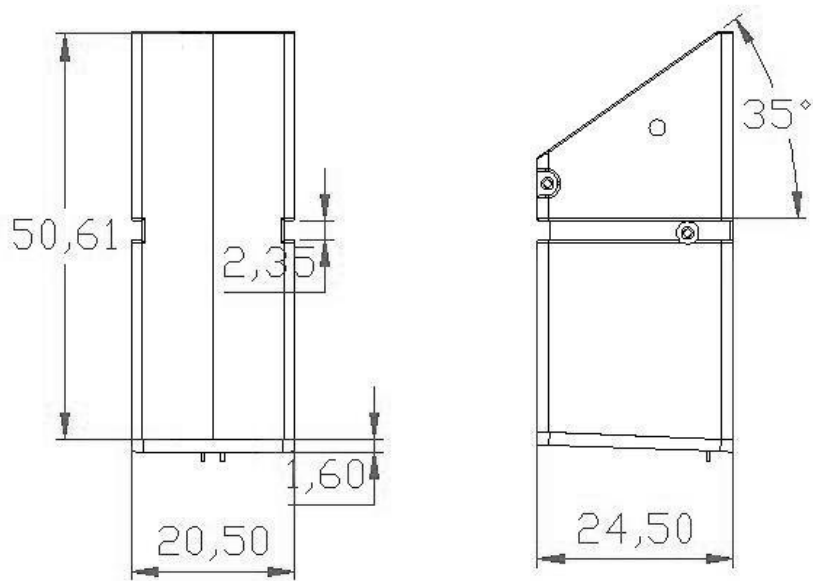
No.	Item	Description
1	Model	OP-69
2	CPU	AS602
3	Sensor	CMOS image sensor: GC0303; Optical total reflection Scanner area: SM12: 18x20mm; SM20:18 x 22 mm
4	Fingerprint image	210 x 250 (pixel)
5	Resolution	500 dpi
6	Power Supply	DC 3.3V±10%
7	Work current	<100mA
8	Communication port	UART

2.2 Configuration and Connector

2.2.1 Configuration



OP-69

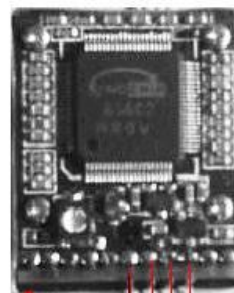
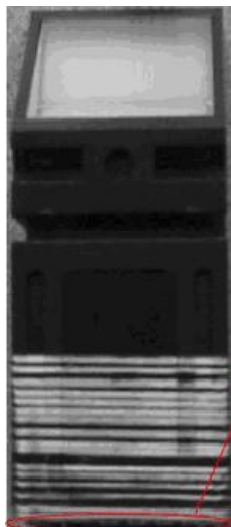


OP-69

2.2.2 Connector Signal Description



3.3V :DC Input
 GND
 Rx :Module Receive, TTL
 Tx :Module Transmit, TTL



Tx
 Rx
 GND
 3.3V

Tx: Module Transmit, TTL
 Rx: Module Receive, TTL
 GND:
 3.3V: DC Input, 3.3V

1. Module Tx: Transmit Output 3.3V TTL Logic
2. Module Rx: Receive Input 3.3V TTL Logic
3. GND: GND
4. DC3.3V: Power Supply DC3.3V±5%

Note: Module power supply is DC3.3V, UART Port is 3.3V TTL

3 Module Technical Parameters

3.1 Parameters

Item	Description
CPU	CPU:AS602 120MHz main frequency, 1M Bits Flash Memory, 128KB SRAM
Sensor	CMOS image sensor: GC0303, Optical total reflection Scan: SM12: 18x20mm; SM20:18 x 22 mm
Fingerprint Capacity	3000~5000(Extensible), Default: 3000
FAR	< 0.001 % (Security Level 3)
FRR	< 0.1 % (Security Level 3)
Matching Method	1:N and 1:1
Fingerprint Template	512 Bytes
Security Level	Level 1~ Level 5, Default setting: Level 3
Speed	Register time< 0.45 s
	1:N (2000 fingerprints) < 0.9s
Communication Port	UART, 3.3V TTL
Communication Parameters	Parity = NONE, One Stop Bit = 1 Flow Control = NONE BaudRate:9600, 19200, 38400, 57600, 115200 (bps) Default: 115200bps
Power Supply	DC 3.3V±10%
Work Current	<100mA
Working Situation	Working Temperature: -20 °C- 60 °C Working Humidity : 20%– 80%

3.2 Default Factory Settings

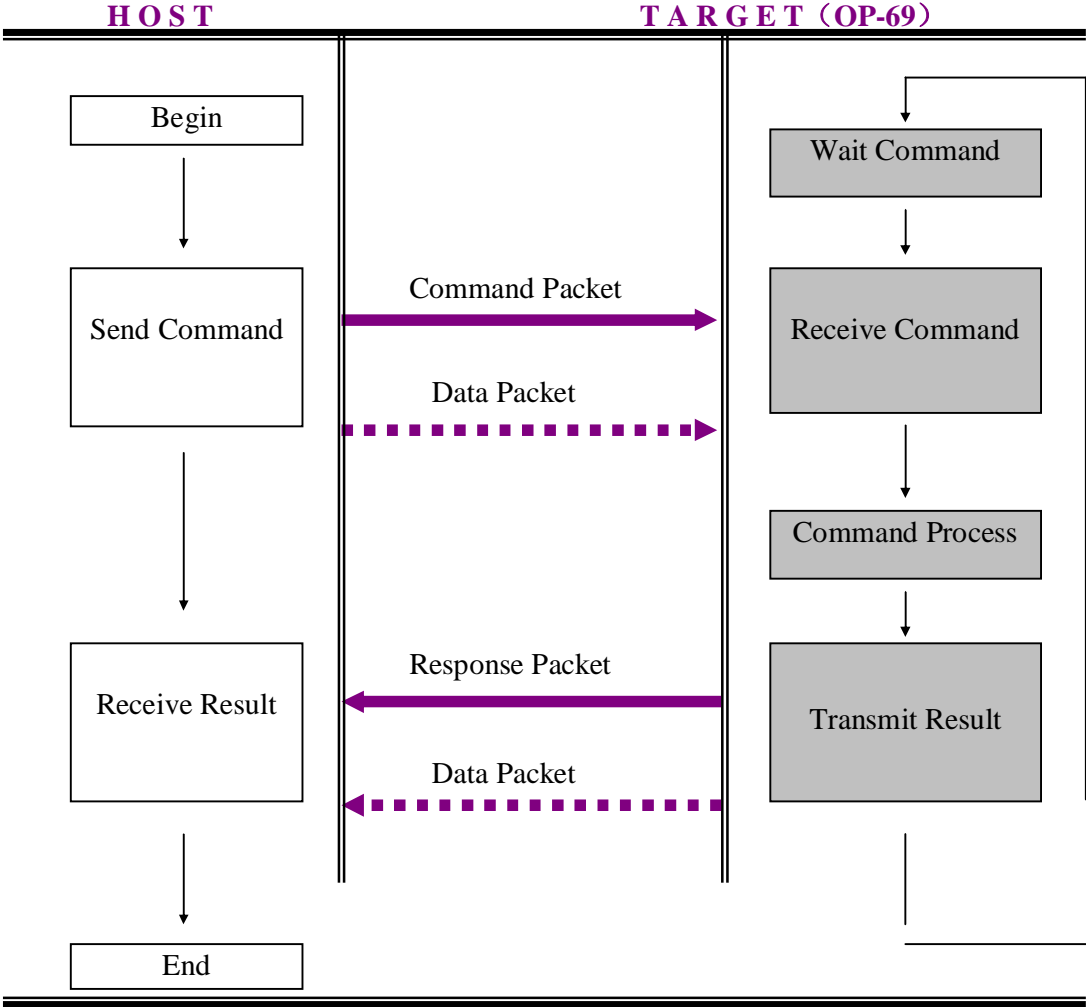
Item	Initial Value
Security Level	3
Finger Print Time Out	5 Seconds
Baud Rate	115200 bps
Duplication Check	ON

4 Command Mode user's guide

The module is used as a slave device. The Master device sends relative commands to control it. The CMD sent by the master and the ACK signal returned by the module.

Command interface: UART (Universal Asynchronous Receiver Transmitter)
115200bps 1 start-bit 1 stop-bit (no check bit)

4.1 The process of communication



4.2 Classify of communication packet

4.2.1 Command packet

Command Packet is the instruction from Host to Target (OP-69),
Total length of the command packet is 24 Bytes

4.2.2 Response packet

Response packet is result of execute command packet, from Target (OP-69) to Host, Total length of the command packet is 24 Bytes

4.2.3 Data Packet

When length of Command Parameter or Data is larger than 16 Bytes, Utilize Data Packet to transmit block Data, the maximum length of Data Packet is 512Bytes

4.3 Packet Structure

4.3.1 Packet Identify Code

Section start 2byte prefix define type of packet

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

4.3.2 Structure of Command Packet

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xAA	L	H	L	H	D0	D1	...	D15	L	H
0	1	2	3	4	5	6	7	...	21	22	23

OFFSET	FIELD	TYPE	SIZE	DESCRIPTION
0	PREFIX	WORD	2byte	Packet Identify code
2	CMD	WORD	2byte	Command Code
4	LEN	WORD	2byte (=n, n < 17)	Length of Command Parameter
6	DATA	Byte array	16byte	Command Parameter
22	CKS	WORD	2byte	Check Sum is the low word of value listed below: offset[0] + offset[1] + ...+offset[21]

4.3.3 Response packet

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	L	H	L	H	L	H	D0	D1	...	D13	L	H
0	1	2	3	4	5	6	7	8	9	...	21	22	23

OFFSET	FIELD	TYPE	SIZE	DESCRIPTION
0	PREFIX	WORD	2byte	Packet Identify code
2	RCM	WORD	2byte	Response Code
4	LEN	WORD	2byte(=n, n < 17)	RET and DATA
6	RET	WORD	2byte	Result Code (0 :success, 1 :fail)
8	DATA	Byte array	14byte	Response Data (n-2 byte)
22	CKS	WORD	2byte	Check Sum is the low word of value listed below: offset[0] + offset[1] + ...+offset[21]

4.3.4 Command Data Packet

PREFIX		CMD		LEN		DATA				CKS	
0xA5	0xA5	L	H	L	H	D0	D1	...	Dn-1	L	H
0	1	2	3	4	5	6	7	...	6+n-1	6+n	6+n+1

OFFSET	FIELD	TYPE	SIZE	DESCRIPTION
0	PREFIX	WORD	2byte	Packet Identify code
2	CMD	WORD	2byte	Command Code
4	LEN	WORD	2byte(=n, n < 512)	Length of DATA
6	DATA	Byte Array	nbyte	Command parameter
6+n	CKS	WORD	2byte	Check Sum is the low word of value listed below: offset[0] + offset[1] + ...+offset[6+n-1]

Before send Command Data packet, Host first send Command packet which set the length of next command data packet in Data Field

4.3.5 Response Data Packet

PREFIX		RCM		LEN		RET		DATA				CKS	
0xA5	0xA5	L	H	L	H	L	H	D0	D1	...	Dn-3	L	H
0	1	2	3	4	5	6	7	8	9	...	6+n-1	6+n	6+n+1

OFFSET	FIELD	TYPE	SIZE	DESCRIPTION
0	PREFIX	WORD	2byte	Packet Identify code
2	CMD	WORD	2byte	Response Code
4	LEN	WORD	2byte(=n, n < 512)	Length of result data (RET + DATA)
6	RET	WORD	2byte	Result code (0 : success 1 : fail)
8	DATA	Byte Array	(n-2) byte	Response data
6+n	CKS	WORD	2byte	Check Sum is the low word of value listed below: offset[0] + offset[1] + ...+offset[6+n-1]

5 Packet Command Detail Description

5.1 Structure of Fingerprint Template Data

$$\text{Template Data}(496\text{Bytes})+\text{Checksum}(2\text{Bytes})=498\text{Bytes}$$

Template Data	Checksum
496 bytes Template Data D0,D1,...D495	2 byte Checksum is the low word of value listed below: D0+D1+...+D495

5.2 Command List

No	Function	Command			Response			
		CMD	LEN	DATA	RCM	LEN	RET	DATA
1	Verify	0x0101	2	Template No.	0x0101	4	0/1	Template No /Error Code
2	Identify	0x0102	0	-	0x0102h	4	0/1	Template No /Error Code
3	Enroll	0x0103	2	Template No.	0x0103	4/6	0/1	Template No /Error Code + Duplication ID
4	Enroll One Time	0x0104	2	Template No.	0x0104	4/6	0/1	Template No /ErrorCode
5	Clear Template	0x0105	2	Template No.	0x0105	4	0/1	Template No /ErrorCode
6	Clear All Template	0x0106	0	-	0x0106	4	0/1	Cleared Template Count /Error Code
7	Get Empty ID	0x0107	0	-	0x0107	4	0/1	Template No /Error Code
8	Get Template Status	0x0108	2	Template No.	0x0108	4	0/1	Template Status /Error Code
9	Get Broken Template	0x0109	0	-	0x0109	6	0/1	Broken Template Count + First Broken Template No /Error Code
10	Read Template	0x010A	2	Template No.	0x010A	(Template Record Size + 4) or (4)	0/1	Template No + Template Record Data /Error Code
11	Write Template	0x010B	Template Record Size + 2	Template No + Template Record Data	0x010B	4	0/1	Template No /Error Code
12	Set Security Level	0x010C	2	Security Level Value	0x010C	4	0/1	Security Level Value /Error Code
13	Get Security Level	0x010D	0	-	0x010D	4	0	Security Level Value
14	Set Finger Time Out	0x010E	2	Time Out Value	0x010E	4	0/1	TimeOut Value /Error Code
15	Get Finger Time Out	0x010F	0	-	0x010F	4	0	TimeOut Value
16	Set Device ID	0x0110	2	Device ID	0x0110	4	0	Device ID
17	Get Device ID	0x0111	0	-	0x0111	4	0/1	Device ID /Error Code
18	Get F/W Version	0x0112	0	-	0x0112	4	0	F/W Version
19	Finger Detect	0x0113	0	-	0x0113	4	0	Detect Result

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

5.3 Packet Description

5.3.1 Verify

[Function] One to one match

Verify the appointed template to live scanning fingerprint, then response result.

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x0101
LEN	2
DATA	Template No.
CKS	Check Sum
Response Packet	
PREFIX	0x55AA
RCM	0x0101
LEN	4
RET	ERR_SUCCESS or ERR_FAIL
DATA	2byte Success: Template No. Fail: Error Code is as follow ERR_INVALID_TMPL_NO ERR_TMPL_EMPTY ERR_TIME_OUT ERR_BAD_QUALITY and so on
CKS	Check Sum

[Command Example]

Command Packet: 1:1 verify by ID 1 with scanning fingerprint

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xA A	0x01	0x01	0x02	0x00	0x01	0x00	0x00	0x04	0x01

Response Packet: Captured fingerprint within Timeout,

Firstly return to (GD_NEED_RELEASE_FINGER)

Then return to Result

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

5.3.2 Identify

[Function] One to many match

The live scanning fingerprint matches to all templates stored in FLASH memory, and then respond the result.

Please refer to appendix about the flow of identify

[Command and Response]

Command Packet		
PREFIX	0xAA55	
CMD	0x0102	
LEN	0	
DATA	null	
CKS	Check Sum	
Response Packet		
PREFIX	0x55AA	
RCM	0x0102	
LEN	4	
RET	ERR_SUCCESS or ERR_FAIL	
DATA	2byte	Success: ID number of the matched template or Fail: Error Code is as follow ERR_ALL_TMPL_EMPTY ERR_TIME_OUT ERR_BAD_QUALITY and so on
CKS	Check Sum	

[Command Example]

Command Packet :

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xA A	0x02	0x01	0x00	0x00	0x00	0x00	0x00	0x02	0x01

Response Packet: To (GD_NEED_RELEASE_FINGER) , then return to result

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

5.3.3 Enroll

[Function] Enroll

In the process of enroll, User's finger must be press on the reader for 3 times, each time module get template temporary and storage in RAM, if the three templates are correct, the module generalize the three templates to one template then write to Flash memory.

Please refer to appendix about the flow of Enroll

[Command and Response]

Command Packet		
PREFIX	0xAA55	
CMD	0x0103	
LEN	2	
DATA	Enrollment Template No.	
CKS	Check Sum	
Response Packet		
PREFIX	0x55AA	
RCM	0x0103	
LEN	6	
RET	ERR_SUCCESS or ERR_FAIL	
DATA	2byte	Success: GD_NEED_FIRST_SWEEP GD_NEED_SECOND_SWEEP GD_NEED_THIRD_SWEEP GD_NEED_RELEASE_FINGER New enrollment ID Fail: Error Code is as follow ERR_INVALID_TMPL_NO ERR_TMPL_NOT_EMPTY ERR_TIME_OUT ERR_GENERALIZE and so on
	2byte	0 or ERR_DUPLICATION_ID
CKS	Check Sum	

[Command Example]

Command Packet : Enroll 1st fingerprint

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

Response Packet: Enroll 1st fingerprint success

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

5.3.4 Enroll One Time

[Function] Enroll One Time

User's finger only press on the fingerprint reader for one times

[Command and Response]

Command Packet		
PREFIX	0xAA55	
CMD	0x0104	
LEN	2	
DATA	Enrollment Template No.	
CKS	Check Sum	
Response Packet		
PREFIX	0x55AA	
RCM	0x0104	
LEN	6	
RET	ERR_SUCCESS or ERR_FAIL	
DATA	2byte	Success: Template No. Fail: Error Code is as follow ERR_INVALID_TMPL_NO ERR_TMPL_NOT_EMPTY ERR_TIME_OUT ERR_BAD_QUALITY and so on
	2byte	0 or ERR_DUPLICATION_ID
CKS	Check Sum	

[Command Example]

Command Packet: Enroll 1st fingerprint

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

Response Packet: Enroll 1st fingerprint success

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

5.3.5 Clear Template

[Function]

Delete fingerprint data with specified ID from database. After this command is executed, fingerprint data with specified ID are deleted immediately.

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x0105
LEN	2
DATA	Template No. which will be deleted
CKS	Check Sum
Response Packet	
PREFIX	0x55AA
RCM	0x0105
LEN	4
RET	ERR_SUCCESS or ERR_FAIL
DATA	2byte Success: Template No. which will be deleted Fail: Error Code is as follow ERR_INVALID_TMPL_NO ERR_TMPL_EMPTY and so on
CKS	Check Sum

[Command Example]

Command Packet: Delete 1st fingerprint

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

Response Packet: Delete 1st fingerprint success

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

5.3.6 Clear All Template

[Function] Clear All Template

Delete all fingerprint data in database. After this command is executed, all fingerprint data in database are deleted immediately.

[Command and Response]

Command Packet		
PREFIX	0xAA55	
CMD	0x0106	
LEN	0	
DATA	Null	
CKS	Check Sum	
Response Packet		
PREFIX	0x55AA	
RCM	0x0106	
LEN	4	
RET	ERR_SUCCESS or ERR_FAIL	
DATA	2byte	Success: Total Number of template which have been deleted Fail: Error Code is as follow ERR_ALL_TMPL_EMPTY and so on
CKS	Check Sum	

[Command Example]

Command Packet : Clear all template

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xA A	0x06	0x01	0x00	0x00	0x00	0x00	0x00	0x06	0x01

Response Packet: Clear all template Success

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

5.3.7 Get Empty ID

[Function] Get Empty ID

Get the first template No. it can be used for storage fingerprint template

[Command and Response]

Command Packet		
PREFIX	0xAA55	
CMD	0x0107	
LEN	0	
DATA	Null	
CKS	Check Sum	
Response Packet		
PREFIX	0x55AA	
RCM	0x0107	
LEN	4	
RET	ERR_SUCCESS or ERR_FAIL	
DATA	2byte	Success: The template No. that can be useable Fail: Error Code is as follow ERR_EMPTY_ID_NOEXIST and so on
CKS	Check Sum	

[Command Example]

Command Packet : Get empty ID

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xA A	0x07	0x01	0x00	0x00	0x00	0x00	0x00	0x07	0x01

Response Packet: Get empty ID Success

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

5.3.8 Get Template Status

[Function]

Check the specified Template No. whether can be stored.

[Command and Response]

Command Packet		
PREFIX	0xAA55	
CMD	0x0108	
LEN	2	
DATA	Template No.	
CKS	Check Sum	
Response Packet		
PREFIX	0x55AA	
RCM	0x0108	
LEN	4	
RET	ERR_SUCCESS or ERR_FAIL	
DATA	2byte	Success: GD_TEMPLATE_NOT_EMPTY GD_TEMPLATE_EMPTY Fail: Error Code is as follow ERR_INVALID_TMPL_NO and so on
CKS	Check Sum	

[Command Example]

Command Packet: Get the 1st ID template status

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

Response Packet: Get the 1st ID template status Success

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

5.3.9 Get Broken Template

[Function]

Check fingerprint template Database whether is damage, although Database damage is hardly ever. For the broken template data, you can delete the template and then enroll again.

[Command and Response]

Command Packet		
PREFIX	0xAA55	
CMD	0x0109	
LEN	0	
DATA	Null	
CKS		
Response Packet		
PREFIX	0x55AA	
RCM	0x0109	
LEN	6	
RET	ERR_SUCCESS or ERR_FAIL	
DATA	D0, D1	Success: Total number of broken template, 0 is no template are damage Fail: Error Code
	D2, D3	Success: The first template No. of broken template Fail:0
CKS	Check Sum	

[Command Example]

Command Packet: Get broken template

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xA A	0x09	0x01	0x00	0x00	0x00	0x00	0x00	0x09	0x01

Response Packet: Get broken template Success

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

5.3.10 Read Template

[Function]

Read fingerprint Template data with specified Template No. from the module

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x010A
LEN	2
DATA	Template No.
CKS	Check Sum
Response Packet	
PREFIX	0x55AA
RCM	0x010A
LEN	4
RET	ERR_SUCCESS or ERR_FAIL
DATA	Success: Length of next Data Response Packet (Template Record Size + 2) Fail: Error Code is as follow ERR_INVALID_TMPL_NO ERR_EMPTY_TMPL and so on
CKS	Check Sum
Response Data Packet	
PREFIX	0x5AA5
RCM	0x010A
LEN	Template Record Size + 4
RET	ERR_SUCCESS

DATA	Template No.(2byte) + Template Record Data
CKS	Check Sum

[Command Example]

Command Packet: Read 1st template data

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

Response Packet: Return Template No.(2byte) and Template Record Data

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

5.3.11 Write Template

[Function]

Download fingerprint data with specified ID from host to module, then write to fingerprint template Database

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x010B
LEN	2
DATA	Template Record Size
CKS	Check Sum
Response Packet	
PREFIX	0x55AA
RCM	0x010B
LEN	4
RET	ERR_SUCCESS or ERR_FAIL
DATA	2byte Success: 0 Fail: Error Code is as follow ERR_INVALID_PARAM and so on
CKS	Check Sum

Command Data Packet	
PREFIX	0xA55A
CMD	0x010B
LEN	Template Record Size + 2
DATA	Template No.(2byte) + Template Record Data
CKS	Check Sum
Response Data Packet	
PREFIX	0x5AA5
RCM	0x010B
LEN	4
RET	ERR_SUCCESS or ERR_FAIL
DATA	2byte Success: Template No. Fail: ERR_INVALID_TMPL_NO ERR_INVALID_PARAM and so on
CKS	Check Sum

[Command Example]

Command Packet: Specified Template Record Size

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

Response Packet: Inform HOST has entered into template data accepting status

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

Command Packet: Template data written into ID 1

PREFIX		CMD		LEN		DATA				CKS	
0x5A	0xA5	0x0B	0x01	0xF4	0x01	0x01	0x00	498Byte data			CKS

Response Packet: Write template data success

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

5.3.12 Set Security Level

[Function]

Set up threshold of fingerprint identification engine, Integer of 1-5 can be selected, 1 is the lowest identification level and 5 is the highest identification level, Default is 3

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x010C
LEN	2
DATA	Value of Security Level
CKS	Check Sum
Response Packet	
PREFIX	0x55AA
RCM	0x010C
LEN	4
RET	ERR_SUCCESS or ERR_FAIL
DATA	Success: Value Of Security Level Fail: ERR_INVALID_SEC_VAL and so on
CKS	Check Sum

[Security Level Details]

Security Level	Parameters	
Level 1	FAR (False Acceptance Rate)	0.01%
	FRR (False Rejection Rate)	0.005%
Level 2	FAR (False Acceptance Rate)	0.003%
	FRR (False Rejection Rate)	0.01%
Level 3	FAR (False Acceptance Rate)	0.001 %
	FRR (False Rejection Rate)	0.1 %
Level 4	FAR (False Acceptance Rate)	0.0003%
	FRR (False Rejection Rate)	0.5%
Level 5	FAR (False Acceptance Rate)	0.0001%
	FRR (False Rejection Rate)	1%

[Command Example]

Command Packet : Set Security level as 3

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

Response Packet: Set Security level as 3 Success

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

5.3.13 Get Security Level

[Function]

Read the value of Security Level from target module

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x010D
LEN	0
DATA	Null
CKS	Check Sum
Response Packet	
PREFIX	0x55AA
RCM	0x010D
LEN	4
RET	ERR_SUCCESS or ERR_FAIL
DATA	Success: Value of Security Level this module Fail: Error code
CKS	Check Sum

[Command Example]

Command Packet: Get security level

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xA A	0x0D	0x01	0x00	0x00	0x00	0x00	0x00	0x0D	0x01

Response Packet: Get Security Level Success

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

5.3.14 Set Finger Time Out

[Function] Set Finger Time out to Module

The process of execute “Verify”, “Identify”, “Enroll”, “Enroll One Time” command .
Module waiting for capture fingerprint image, if time out then return ERR_TIME_OUT
1-10S can be selected, Default is 5s

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x010E
LEN	2
DATA	Value of Time Out
CKS	Check Sum
Response Packet	
PREFIX	0x55AA
RCM	0x010E
LEN	4
RET	ERR_SUCCESS or ERR_FAIL
DATA	Success: Value of Time Out Fail: ERR_INVALID_TIME_OUT And so on
CKS	Check Sum

[Command Example]

Command Packet : Set Time Out time as 10s

PREFIX		CMD		LEN		DATA				CKS	
0xA 0x55	A	0x0E	0x01	0x02	0x00	0x00	0x00	0x00	0x1A	0x01

Response Packet: Set Time Out time as 10s Success

PREFIX		RCM		LEN		RET		DATA				CKS	
0xA A	0x55	0x0E	0x01	0x04	0x00	0x00	0x00	0x0 A	0x00	...	0x00	0x1C	0x01

5.3.15 Get Finger Time Out

[Function] Read Value of Finger Time Out from module

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x010F
LEN	0
DATA	Null
CKS	Check Sum
Response Packet	
PREFIX	0x55AA
RCM	0x010F
LEN	4
RET	ERR_SUCCESS or ERR_FAIL
DATA	Value of Time Out
CKS	Check Sum

[Command Example]

Command Packet : Get Time Out Value

PREFIX	CMD	LEN	DATA	CKS
0xA A 0x55	0x0F 0x01	0x00 0x00	0x00 0x00 0x00	0x0F 0x01

Response Packet: Get Time Out Value Success

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

5.3.16 Set Device ID

[Function] Set Device ID Number, 1-254 can be selected, Default is 1

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x0110
LEN	2
DATA	Device ID
CKS	Check Sum
Response Packet	

PREFIX	0x55AA
RCM	0x0110
LEN	4
RET	ERR_SUCCESS or ERR_FAIL
DATA	Success: Device ID Fail: ERR_INVALID_PARAM and so on
CKS	Check Sum

[Command Example]

Command Packet : Set Device ID as 1

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

Response Packet: Set Device ID as 1 Success

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

5.3.17 Get Device ID

[Function] Read Device ID from Module

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x0111
LEN	0
DATA	Null
CKS	Check Sum
Response Packet	
PREFIX	0x55AA
RCM	0x0111
LEN	4
RET	ERR_SUCCESS or ERR_FAIL
DATA	Success: Device ID this module Fail: Error Code
CKS	Check Sum

[Command Example]

Command Packet : Get Device ID

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xA A	0x11	0x01	0x00	0x00	0x00	0x00	0x00	0x11	0x01

Response Packet: Get Device ID Success

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

5.3.18 Get F/W Version

[Function] Get Firmware Version

[Command and Response]

Command Packet		
PREFIX	0xAA55	
CMD	0x0112	
LEN	0	
DATA	Null	
CKS	Check Sum	
Response Packet		
PREFIX	0x55AA	
RCM	0x0112	
LEN	4	
RET	ERR_SUCCESS or ERR_FAIL	
DATA	D0	Version Major
	D1	Version Minor
CKS	Check Sum	

[Command Example]

Command Packet: Get F/W Version

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xA A	0x12	0x01	0x00	0x00	0x00	0x00	0x00	0x12	0x01

Response Packet: Get F/W Version 2.9 Success

PREFIX		RCM		LEN		RET		DATA				CKS	
0xA A	0x55	0x12	0x01	0x04	0x00	0x00	0x00	0x02	0x09	...	0x00	0x0D	0x01

5.3.19 Finger Detect

[Function] Detect whether a finger is placed on the sensor.

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x0113
LEN	0
DATA	Null
CKS	Check Sum
Response Packet	
PREFIX	0x55AA
RCM	0x0113
LEN	4
RET	ERR_SUCCESS
DATA	1:Detect an finger on the sensor or 0:no finger on sensor
CKS	Check Sum

[Command Example]

Command Packet: Detect whether a finger is placed on the sensor

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xA A	0x13	0x01	0x00	0x00	0x00	0x00	0x00	0x13	0x01

Response Packet: Detect result: a finger is placed on the sensor

5.3.20 Set Baudrate

[Function] Set UART Baudrate

New value is active by reset or power on

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x0114
LEN	2
DATA	Baudrate Index 1 : 9600bps 2 : 19200bps 3 : 38400bps 4 : 57600bps 5 : 115200bps
CKS	Check Sum
Response Packet	
PREFIX	0x55AA
RCM	0x0114
LEN	4
RET	ERR_SUCCESS or ERR_FAIL
DATA	Success: Baudrate Index Fail: ERR_INVALID_BAUDRATE and so on
CKS	Check Sum

[Command Example]

Command Packet : Set Baudrate 9600

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

Response Packet: Set Baudrate 9600 Success

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

5.3.21 Set Duplication Check

[Function]

Setup Enable/Disable fingerprint duplication check in the period of “Enroll” or “Enroll One Time” command,

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x0115
LEN	2
DATA	1:Enable,0:Disable
CKS	Check Sum
Response Packet	
PREFIX	0x55AA
RCM	0x0115
LEN	4
RET	ERR_SUCCESS or ERR_FAIL
DATA	Success: Enable/Disable Fail: ERR_INVALID_DUP_VAL ect.
CKS	Check Sum

[Command Example]

Command Packet : Set Duplication Check Enable

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

Response Packet: Set Duplication Check Enable Success

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

5.3.22 Get Duplication Check

[Function] Read module Duplication Check state

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x0116
LEN	0
DATA	Null
CKS	Check Sum
Response Packet	
PREFIX	0x55AA
RCM	0x0116
LEN	4
RET	ERR_SUCCESS
DATA	1 or 0 1:Enable,0:Disable
CKS	Check Sum

[Command Example]

Command Packet : Get Duplication Check

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xA A	0x16	0x01	0x00	0x00	0x00	0x00	0x00	0x16	0x01

Response Packet: Get Duplication Check as Enable

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

5.3.23 Enter Standby Mode

[Function] Lead module into standby mode, standby current Max 5mA

Reset or power on will lead module from standby mode to active mode

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x0117
LEN	0
DATA	Null
CKS	Check Sum
Response Packet	
PREFIX	0x55AA
RCM	0x0117
LEN	2
RET	ERR_SUCCESS
DATA	Null
CKS	Check Sum

Note: Send this command (Enter Standby Mode) before cut off the module power supply, then the module would enter into standby mode.

[Command Example]

Command Packet: Set Standby mode

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xA A	0x17	0x01	0x00	0x00	0x00	0x00	0x00	0x17	0x01

Response Packet: Set Standby mode Success

PREFIX		RCM		LEN		RET		DATA				CKS	
0xA A	0x55	0x17	0x01	0x04	0x00	0x00	0x00	0x01	0x00	...	0x00	0x0B	0x01

5.3.24 Enroll And Store in RAM

[Function]

The same as “Enroll” command, In the process of Enroll and Store RAM, User’s finger must be put on the reader for 3 times, each time module get template temporary and storage in RAM, if the three templates are correct, the module generalize the three templates to one template then write to RAM memory temporarily.

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x0118
LEN	0
DATA	null
CKS	Check Sum
Response Packet	
PREFIX	0x55AA
RCM	0x0118
LEN	4
RET	ERR_SUCCESS or ERR_FAIL
DATA	<p>Success:</p> <p>GD_NEED_FIRST_SWEEP 、</p> <p>GD_NEED_SECOND_SWEEP</p> <p>GD_NEED_THIRD_SWEEP 、</p> <p>GD_NEED_RELEASE_FINGER</p> <p>0</p> <p>Fail:Error Code is as follow</p> <p>ERR_INVALID_TMPL_NO</p> <p>ERR_TMPL_NOT_EMPTY</p> <p>ERR_TIME_OUT</p> <p>ERR_GENERALIZE</p> <p>and so on</p>
CKS	Check Sum

[Command Example]

Command Packet : Enroll and store in RAM

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xA A	0x18	0x01	0x00	0x00	0x00	0x00	0x00	0x18	0x01

Response Packet: Enroll and store in RAM Success

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

5.3.25 Get Enroll Data

[Function]

Get template data in RAM that execute the “Enroll and Store in RAM” command.

Please refer to “Read Template” command

Before send “Get Enroll Data” command, module must execute the “ Enroll And Store in RAM ”command

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x0119
LEN	0
DATA	null
CKS	Check Sum
Response Packet	
PREFIX	0x55AA
RCM	0x0119
LEN	4
RET	ERR_SUCCESS or ERR_FAIL
DATA	Success: Template Record Size for next Response Data Packet Fail: Error Code
CKS	Check Sum
If success then module send Response Data Packet	

PREFIX	0x5AA5
RCM	0x0119
LEN	Template Record Size + 2
RET	ERR_SUCCESS
DATA	Template Record Data
CKS	Check Sum

[Command Example]

Command Packet : Get enrolled template data

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xA A	0x19	0x01	0x00	0x00	0x00	0x00	0x00	0x19	0x01

Response Packet: Return to the template record size and data

PREFIX		RCM		LEN		RET		DATA				CKS	
0xA A	0x55	0x19	0x01	0x04	0x00	0x00	0x00	0xF2	0x01	...	0x00	0x10	0x02
0xA5	0x5A	0x19	0x01	0xF4	0x01	0x00	0x00	498 Bytes data				CKS	

5.3.26 Get Feature Data of Captured FP

[Function] Get template data when capture an fingerprint

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x011A
LEN	0
DATA	null
CKS	Check Sum
Response Packet	
PREFIX	0x55AA
RCM	0x011A
LEN	4
RET	ERROR_SUCCESS or ERR_FAIL
DATA	Success: Template Record Size for next Response Data Packet Fail: Error Code
CKS	Check Sum
If success then module send Response Data Packet	
PREFIX	0x5AA5
RCM	0x011A
LEN	Template Record Size + 2

RET	ERR_SUCCESS
DATA	Template Record Data
CKS	Check Sum

[Command Example]

Command Packet: Get Feature Data of Captured FP

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

Response Packet: Return to the template record size and data

PREFIX		RCM		LEN		RET		DATA				CKS	
0xA A	0x55	0x1A	0x01	0x04	0x00	0x00	0x00	0xF2	0x01	...	0x00	0x11	0x02
0xA5	0x5A	0x1A	0x01	0xF4	0x01	0x00	0x00	498 Bytes data				CKS	

5.3.27 Verify Downloaded Feature with Captured FP

[Function] Verify downloaded template data with live captured fingerprint

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x011B
LEN	2
DATA	Template Record Size
CKS	Check Sum
Response Packet	
PREFIX	0x55AA
RCM	0x011B
LEN	4
RET	ERR_SUCCESS or ERR_FAIL
DATA	Success:0 , Fail: Error Code
CKS	Check Sum
If Success then host send Command Data Packet to module	
PREFIX	0xA55A
CMD	0x011B
LEN	Template Record Size
DATA	Template Record Data
CKS	Check Sum

Response Data Packet	
PREFIX	0x5AA5
RCM	0x011B
LEN	4
RET	ERR_SUCCESS or ERR_FAIL
DATA	Success:0 , Fail: Error Code
CKS	Check Sum

[Command Example]

Command Packet : FP templates data size

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

Response Packet: Check FP templates data size correct

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

5.3.28 Identify Downloaded Feature with Captured FP

[Function] Download 2 templates and identify with live captured fingerprint

[Command and Response]

Command packet	
PREFIX	0xAA55
CMD	0x011C
LEN	2
DATA	Template Record Size
CKS	Check Sum
Command Data packet	
PREFIX	0xA55A
CMD	0x011C
LEN	Template Record Data + 2
DATA	Template Index No.(2Byte) + Template Record Data
CKS	Check Sum
Response Packet	
PREFIX	0x55AA
RCM	0x011C
LEN	4
RET	ERR_SUCCESS or ERR_FAIL
DATA	Success:0 , Fail: Error Code
CKS	Check Sum

[Command Example]

Command Packet: FP templates data size

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

Response Packet: Check FP templates data size correct

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

Command Packet: Download 1st template feature data

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

Response Packet: Download 1st template feature data Success

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

Command Packet: Download 1st template feature data

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

Response Packet: Download 2 templates and identify with scanning fingerprint success

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

5.3.29 Get Device Name

[Function] Read Device name of target

This device name is "STO20-OEM".

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x0121
LEN	0

DATA	null
CKS	Check Sum
Response Packet	
PREFIX	0x55AA
RCM	0x0121
LEN	16
RET	ERR_SUCCESS or ERR_FAIL
DATA	Success: "STO20-OEM", Fail: Error Code
CKS	Check Sum

[Command Example]

Command Packet: Get device name

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xA A	0x21	0x01	0x00	0x00	0x00	0x00	0x00	0x21	0x01

Response Packet: Get device name Success

PREFIX		RCM		LEN		RET		DATA	CKS
0xA A	0x55	0x21	0x01	0x10	0x00	0x00	0x00	ASCII format Device Name	CKS

5.3.30 Sensor LED Control

[Function] On or Off the sensor LED control

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x0124
LEN	2
DATA	LED (1: on; 0: off)
CKS	Check Sum
Response Packet	
PREFIX	0x55AA
RCM	0x0124
LEN	4
RET	Success: ERR_SUCCESS
DATA	0
CKS	Check Sum

[Command Example]

Command Packet: Set LED on

PREFIX		CMD		LEN		DATA				CKS	
0x5	0xA	0x24	0x0	0x0	0x0	0x01	0x00	0x00	0x2	0x0
5	A	1	1	2	0					7	1

Response Packet: Set LED on success

PREFIX		RCM		LEN		RET		DATA				CKS	
0xA	0x5	0x24	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x2	0x0
A	5	1	1	4	0	0	0	0	0	0	0	8	1

5.3.31 Identify Free

[Function]

Identify 1:N for continuously identifying input fingerprints with the enrolled until received FPCancel command.

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x0125
LEN	0
DATA	Null
CKS	Check Sum
Response Packet	
PREFIX	0x55AA
RCM	0x0125
LEN	4
RET	ERR_SUCCESS or ERR_FAIL
DATA	Success: GD_NEED_RELEASE_FINGER / Template ID, Fail: Error Code
CKS	Check Sum

[Command Example]

Command Packet: Identify free

PREFIX		CMD		LEN		DATA				CKS	
0x5	0xA	0x25	0x0	0x0	0x0	0x00	0x00	0x00	0x2	0x0
5	A	1	1	0	0					5	1

Response Packet: Identify free success

PREFIX		RCM		LEN		RET		DATA				CKS	
0xA	0x5	0x25	0x0	0x0	0x0	0x0	0x0	0xF	0xF	0x0	0x0	0x1	0x0
A	5	1	1	4	0	0	0	4	F	0	0	C	3
0xA	0x5	0x25	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x2	0x0
A	5	1	1	4	0	0	0	1	0	0	0	A	1

5.3.32 Set Device Password

[Function]

Set device password, can't read if has set password

Password must be 14 byte, if less than 14 byte, then it would fail

For Password operation please refer "Verify Device Password".

Cancel the password, all 14 byte should be 0x00

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x0126
LEN	14
DATA	14 byte password
CKS	Check Sum
Response Packet	
PREFIX	0x55AA
RCM	0x0126
LEN	4
RET	Success: ERR_SUCCESS, Fail: ERR_FAIL
DATA	0/Error Code
CKS	Check Sum

[Command Example]

Command Packet: Set device password

PREFIX		CMD		LEN		DATA	CKS
0x55	0xA A	0x26	0x01	0x0E	0x00	14byte ASCII	CKS

Response Packet: Set device password Success

PREFIX		RCM		LEN		RET		DATA				CKS	
0xA A	0x55	0x26	0x01	0x04	0x00	0x00	0x00	0x00	0x00	...	0x00	0x2A	0x01

5.3.33 Verify Device Password

[Function]

Verify whether the input password is correct with the device password

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x0127
LEN	14

DATA	14 byte password
CKS	Check Sum
Response Packet	
PREFIX	0x55AA
RCM	0x0127
LEN	4
RET	Success: ERR_SUCCESS, Fail: ERR_FAIL
DATA	0 / ERR_NOT_AUTHORIZED
CKS	Check Sum

[Command Example]

Command Packet: Verify Device Password

PREFIX		CMD		LEN		DATA	CKS
0x55	0xA A	0x27	0x01	0x0E	0x00	14byte ASCII	CKS

Response Packet: Verify Device Password Success

PREFIX		RCM		LEN		RET		DATA				CKS	
0xA A	0x55	0x27	0x01	0x04	0x00	0x00	0x00	0x00	0x00	...	0x00	0x2B	0x01

5.3.34 Get Enroll Count

[Function]

Get the total count of enroller templates

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x0128
LEN	0
DATA	NULL
CKS	Check Sum
Response Packet	
PREFIX	0x55AA
RCM	0x0128
LEN	4
RET	Success: ERR_SUCCESS, Fail: ERR_FAIL
DATA	Success: Total amount of enrolled template Fail: Error Code
CKS	0x55AA

[Command Example]

Command Packet: Get Total Count of enrolled template

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xA A	0x28	0x01	0x00	0x00	0x00	0x00	0x00	0x28	0x01

Response Packet: Get Total Count of enrolled template Success

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

5.3.35 FP Cancel

[Function] Cancel all operation with fingerprint such as “Verify”, “Identify”, “Enroll”, “Enroll One Time”, “Enroll And Store in RAM”, “Get Feature Data of Captured FP”, “Verify Downloaded Feature with Captured FP”, “Identify Downloaded Feature with Captured FP”command

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x0130
LEN	0
DATA	null
CKS	Check Sum
Response Packet	
PREFIX	0x55AA
RCM	0x0130
LEN	4
RET	ERR_SUCCESS
DATA	Null
CKS	Check Sum

[Command Example A- Send FP Cancel at no operation status]

Command Packet : Cancel command

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xA A	0x30	0x01	0x00	0x00	0x00	0x00	0x00	0x30	0x01

Response Packet: Cancel success

PREFIX		RCM		LEN		RET		DATA				CKS	
0xA	A	0x30	0x01	0x04	0x00	0x00	0x00	0x00	0x00	...	0x00	0x34	0x01

[Command Example B- Send FP Cancel when Verify, Identify, Enroll, Enroll One Time, Enroll And Store in RAM, Get Feature Data of Captured FP, Verify Downloaded Feature with Captured FP, Identify Downloaded Feature with Captured FP]

Command Packet : FP cancel when Enroll

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xA A	0x30	0x01	0x00	0x00	0x00	0x00	0x00	0x30	0x01

Response Packet: FP cancel Success when Enroll

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

5.3.36 Test Connection

[Function] Check connection state between Host and Target

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x0150
LEN	0
DATA	null
CKS	Check
Response Packet	
PREFIX	0x55AA
RCM	0x0150
LEN	4
RET	ERR_SUCCESS or ERR_FAIL
DATA	Null
CKS	Check Sum

[Command Example]

Command Packet : Test Connection

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xA A	0x50	0x01	0x00	0x00	0x00	0x00	0x00	0x50	0x01

Response Packet: Connection Success

PREFIX		RCM		LEN		RET		DATA				CKS	
0xA A	0x55	0x50	0x01	0x04	0x00	0x00	0x00	0x00	0x00	...	0x00	0x54	0x01

5.3.37 Incorrect Command

[Function] Response to host that the command is incorrect, as a result of interference effect[

Command and Response]

Response Packet	
PREFIX	0x55AA
RCM	0x0160
LEN	4
RET	ERR_SUCCESS
DATA	Null
CKS	Check Sum

[Command Example]

Command Packet : Send one incorrect command

PREFIX		CMD		LEN		DATA				CKS	
0x55	0xA A	0x01	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x01

Response Packet: Response of incorrect command

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

5.4 Attentions

- A. The module would automatically turn on the sensor LED when execute Enroll, Enroll One Time, Identify, Verify, Enroll And Store in RAM, Get Feature Data of Captured FP, Verify Downloaded Feature with Captured FP, Identify Downloaded Feature with Captured FP. If get the correct fingerprint, LED would be off; unless the LED would keep on. User can control the LED on or off according to their demand. It is also available for adopting Sensor LED Control command to control the LED on or off.**

- B. This module features device password function. Once set the password, all the commands are unavailable without password authentication. So it can protect the enrolled FP data as anti-theft function with high security. If not set password, all the commands can be available without password.**

6 Appendix

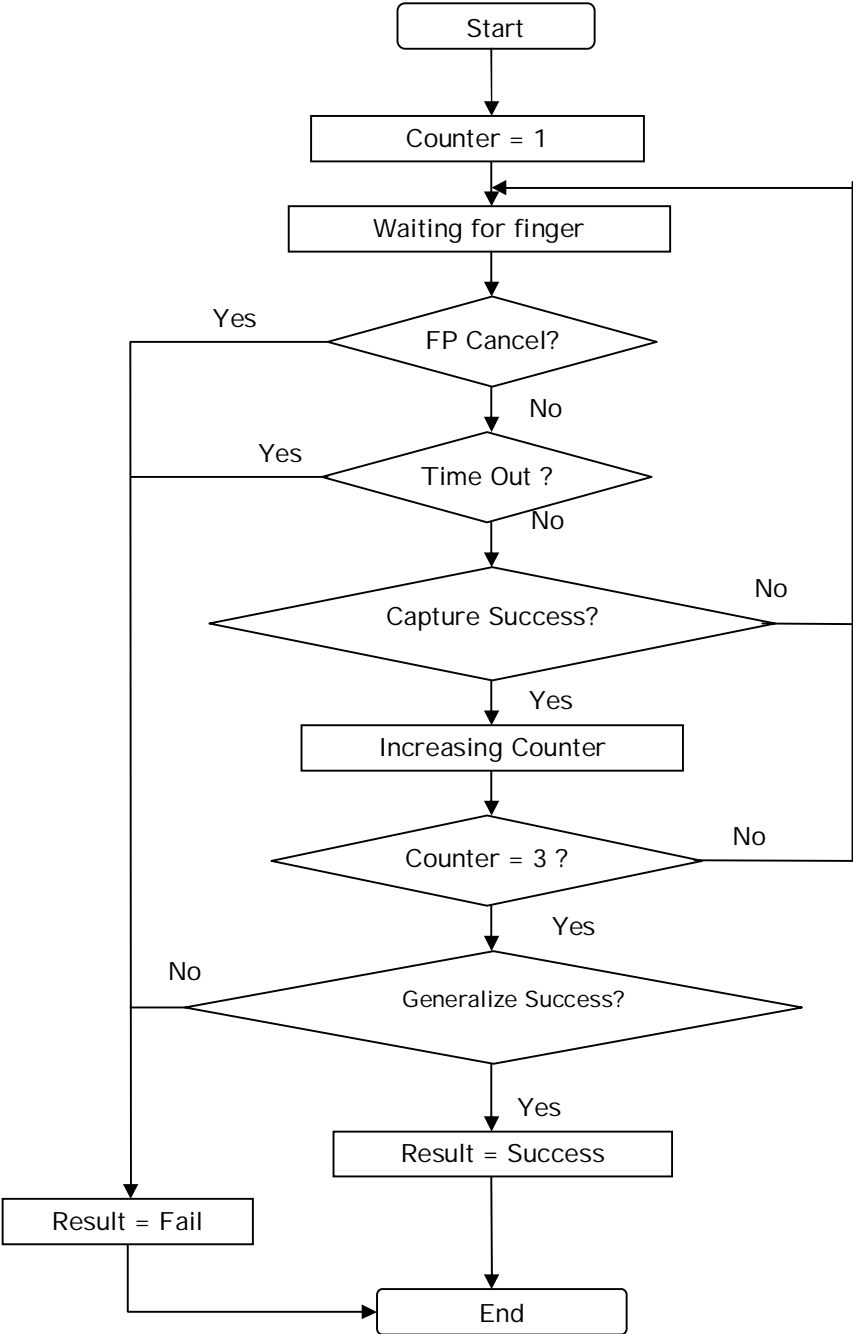
6.1 Acknowledge and Error Code List

No	ACK and Error Code	Value	Description
1	ERR_SUCCESS	0x00	Success for instruction execute
2	ERR_FAIL	0x01	Fail for instruction execute
3	ERR_VERIFY	0x11	1 to 1 match fail
4	ERR_IDENTIFY	0x12	Identify fail, no matched template
5	ERR_TMPL_EMPTY	0x13	Without the Template for the appointed ID.
6	ERR_TMPL_NOT_EMPTY	0x14	Existed template for the appointed ID.
7	ERR_ALL_TMPL_EMPTY	0x15	DataBase is null, Without enrolled Template.
8	ERR_EMPTY_ID_NOEXIST	0x16	Without available Template ID.
9	ERR_BROKEN_ID_NOEXIST	0x17	Without damaged Template.
10	ERR_INVALID_TMPL_DATA	0x18	Appointed Template Data invalid.
11	ERR_DUPLICATION_ID	0x19	The fingerprint has been enrolled.
12	ERR_BAD_QUALITY	0x21	Bad quality fingerprint image.
13	ERR_TIME_OUT	0x23	During Time Out period, not get any fingerprint input.
14	ERR_NOT_AUTHORIZED	0x24	Not authorized by the password If set password, and not use Verify Device Password command, then all commands would return error code expect Test Connection, Verify Device Password; If no password, all commands are available without the password.
15	ERR_GENERALIZE	0x30	Generalize template data fail
16	ERR_FP_CANCEL	0x41	Instruction cancelled
17	ERR_INTERNAL	0x50	Internal error
18	ERR_MEMORY	0x51	Memory Error
19	ERR_EXCEPTION	0x52	Firmware error.
20	ERR_INVALID_TMPL_NO	0x60	Template ID invalid
21	ERR_INVALID_SEC_VAL	0x61	The Value of Security Level is invalid
22	ERR_INVALID_TIME_OUT	0x62	The Value of Time Out is invalid
23	ERR_INVALID_BAUDRATE	0x63	The Value of Baudrate is invalid
24	ERR_DEVICE_ID_EMPTY	0x64	Not setting Device ID
25	ERR_INVALID_DUP_VAL	0x65	The Value of Duplication Check Option is invalid
26	ERR_INVALID_PARAM	0x70	Invalid parameter.
27	ERR_NO_RELEASE	0x71	Failed finger doesn't move away for Identify Free command
28	GD_DOWNLOAD_SUCCESS	0xA1	Download Template data was successful
29	GD_NEED_FIRST_SWEEP	0xFFF1	Waiting input fingerprint for first time.
30	GD_NEED_SECOND_SWEEP	0xFFF2	Waiting input fingerprint for second time.
31	GD_NEED_THIRD_SWEEP	0xFFF3	Waiting input fingerprint for third time.

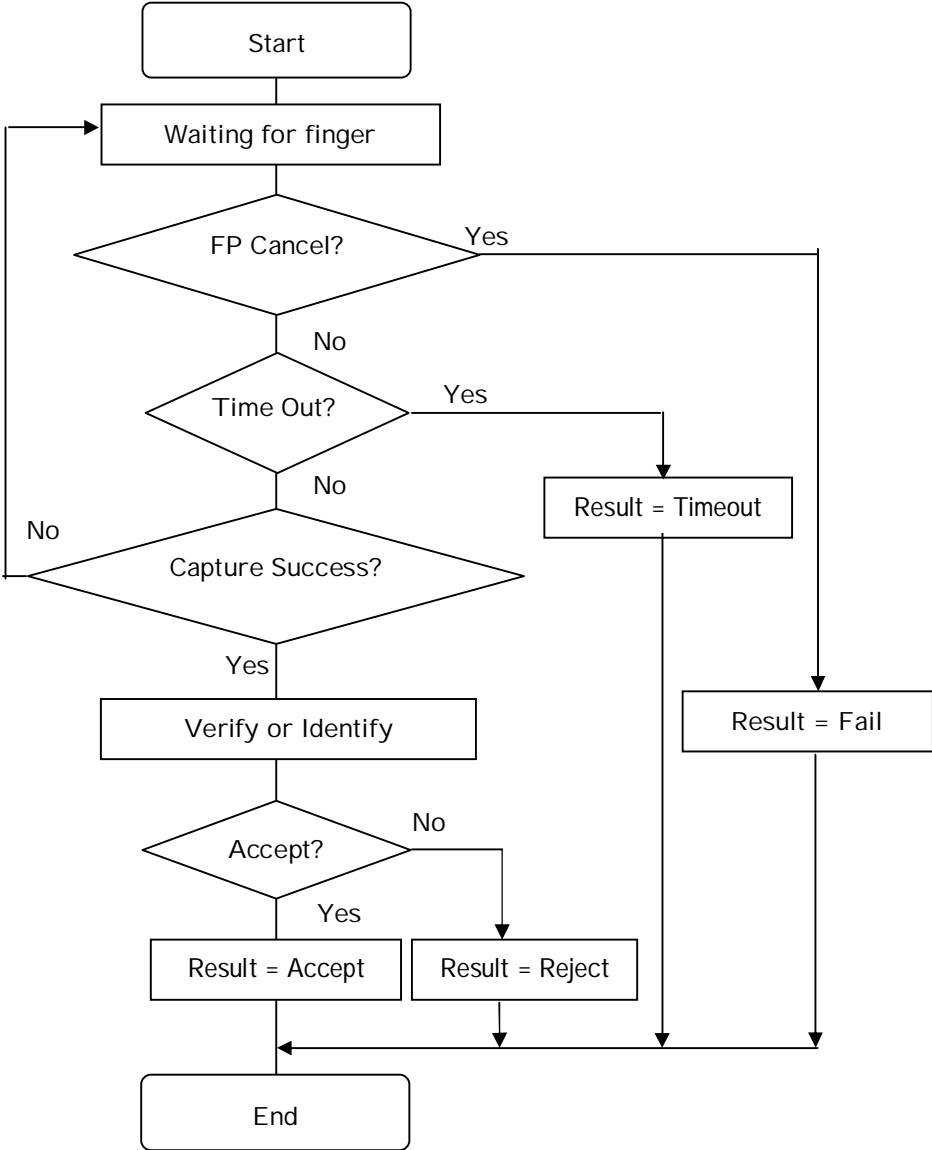
32	GD_NEED_RELEASE_FINGER	0xFFF4	Leave finger
33	GD_DETECT_FINGER	0x01	Detect an finger on sensor when execute "Finger Detect" command
34	GD_NO_DETECT_FINGER	0x00	Detect no any finger on sensor when execute "Finger Detect" command
35	GD_TEMPLATE_NOT_EMPTY	0x01	Template are not empty
36	GD_TEMPLATE_EMPTY	0x00	Template have been emptied

6.2 The Flow of Enroll and Identify

6.2.1 The Flow of Enroll



6.2.2 The Flow of Verify & Identify



Contact Information: