CAMA-SM25 Series

Fingerprint Identification Module

User's Manual

Shenzhen CAMA Biometrics Co., Ltd.

WWW.CAMABIO.COM

Table of Contents

1 Summary

CAMA-SM25 is the fingerprint module for secondary development which has integrated fingerprint optical sensor, CMOS image sensor, CPU, and Flash memory together. It features small size, low power consumption, simple ports, high reliability, small fingerprint template (496bytes), large fingerprint capacity, etc. It is convenient to be embedded to user system for realizing clients required fingerprint verification products.

CAMA-SM25 outstandingly features Auto-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.

CAMA-SM25 module is UART communication interface with adjustable safety level function, fingerprint data reading & writing function, 1:N and 1:1 verification function.

It has adapted the most advanced technology of Auto wake-up function. When the finger touching the sensor, system detected and will automatically power on as well as scanning & identify the fingerprint, which is much more convenient and intelligent instead of the button power-on design.

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
- Auto-learning 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
- Automatic Touch sense function
- 360 degree verification

1.2 Applications

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

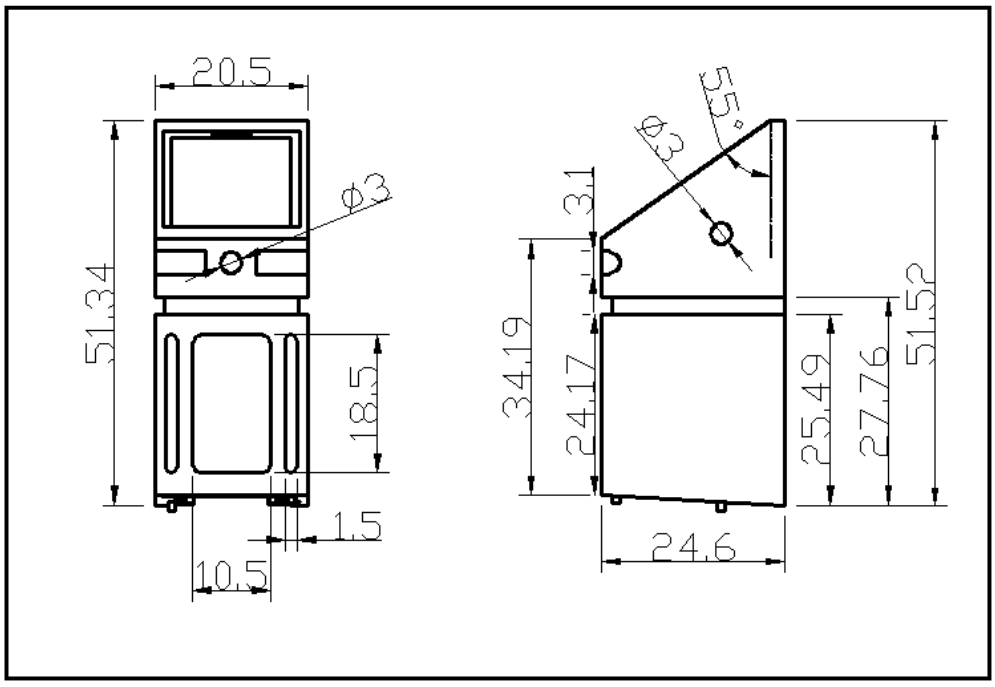
2 Module Hardware Description

2.1 Hardware Features

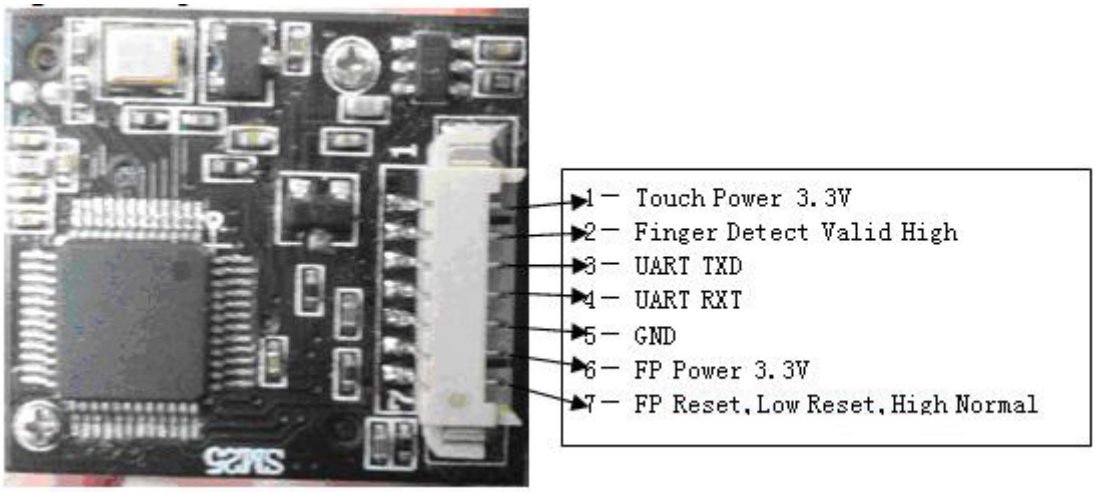
No.	Item	Description
1	Model	CAMA-SM25
2	CPU	120MHz, 1M flash memory,128SRAM
3	Sensor	CAMA Optical Sensor Scan Area 18 x 22 mm
4	Fingerprint image	242x 266(pixel)
5	Resolution	500 dpi
6	Power Supply	DC 3.3V±10%
7	Work current	<150mA
8	Communication port	UART

2.2 Configuration and Connector

2.2.1 Configuration



2.2.2 Connector Signal Description



Pin No.	Signal name	Signal Description
PIN 1	Touch Power 3.3V	Touch sense function need to supply power separately (It is better the power can be controlled). Power: DC3.3V± 0.1V Current:1.5uA to 2.5uA
PIN 2	Finger Detect	output high level when detected finger, and output low level once finger left
PIN 3	UART-TX	Module Transmit Output,3.3V TTL Logic
PIN 4	UART-RX	Module Receive Input,3.3V TTL Logic
PIN 5	GND	GND
PIN 6	VIN	DC Input, Range of Voltage is 3.3V±10%
PIN 7	FP Reset	Reset at low level (reset time>100ms), Work at high level. After Reset, it will cost about 350-400ms for system initializing and programs loading; Only after that the module can communicate with MCU by UART

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

3 Module Technical Parameters

3.1 Parameters

Item	Description
CPU	CPU: 120MHz, 1M flash memory,128SRAM
Sensor	CAMA Optical Sensor Scan： SM25:18 x 22 mm
Fingerprint Capacity	3000
FAR	< 0.001 %（Security Level 3）
FRR	< 0.1 %（Security Level 3）
Matching Method	1:N Identification and 1:1 Verification
Fingerprint Template	496 Bytes
Security Level	Level 1~ Level 5 can be set, Default setting：Level 3
Speed	Fingerprint pretreatment < 0.45 s
	1:N（2000 fingerprints）< 0.9s
Resistant to Abrasion	>1 Million Finger Press
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
Working Voltage	DC 3.3V±10%
Working Current	<150mA
Working Environment	Working Temperature: -20 ℃- 60 ℃ Working Humidity: 20%– 80%
Storage Environment	Storage Temperature: -20℃ ~80℃ Storage Humidity: <90%

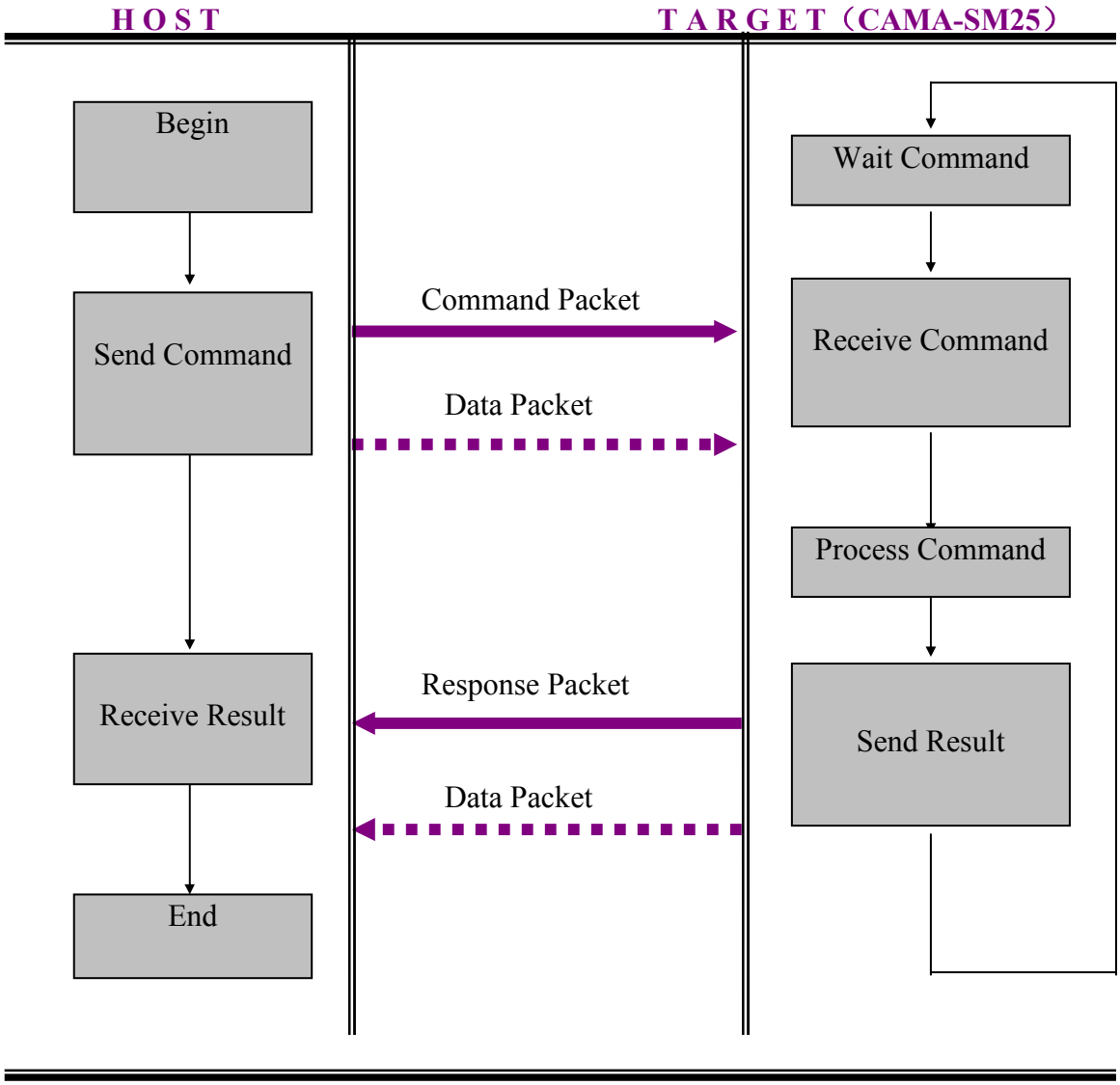
3.2 Default Factory Settings

Item	Initial Value
Security Level	3
Finger Print Time Out	0 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)
Default Setting is: 115200bps, 1 start-bit, 1 stop-bit, no check bit

4.1 The process of communication



Notice. Host do not send next command until receive result, except send FP Cancel

4.2 Classify of communication packet

4.2.1 Command packet

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

4.2.2 Response packet

Response packet is result of execute command packet, from Target (CAMA-SM Series) 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	DESCRYPTION
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	DESCRYPTION
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	
0x5A	0xA5	L	H	L	H	D0	D1	…	Dn-1	L	H
0	1	2	3	4	5	6	7	…	6+n-1	6+n	6+n+1

OFFSET	FIELD	TYPE	SIZE	DESCRYPTION
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	0x5A	L	H	L	H	L	H	D0	D1	…	Dn-3	L	H
0	1	2	3	4	5	6	7	8	9	…	6+n-1	6+n	6+n+1

OFFSET	FIELD	TYPE	SIZE	DESCRYPTION
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

Template Data(496Bytes)+CheckSum(2Bytes)=498Bytes

Template Data	CheckSum
496 bytes Template Data D0,D1,…D495	2 byte Check Sum 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	-	0x0102	4	0/1	Template No /Error Code
3	Enroll	0x0103	2	Template No.	0x0103	4/6	0/1	Template No /Error Code + Duplicaion ID
4	Enroll One Time	0x0104	2	Template No.	0x0104	4/6	0/1	Template No /ErrorCode
5	Clear Template	0x0105	2	Template No.	0x0105	4	0/1	Template No /ErrorCode
6	Clear All Template	0x0106	0	-	0x0106	4	0/1	Cleared Template Count /Error Code
7	Get Empty ID	0x0107	0	-	0x0107	4	0/1	Template No /Error Code
8	Get Template Status	0x0108	2	Template No.	0x0108	4	0/1	Template Status /Error Code
9	Get Broken Template	0x0109	0	-	0x0109	6	0/1	Broken Template Count + First Broken Template No /Error Code
10	Read Template	0x010A	2	Template No.	0x010A	(Template Record Size + 4) or (4)	0/1	Template No + Template Record Data /Error Code
11	Write Template	0x010B	Template Record Size + 2	Template No + Template Record Data	0x010B	4	0/1	Template No /Error Code
12	Set Security Level	0x010C	2	Security Level Value	0x010C	4	0/1	Security Level Value /Error Code
13	Get Security Level	0x010D	0	-	0x010D	4	0	Security Level Value
14	Set Finger Time Out	0x010E	2	Time Out Value	0x010E	4	0/1	TimeOut Value /Error Code
15	Get Finger Time Out	0x010F	0	-	0x010F	4	0	TimeOut Value
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

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

5.3 Command Packet Description

5.3.1 Verify

[Function] One to one match

[Operation Sequence]

1) if the template No. is invalid, result= ERR_INVALID_TMPL_NO;
2) if the template No. there is no template data, result= ERR_TMPL_EMPTY;
3) detect finger whether press on sensor or not repeatedly,

if no any finger press on sensor in the period of timeout, result= ERR_TIME_OUT;

4) else check quality of the captured fingerprint image

if image is not good, result= ERR_BAD_QUALITY

5) else result=GD_NEED_RELEASE_FINGER, denote that lift finger ;
6) verify the captured fingerprint with the appoint template data

if verify ok ,result=Template No.; else result= ERR_VERIFY;

7) In the period of process verify command, if module received “FP Cancel”command,

The module then stop verify command and return ACK that is ERR_FP_CANCEL;

Please refer to appendix 6.2.2 about the flow of Verify & Identify

[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: GD_NEED_RELEASE_FINGER or Template No. Fail: Error Code is as follow ERR_VERIFY ERR_INVALID_TMPL_NO ERR_TMPL_EMPTY ERR_TIME_OUT ERR_BAD_QUALITY
CKS		Check Sum

[Example of Verify Command]

Host send Verify ,Template No.=1,command packet as follow

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

If finger is on sensor and captured image is good,

module then return "GD_NEED_RELEASE_FINGER"

Module return the result of verify response packet as follow

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

5.3.2 Identify

[Function] One to many match

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

[Operation Sequence]

1) if Template Data Base is null, result= ERR_ALL_TMPL_EMPTY;
2) detect finger whether press on sensor or not repeatedly;

if no any finger press on sensor in the period of timeout, result= ERR_TIME_OUT;

3) else check quality of the captured fingerprint image,

if image is not good, result= ERR_BAD_QUALITY

4) else result=GD_NEED_RELEASE_FINGER, denote that lift finger
5) identify the captured fingerprint with the all template data in FLASH memory,

if identify ok ,result=matched template No.; else result= ERR_IDENTIFY

6) In the period of process identify command, if module received "FP Cancel "command, The module then stop identify command and return ACK that is ERR_FP_CANCEL

Please refer to appendix about the flow of Verify & 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: GD_NEED_RELEASE_FINGER or Template No. Fail: Error Code is as follow ERR_IDENTIFY ERR_ALL_TMPL_EMPTY ERR_TIME_OUT ERR_BAD_QUALITY
CKS		Check Sum

[Example of Identify Command]

① Host send identify command,command packet as follow

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

② if finger is on sensor and captured image is good,
module return "GD_NEED_RELEASE_FINGER"

③ Module return the result of identify

②,③response packet as follow

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

5.3.3 Enroll

[Function] Enroll

In the process of enroll, The same finger must be press on the sensor 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.

[Operation Sequence]

1) if the appoint template No. is invalid,result= ERR_INVALID_TMPL_NO;
2) if the template No. have existed template data,result=ERR_TMPL_NOT_EMPTY;
3) else result =GD_NEED_FIRST_SWEEP,denote that press finger for the first time;
4) detect finger whether press on sensor or not repeatedly,
if no finger press on sensor in the period of timeout,result=ERR_TIME_OUT;
5) else check quality of captured finger image,
if image is no good,result=ERR_BAD_QUALITY
6) else result=GD_NEED_RELEASE_FINGER denote that lift finger,
after finger release then goto next steep;
7) result=GD_NEED_SECOND_SWEEP denote that press the same finger for the second time,
and then goto steep4,5
8) if the second input fingerprint is correct,
result=GD_NEED_RELEASE_FINGER denote that lift finger,
after finger release then goto next steep;
9) result=GD_NEED_THIRD_SWEEP denote that press the same finger for the third time
and then goto steep4,5
10) if the third input fingerprint is correct,
result=GD_NEED_RELEASE_FINGER denote that lift finger;
11) the module generalize the three templates to one template,
success goto steep12, fail result=ERR_GENERALIZE;
12) if Duplication Check=OFF, storage the template data and return result=Template No;

13) if Duplication Check=ON,the template data match to all template to check whether exist duplicated fingerprint or not. if yes, result = ERR_DUPLICATION_ID;
else result = Template No. and sorage the template data;

14) In the period of process ENROLL command,if module received "FP Cancel "command, the module then stop ENROLL command and return ACK that is ERR_FP_CANCEL;

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		4/6
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 or Template No. Fail: Error Code is as follow ERR_INVALID_TMPL_NO ERR_TMPL_NOT_EMPTY ERR_TIME_OUT ERR_BAD_QUALITY ERR_GENERALIZE
	2byte	Null or ERR_DUPLICATION_ID
CKS		Check Sum

[Example of Enroll Command]

Enroll to Template No.=1, Command Packet

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

Response Packet

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

5.3.4 Enroll One Time

[Function] Enroll One Time

Enroll command require the same finger press on the sensor for 3 times,Corresponding Enroll One Time command require the finger press on sensor for one time only

[Operation Sequence]

1) if the appoint template No. is invalid,result= ERR_INVALID_TMPL_NO;
2) if the template No. have existed template data,result=ERR_TMPL_NOT_EMPTY;
3) detect finger whether press on sensor or not repeatedly,
if no finger press on sensor in the period of timeout,result=ERR_TIME_OUT;
4) else check quality of captured finger image,
if image is no good,result=ERR_BAD_QUALITY;
5) else result=GD_NEED_RELEASE_FINGER denote that lift finger;
6) if Duplication Check=OFF, storage the template data and return result=Template No;
7) if Duplication Check=ON,the template data match to all template to check whether exist duplicated fingerprint or not .Yes, result = ERR_DUPLICATION_ID;
else result = Template No. and storage the template data

8) In the period of process Enroll One Time command,if module received "FP Cancel ",

the module then stop the command and return ACK that is ERR_FP_CANCEL;

Notice:
we recommend using Enroll command,disapproval using Enroll one time command

[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		4/6
RET		ERR_SUCCESS or ERR_FAIL
DATA	2byte	Success: GD_NEED_RELEASE_FINGER / Template No. Fail: Error Code is as follow ERR_INVALID_TMPL_NO ERR_TMPL_NOT_EMPTY ERR_TIME_OUT ERR_BAD_QUALITY
	2byte	Null or ERR_DUPLICATION_ID
CKS		Check Sum

[Example of Enroll One Time]

Enroll to Template No.=1,Enroll One Time Command packet is as follow

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

Response Packet is as follow

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

5.3.5 Clear Template

[Function]

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

[Operation Sequence]

1 if the appoint template No. is invalid,result= ERR_INVALID_TMPL_NO
2 if the appoint template No. is inexistence template data,result=ERR_TMPL_EMPTY
3 else delete specified template data and then return response packet

[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. to be deleted Fail: Error Code is as follow ERR_INVALID_TMPL_NO ERR_TMPL_EMPTY
CKS	Check Sum

[Example of Clear Template]

Clear Template No.=1 command packet

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

Response packet

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	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 deleted template Fail:Error code
CKS	Check Sum

[Example of Clear All Template]

Host send Clear All Template Command Packet

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

Response Packet

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	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

[Operation Sequence]

Search the first number that can be used for storage fingerprint template,
if all fingerprint data base is full,result= ERR_EMPTY_ID_NOEXIST;
else result=template No. that can be utilized

[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: template No. that can be utilized Fail: ERR_EMPTY_ID_NOEXIST
CKS	Check Sum

[Example of Get Empty ID]

Command packet:

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

Response Packet:

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	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 utilized or not.

[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: ERR_INVALID_TMPL_NO
CKS		Check Sum

[Example of Get Template Status]

Get template status that No.=1,command packet

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

Response Packet

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

5.3.9 Get Broken Template

[Function]

Check fingerprint template Data base whether is damage or not.some unit of FLASH memory may be damaged by chance failure

For the broken template data, you can delete the template and then enroll again.

[Operation Sequence]

If exist broken template data,result=total number of broken template + template No.

else total number=0,Template No.=0

[Command and Response]

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

[Example of Get Broken Template]

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

Response Packet

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	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

[Operation Sequence]

1) if the appoint template No. is invalid,result= ERR_INVALID_TMPL_NO
2) if the appoint template No. is inexistence template data,result=ERR_TMPL_EMPTY
3) else result= (Template Record Size + 2), denote that length of the data field of Response data packet
4) utilize Response data packet to transmit the template data

[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: （Template Record Size + 2） Fail:Error Code
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

[Example of Read Template]

Read Template Command Packet

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

Response and Response Data Packet

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x0A	0x01	0x04	0x00	0x00	0x00	0xF4	0x01	…	0x00	0x03	0x02
0xA5	0x5A	0x0A	0x01	0xF6	0x01	0x00	0x00	0x01	0x00	498 Bytes Template Data		CKS	

5.3.11 Write Template

[Function]

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

[Operation Sequence]

Host send Write Template command to cause module wait for receive Command Data Packet to get template data from Host

1) Module check the Command packet whether is correct or not
   if size of Template record is incorrectness,result=ERR_INVALID_PARAM
   else return response packet and prepare receive template data (Command Data Packet)
2) Host send Command Data Packet with template No. and Template Data
3) after receive Command Data Packet is successful,
   if the appoint template No. is invalid,result= ERR_INVALID_TMPL_NO
   if CheckSum of Template Data is incorrectness,result=ERR_INVALID_PARAM
4) else Template Data that received write to Flash Memory

[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：ERR_INVALID_PARAM
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:Error Code
CKS	Check Sum

[Example of Write Template]

1) Host send Write Template command packet

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

2) Response to the write template command

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

3) Host send Command Data Packet with Template No. and Template Data

PREFIX		CMD		LEN		DATA			CKS
0x5A	0xA5	0x0B	0x01	0xF4	0x01	0x01	0x00	498Bytes Template Data	CKS

4) After check the command data packet is correct,Module write the template data to Flash memory ,and then response the command data packet

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 the threshold of fingerprint identification engine, Integer of 1-5 can be selected, one is the lowest identification level and five is the highest identification level, Default is three

[Operation Sequence]

1) if value of the security level is invalid,result=ERR_INVALID_SEC_VAL

2) else update the value of Security Level,then response the command

[Security Level and Fingerprint Identification precision]

Security Level	Value of FAR & FRR	
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 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:the Value of Security Level Fail: ERR_INVALID_SEC_VAL
CKS	Check Sum

[Example of Set Security Level]

Set Security Level command packet

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

Set Security Level command response packet

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

5.3.13 Get Security Level

[Function]

Host read the value of Security Level from 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	Value of Security Level
CKS	Check Sum

[Example of Get Security Level]

Get Security Level command packet

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

Get Security Level command response packet

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

5.3.14 Set Finger Time Out

[Function]

Verify,Identify,Enroll,Enroll One Time,Enroll And Store in RAM,
Get Feature Data of Captured FP,Verify Downloaded Feature with Captured FP,
Identify Downloaded Feature with Captured FP,Identify Free

In the period of above command executing,the parameter of the Finger TimeOut is the time limit of detect finger on sensor repeatedly

0-60s can be selected, Default is 5s

[Operation Sequence]

1) if the value of TimeOut is out off range,result=ERR_INVALID_TIME_OUT
2) else update the value of TimeOut and response the command

[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	Value of Time Out or ERR_INVALID_TIME_OUT
CKS	Check Sum

[Example of Set Finger TimeOut]

Set Finger TimeOut command packet

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

Command response packet

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

[Example of Get Finger TimeOut]

Command packet

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

Response packet

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	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

[Operation Sequence]

1) if the value is out off range,result=ERR_INVALID_PARAM
2) else update the value of Device ID and response the command

[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	Device ID or ERR_INVALID_PARAM
CKS	Check Sum

[Example of Set Device ID]

Command Packet

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

Response Packet

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	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	Device ID or Error Code
CKS	Check Sum

[Example of Get Device ID]

Command Packet

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

Response packet

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

5.3.18 Get F/W Version

[Function] Get Firmware Version of the Module

[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
DATA	D0	Version Major
	D1	Version Minor

[Example]

Command Packet

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

Response packet

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x12	0x01	0x04	0x00	0x00	0x00	0x02	0x09	…	0x00	0x0D	0x01

5.3.19 Finger Detect

[Function] Detect whether finger press on the sensor or not

[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	Result of detect: 1：Finger on sensor, 0：no finger on sensor
CKS	Check Sum

Notice:
Host need send Sensor LED Control enable command to power on LED before Finger Detect command, otherwise the result of Finger Detect is incorrect

[Example]

Detect Finger command packet

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

Response Packet of Finger detect

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

5.3.20 Set Baudrate

[Function] Set UART Baudrate

[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

Notice: New value is active only by reset or power on

[Example]

Set Baudrate=9600BPS

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

Response packet

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

5.3.21 Set Duplication Check(ON/OFF)

[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: 1/0 (status of duplication check) Fail:ERR_INVALID_DUP_VAL
CKS	Check Sum

[Example]

Set duplication check = Enable

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

Response Packet

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

5.3.22 Get Duplication Check

[Function] Get Status of Duplication Check

[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	Status of duplication check currently 1:Enable ; 0:Disable
CKS	Check Sum

[Command Example]

Get status of Duplication Check

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

Response Packet

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

5.3.23 Enter Standby Mode

[Function]

Put module into standby mode

Only reset or power on will lead module from standby mode to active mode

[Sequence]

1) Module enter into Standby Mode and result=ERR_SUCCESS。

2) From standby mode to active mode only by RESET or Power ON

[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: Before power off module, Enter Standby Mode Command is recommand

[Example]

Enter Standby Mode

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

Response Packet

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	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, exclusive difference is "Enroll" command store template data into FLASH memory,but "Enroll And Store in RAM" command store templatedata into RAM temporarily,Host can utilize "Get Enroll Data" command acquire template data.

[Sequence]

1) else result =GD_NEED_FIRST_SWEEP,denote that press finger for the first time;
2) detect finger whether press on sensor or not repeatedly,
if no finger press on sensor in the period of timeout,result=ERR_TIME_OUT;
3) else check quality of captured finger image,
if image is no good,result=ERR_BAD_QUALITY
4) else result=GD_NEED_RELEASE_FINGER denote that lift finger,
after finger release then goto next steep;
5) result=GD_NEED_SECOND_SWEEP denote that press the same finger for the second time,
and then goto steep2,3
6) if the second input fingerprint is correct,
result=GD_NEED_RELEASE_FINGER denote that lift finger,
after finger release then goto next steep;
7) result=GD_NEED_THIRD_SWEEP denote that press the same finger for the third time
and then goto steep2,3
8) if the third input fingerprint is correct,
result=GD_NEED_RELEASE_FINGER denote that lift finger;
9) the module generalize the three templates to one template,
success goto steep ⑩, fail result=ERR_GENERALIZE;
10) result=0 and sorage the template data into RAM ;
11) In the period of process the command,if module received "FP Cancel "command,
the module then stop this command and return ACK that is ERR_FP_CANCEL;

Please refer to appendix about the flow of Enroll

[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	Success: GD_NEED_FIRST_SWEEP/GD_NEED_SECOND_SWEEP/ GD_NEED_THIRD_SWEEP/GD_NEED_RELEASE_FINGER/0 Fail: ERR_TIME_OUT or ERR_BAD_QUALITY or ERR_GENERALIZE
CKS	Check Sum

[Example]

Enroll And Store in RAM command packet

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

Response Packet

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

5.3.25 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 indicate the length of data field in Response Data Packe Fail:Error Code
CKS	Check Sum
Response Data Packet	
PREFIX	0x5AA5
RCM	0x0119
LEN	Template Record Size + 2
RET	ERR_SUCCESS
DATA	Template Record Data
CKS	Check Sum

[Example of Get Enroll Data]

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

Response Packet and Response Data Packet

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x19	0x01	0x04	0x00	0x00	0x00	0xF2	0x01	…	0x00	0x10	0x02
0xA5	0x5A	0x19	0x01	0xF4	0x01	0x00	0x00	498Bytes Template Data				CKS	

5.3.26 Get Feature Data of Captured FP

[Function] Get template data when capture an fingerprint image

This command is equivalent "Enroll One Time" + "Get Enroll Data" command

[Sequence]

After module have received the "Get Feature Data of Captured FP" command,
Module detect finger continually
1) if no any finger on sensor in the period detect finger,result=ERR_TIME_OUT;
2) check the quality of captured fingerprint image ,
if the image is not good,result=ERR_BAD_QUALITY;
3) else acquire the template data and upload to host utilize Response data packet
4) In the period of process the command,if module received "FP Cancel "command,
the module then stop this command and return ACK that is ERR_FP_CANCEL;

[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 Fail:ERR_TIME_OUT or ERR_BAD_QUALITY/
CKS	Check Sum
Response Data Packet	
PREFIX	0x5AA5
RCM	0x011A
LEN	Template Record Size + 2
RET	ERR_SUCCESS
DATA	Template Record Data
CKS	Check Sum

[Example]

Get Feature Data of Capture FP

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

Response packet and Response Data Packet

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x1A	0x01	0x04	0x00	0x00	0x00	0xF2	0x01	…	0x00	0x11	0x02
0xA5	0x5A	0x1A	0x01	0xF4	0x01	0x00	0x00	498Bytes Template Data				CKS	

5.3.27 Verify Downloaded Feature with Captured FP

[Function] Verify downloaded template data with live captured fingerprint

This command is equivalent "Write Template Data" and "Verify" command

[Sequence]

1) Host send this command packet cause module wait for receive Command Data Packet to get template data from Host.
host set length of template data in the data field of command packet
2) Module return response packet to inform host that module have get ready for receive template data packet.
3) Host send command data pakect with template data to module
4) After module received the template data successful,module detect finger continually
5) if no any finger on sensor in the period detect finger,result=ERR_TIME_OUT;
6) check the quality of captured fingerprint image ,
if the image is not good,result=ERR_BAD_QUALITY.
7) else verify and then response the result of verify
8) In the period of process the command,if module received "FP Cancel "command,
the module then stop this command and return ACK that is ERR_FP_CANCEL

[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

[Example]

1) Host send verify downloaded feature data with captured FP command packet

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

2) Module return response packet to host

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

3) Host download template data to module

PREFIX		CMD		LEN		DATA	CKS
0x5A	0xA5	0x1B	0x01	0xF2	0x01	498Bytes Template Data	CKS

4) Module detect finger then verify and response the result

PREFIX		RCM		LEN		RET		DATA		CKS	
0xA5	0x5A	0x1B	0x01	0x04	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
This command is equivalent "Write Template Data" 2 times and "Identify" command
[Sequence]

1) Host send this command packet cause module wait for receive Command Data Packet to get template data from Host.
host set length of template data in the data field of command packet
2) Module return response packet to inform host that module have get ready for receive the first template data packet.
3) Host send command data pakect with the first template data to module
4) Module return response packet with result= GD_DOWNLOAD_SUCCESS,inform host waitting for the next template data.
5) Host send command data pakect with the second template data to module
6) After module received the 2 templates data successful,module detect finger continually
7) if no any finger on sensor in the period detect finger,result=ERR_TIME_OUT;
8) check the quality of captured fingerprint image ,
if the image is not good,result=ERR_BAD_QUALITY.
9) else identify the 2 template's data with captured fingprint and then response result of identify
10) In the period of process the command,if module received "FP Cancel "command,
the module then stop this command and return ACK that is ERR_FP_CANCEL

[Command and Response]

Command packet	
PREFIX	0xAA55
CMD	0x011C
LEN	2
DATA	Template Record Size
PREFIX	0x55AA
RCM	0x011C
LEN	4
RET	ERR_SUCCESS or ERR_FAIL
DATA	Success:0 ; Fail：Error Code
Command Data packet	
PREFIX	0xA55A
CMD	0x011C
LEN	Template Record Data + 2
DATA	Template Index(2Byte) + Template Record Data
PREFIX	0x5AA5
RCM	0x011C
LEN	4
RET	ERR_SUCCESS or ERR_FAIL
DATA	Success: GD_DOWNLOAD_SUCCESS; Fail：Error Code
PREFIX	0xA55A
CMD	0x011C
LEN	Template Record Data + 2
DATA	Template Index(2Byte) + Template Record Data
PREFIX	0x5AA5
RCM	0x011C
LEN	4
RET	ERR_SUCCESS or ERR_FAIL
DATA	Success：0 ; Fail：Error Code

[Example]

1) Host send Identify Downloaded Feature With Captured FP command packet

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

2) Module return response packet to host

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

3) Host send the first template data to module

PREFIX		CMD		LEN		DATA			CKS
0x5A	0xA5	0x1C	0x01	0xF4	0x01	0x01	0x00	498Byte Template Data	CKS

4) Module request the next template data

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

5) Host send the second template data to module

PREFIX		CMD		LEN		DATA			CKS
0x5A	0xA5	0x1C	0x01	0xF4	0x01	0x02	0x00	498Bytes Template Data	CKS

6) Module detect finger then identify and response the result

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 module

[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

[Example]

Get Device Name command packet

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

Response Packet

PREFIX		RCM		LEN		RET		DATA	CKS
0xAA	0x55	0x21	0x01	0x10	0x00	0x00	0x00	Device Name in ASCII	CKS

5.3.30 Sensor LED Control

[Function] ON/OFF LED of fingerprint sensor

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x0124
LEN	2
DATA	Sensor LED status（1:ON ， 0:OFF）
CKS	Check Sum
Response Packet	
PREFIX	0x55AA
RCM	0x0124
LEN	4
RET	Success：ERR_SUCCESS
DATA	0
CKS	Check Sum

[Example]

Sensor LED ON command packet

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

Response Packet

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

5.3.31 Identify Free

[Function]

The command cause module detect finger then identify and return result continually till received FPCancel command

→→ detect finger → identify → return identify result →

[Sequence]

1) if no any template data,result=ERR_ALL_TMPL_EMPTY
2) Detect finger continually
3) Check quality of fingerprint image,if image is no good result=ERR_BAD_QUALITY.
4) else image is good,result=GD_NEED_RELEASE_FINGER
5) identify and return the result
if identify OK result=ERR_SUCCESS + Template No. ;
else result=ERR_FAIL + ERR_IDENTIFY
6) goto step2 untill FPCancel command
7) if received FpCancel command,stop identify free command and return ERR_FP_CANCEL

[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 No. Fail： ERR_ALL_TMPL_EMPTY ERR_BAD_QUALITY ERR_IDENTIFY ERR_NO_RELEASE
CKS	Check Sum

[Example]

Host send Identify Free command packet

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

Module respone the result:GD_Need_Release_Finger and

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

5.3.32 Set Device Password

[Function]

Default setting is null,all command can be executed,Verify password is needless.

Password is composed by 14bytes ASCII code,less than 14byte will report error

Passowrd is authentication by Verify Device Password command

If new password is setup successful, all command is invalid unless "Verify Device Password" is successful. but "TestConnection" and "Verify Device Password" command is valid

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x0126
LEN	14
DATA	14 Bytes password (form by ASCII Code)
CKS	Check Sum
Response Packet	
PREFIX	0x55AA
RCM	0x0126
LEN	4
RET	Success：ERR_SUCCESS，Fail：ERR_FAIL
DATA	Success:0 Fail:Error Code
CKS	Check Sum

[Example]

Host send "Set Device Password" command packet

PREFIX		CMD		LEN		DATA	CKS
0x55	0xAA	0x26	0x01	0x0E	0x00	14Bytes ASCII Code	CKS

Module response the result of "Set Device Password"

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x26	0x01	0x04	0x00	0x00	0x00	0x00	0x00	…	0x00	0x2A	0x01

5.3.33 Verify Device Password

[Function] Verify password between host and module

Device Password is null by Default setting and all command is valide,verify device password command is needless

If device's new password is setup,ever time power on the device from power down,the module is safeguard by password,all command except is invalide (Result=ERR_NOT_AUTHORIZED) untill verify password is successful,but "Verify Device Password" and "Test Connection" command is valide in any time.

[Sequence]

1) if verify device password is OK,result= ERR_SUCCESS,all command is valide
2) else result= ERR_NOT_AUTHORIZED, module is safeguard by password

[Command and Response]

Command Packet	
PREFIX	0xAA55
CMD	0x0127
LEN	14
DATA	14 Bytes password (form by ASCII code)
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

[Example]

Host send Verify Device Password command packet

PREFIX		CMD		LEN		DATA	CKS
0x55	0xAA	0x27	0x01	0x0E	0x00	14Bytes passoword form by ASCII	CKS

Module response the result of verify device password

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x27	0x01	0x04	0x00	0x00	0x00	0x00	0x00	…	0x00	0x2B	0x01

5.3.34 Get Enroll Count

[Function]

Get total number of fingerprint template that have enrolled

[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	CheckSum

[Example]

Host send Get Enroll Count

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

Module response result of get enroll count

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

5.3.35 FP Cancel

[Function]

1) In the period of process following command:

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

Once received command of FP Cancel,stop the command in process immediately,

then return 2 response packet :

one is the result=ERR_FP_CANCEL indicate current command have been cancelled.

The other result=ERR_SUCCESS denote that operation of cancel is successful.

2) for the other command except above ①,only return one response packet that is

Result= ERR_SUCCESS denote that operation of cancel is successful

[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

[Example 1] Host send FP Cancel commonly

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

Module response the result of cancel

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

[Example 2]:

In process of executing "Enroll" command, host send FP Cancel command

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

Module response two result as following

Result= ERR_FP_CANCEL denote that module have stop Enroll command													
PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x03	0x01	0x04	0x00	0x01	0x00	0x41	0x00	…	0x00	0x49	0x01
Result= ERR_SUCCESS denote that operation of cancel is successful													
PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	0x55	0x30	0x01	0x04	0x00	0x00	0x00	0x00	0x00	…	0x00	0x34	0x01

5.3.36 Test Connection

[Function] Test connection between host and module

[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

[Example of Test Connection]

Host send Test Connection Command packet

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

Module response connection is successful

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	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

[Example]

Host send incorrect command

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

Response the incorrect command

PREFIX		RCM		LEN		RET		DATA				CKS	
0xAA	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

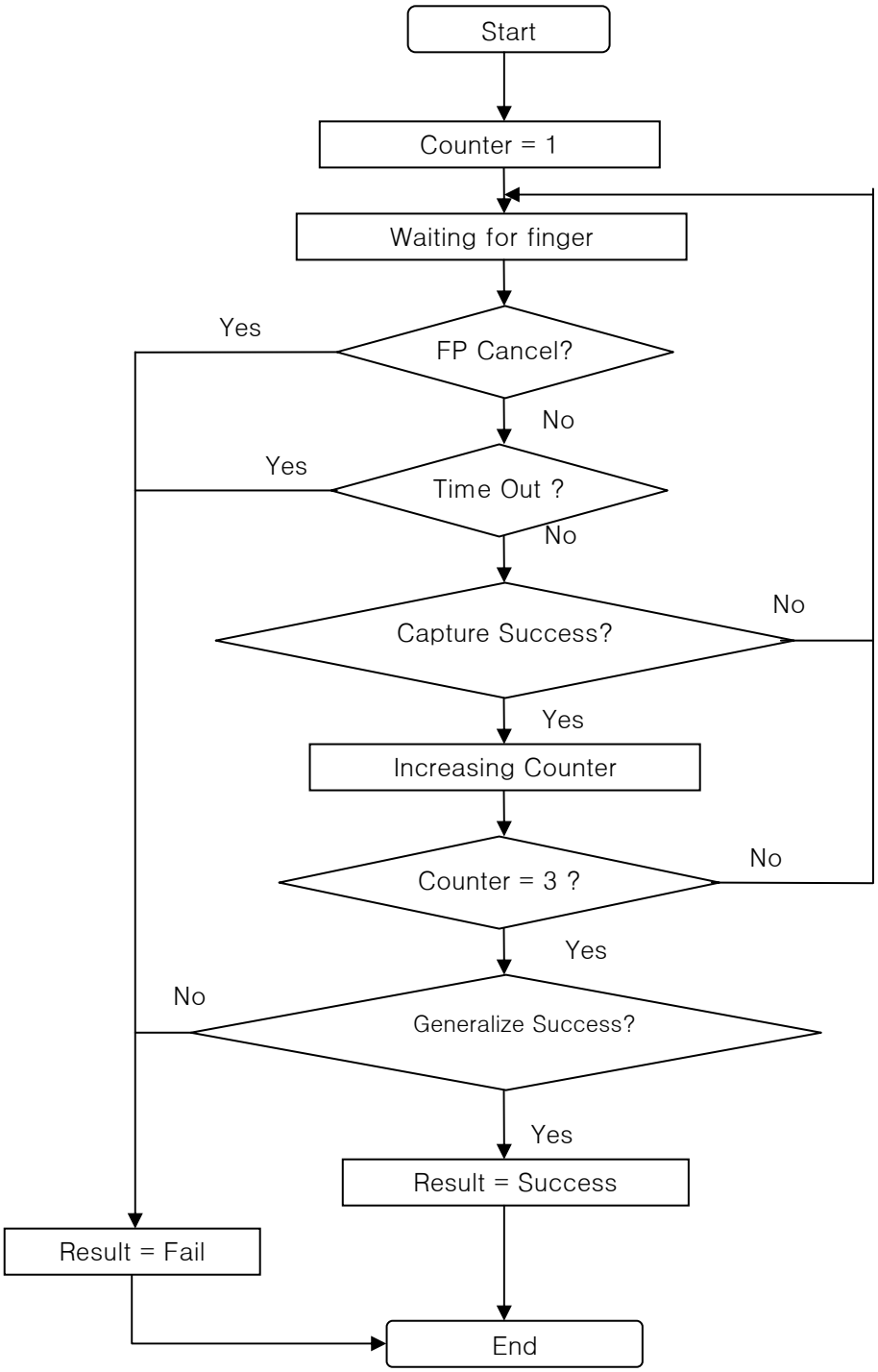
6.1 Acknowledge and Error Code List

No	ACK and Error Code	Value	Description
1	ERR_SUCCESS	0x00	Success for command execute
2	ERR_FAIL	0x01	Fail for instruction execute
3	ERR_VERIFY	0x11	One to one match fail
4	ERR_IDENTIFY	0x12	Identify fail, no matched template
5	ERR_TMPL_EMPTY	0x13	The appointed Template Data is Null
6	ERR_TMPL_NOT_EMPTY	0x14	Existed template for the appointed ID.
7	ERR_ALL_TMPL_EMPTY	0x15	All template 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, no finger is detected
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	Relative fingerprint command have been 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 No. is 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	Option Value of Duplication Check is invalid
26	ERR_INVALID_PARAM	0x70	Parameter is invalid.
27	ERR_NO_RELEASE	0x71	In process of IdentifyFree command, Finger is not release
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	Lift finger

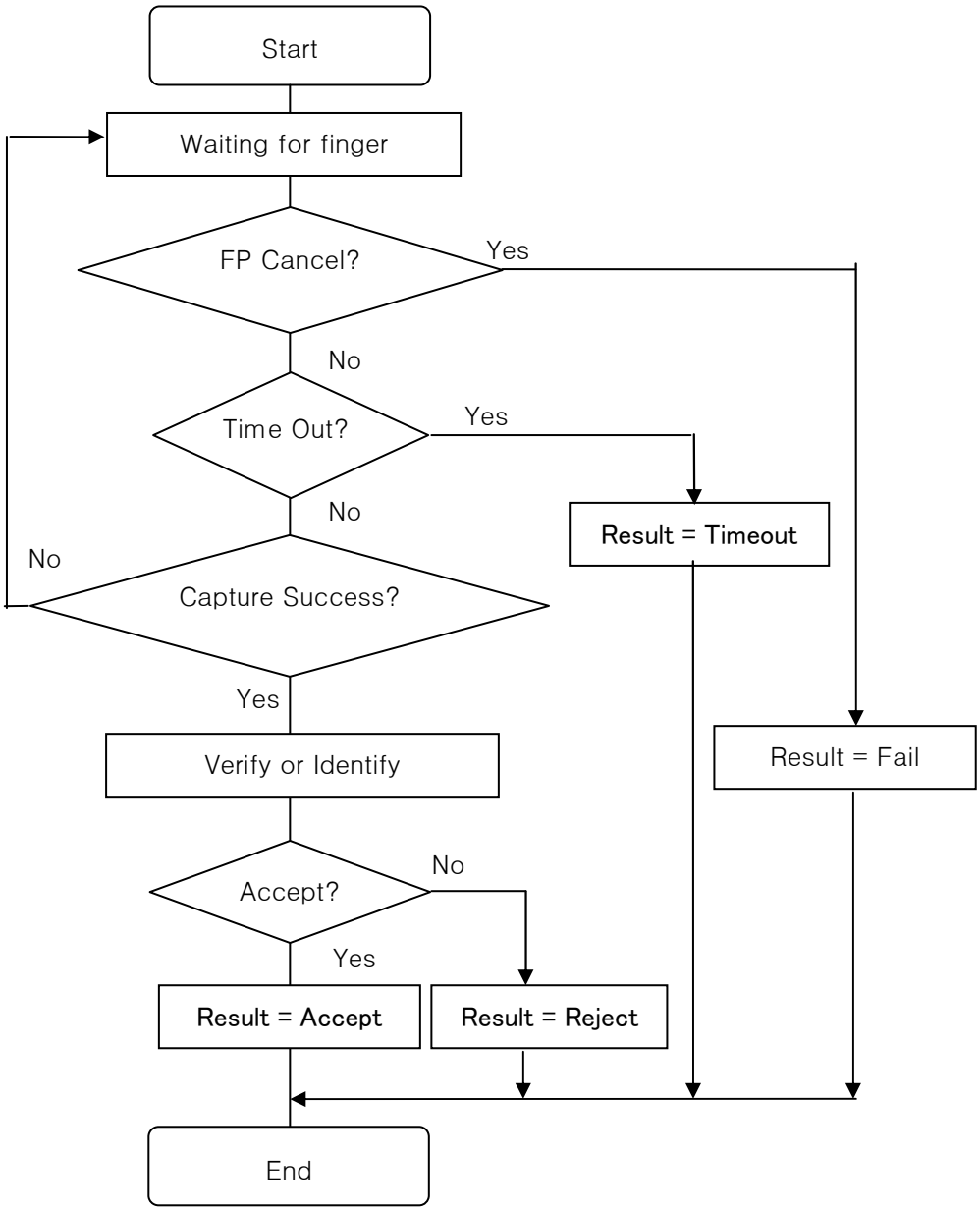
33	GD_DETECT_FINGER	0x01	Detect an finger on sensor when execute “Finger Detect” command
34	GD_NO_DETECT_FINGER	0x00	Detect no any finger on sensor when execute “Finger Detect” command
35	GD_TEMPLATE_NOT_EMPTY	0x01	The appointed Template are not empty
36	GD_TEMPLATE_EMPTY	0x00	The appointed Template have been emptied

6.2 Flow of Enroll and Verify and Identify process

6.2.1 Enroll



6.2.2 Verify & Identify



Contact Information:

Shenzhen CAMA Biometrics Co., Ltd.
Add.: 13/F, Block B, 9th Bldg , Baoneng Science and Technology Park, Qingxiang Road, Longhua, Shenzhen,China
TEL: (86-755)-33581519 FAX:(86-755)-33581529
Website: www.camabio.com