

FPMXXXX

Fingerprint Module User's Guide

Ver: V1.1.10

Date: 2020.09.11

Model: FPMXXXX

Table of Contents

1. Overview.....	4
1.1. Technical Support.....	5
1.2. Application Scope.....	5
2. Main Technical Specifications.....	6
3. Hardware Interface.....	7
3.1. Module External Interface Definition.....	7
3.2. Serial Protocol.....	7
3.3. Power-on/Reset Delay.....	7
4. Communication Protocol.....	8
4.1. Link Layer Description (UART).....	8
4.2. Application Layer Description.....	9
5. Command Set Summary.....	11
5.1. Command Description:.....	13
5.2. Fingerprint Commands:.....	13
5.3. Fingerprint Enrollment.....	14
5.4. Query Enrollment Result :.....	14
5.5. Save Fingerprint Template :.....	16
5.6. Query Save Result:.....	16
5.7. Cancel Enrollment or Matching :.....	17
5.8. Update Stored Template(Self-Learning) :.....	17
5.9. Query Template Update Result:.....	18
5.10. Auto Enrollment:.....	18
5.11. Fingerprint Matching.....	19
5.12. Query Matching Result.....	19
5.13. Fingerprint Matching (Synchronous) :.....	20
5.14. Clear Fingerprint Template :.....	21
5.15. Query Clear Result:	22
5.16. Query Fingerprint ID Existence :	22
5.17. Query Template Storage Map:	23
5.18. Query Finger Presence :	23
5.19. Clear Fingerprint Template (Synchronous):	24
5.20. Enrollment Confirmation:.....	25
5.21. Query Enrollment Confirmation Result :	25
5.22. Download Template Info :	26
5.23. Download Template Data :	26
5.24. Upload Template Info :	26
5.25. Upload Template Data :	27
5.26. Set Password Command :	27
5.27. Reset Fingerprint Module :	28
5.28. Get Fingerprint Template Count:.....	28
5.29. Get Gain:.....	29

5.30. Get Matching Threshold:.....	29
5.31. Set Sleep Mode :.....	30
5.32. Set Enrollment Stitch Count:.....	30
5.33. Set LED Control Info:.....	30
5.34. Get System Policy:.....	31
5.35. Set System Policy:.....	32
5.36. Get Fingerprint Module ID:.....	32
5.37. Heartbeat Command:.....	33
5.38. Set Baud Rate:.....	33
5.39. Set Communication Password :.....	34
6. Appendix.....	35
6.1. Error Code Definition Table:.....	35
6.2. Power-on Operation Flow:.....	36
6.3. Enrollment Flow:.....	37
6.4. Verification Flow:.....	38
6.5. Deletion Flow:.....	39
6.6. Checksum Function Code Example:.....	40
6.7. Reset Function (Single-Supply Module) :.....	40
6.8. Self-Learning Function:.....	40
7. Revision History.....	41

1. Overview

The FPMxxx-xx series fingerprint module is a stable mass-production product launched by Shenzhen iSUN Technology Co., Ltd. in 2015.

The FPMxxx-xx series fingerprint module provides complete fingerprint processing capability, performing fingerprint enrollment, image processing, fingerprint matching, and fingerprint template storage without host computer intervention. Compared with similar fingerprint products, it features the following:

- ❖ Highly integrated module with smaller size and lower power consumption.
- ❖ Uses self-developed fingerprint recognition algorithm and chip, with all performance metrics exceeding comparable products.
- ❖ The fingerprint recognition algorithm supports template stitching to improve user experience.
- ❖ Fingerprint templates are stored inside the algorithm chip, preventing brute-force attacks and enhancing product security.
- ❖ Anti-fog, dust-proof, and tamper-resistant; effectively mitigates ESD, improves image capture quality, and increases durability.
- ❖ The sensor surface color can be customized per customer requirements.
- ❖ Equipped with touch wake-up and effective spoof detection against plastic, silicone, rubber fingers, fingerprint molds, and finger cots.
- ❖ Simple structure and modular design improve stability and consistency, facilitating large-scale mass production.
- ❖ Open interface code and command set enable modular secondary development, reducing customer development effort.
- ❖ Proprietary IP technology provides efficient and flexible secondary development support, fully meeting customer needs without IP disputes.

1.1. Technical Support

ISUN has a complete technical team of fingerprint industry professionals, providing solid technical support and after-sales service for customer development.

1.2. Application Scope

The FPMxxx-xx series fingerprint module is widely applicable to various fingerprint recognition systems, such as:

- ❖ Security: fingerprint door locks, safes, gun cases, finance, etc.;
- ❖ Identity: access control, industrial PCs, POS terminals, driver training, attendance, etc.;
- ❖ Management: private clubs, management software, licensing, etc.;

Customers can develop corresponding fingerprint recognition applications based on the technical documentation in this manual.

2. Main Technical Specifications

Please refer to the specification sheet of each module model.

3. Hardware Interface

3.1. Module External Interface Definition

Please refer to the specification sheet of each module model.

3.2. Serial Protocol

Half-duplex asynchronous UART communication, default baud rate 115200 bps.

The frame format is 10 bits: one low-level start bit, 8 data bits (LSB first), and one stop bit, with no parity bit.

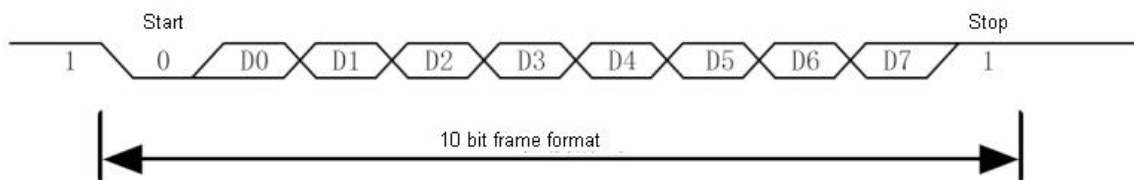


Figure 1. Serial Protocol Diagram

3.3. Power-on/Reset Delay

After power-on, the module requires approximately 200 ms for initialization. During this period, the module cannot respond to host commands.

4. Communication Protocol

The communication protocol defines the rules for data exchange between the FPMxxx-xx series module and the host computer. The hardware uses a UART interface; the software implementation follows the protocol and command set below. This protocol has a two-layer structure: application layer and link layer. The application layer defines specific functions; the link layer handles data transmission. Multi-byte fields (2 or 4 bytes) defined in the protocol use little-endian byte order with the high byte first and low byte last (e.g., 0x12345678 is transmitted as 0x12 0x34 0x56 0x78, not 0x78 0x56 0x34 0x12).

Application Layer
Link Layer (IIC/UART)

4.1. Link Layer Description (UART)

The UART link layer operates in half-duplex point-to-point mode. All commands must be initiated by the host and responded to by the fingerprint module. UART data frame format:

Format	Frame Header	App Layer Data Length	Header Checksum	App Layer Data
Length (bytes)	8	2	1	7+N

Figure 2. UART Data Frame Format Table

Note:

Frame Header: UART start delimiter. Fixed at 0xF1 1F E2 2E B6 6B A8 8A for both host-to-module and module-to-host transmissions;

App Layer Data Length: the actual length of application-layer data, excluding the frame header, length field, and header checksum;

Header Checksum: checksum of frame header + app layer data length, used to verify data length integrity.

4.2. Application Layer Description

The application layer uses a request/response protocol. All commands are initiated by the host and responded to by the fingerprint module, as shown below:

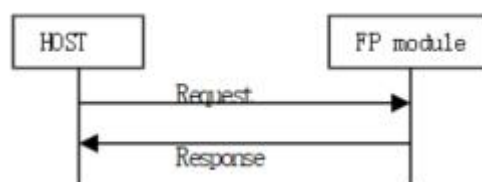


Figure 3. Request/Response Diagram

For ease of host development, the application layer uses fixed-length frames and a sum checksum to ensure data integrity. In practice, the checksum password can be modified to encrypt data and prevent third-party interception. (Note: the checksum password takes effect immediately; keep it safe. If lost, the module must be returned to the factory for initialization.)

Per the application layer protocol, two formats are described below: request command (host transmission) and response (module reply).

Data Transmission:

Format	Checksum Password	Command	Data	Checksum
Length (Byte)	4	2	N(0~n)	1

Figure 4. Data Transmission Table

Note:

Total transmit packet length: (7+N) bytes. Packet length varies by command, but each instruction has a fixed length.

Checksum Password: secures data transmission. Default is 0x00000000 and can be changed via an application-layer command.

Command:

Data Content: the actual payload to transmit.

Checksum: sum checksum, byte-wise operation. (See appendix

for checksum details.) Response:

Format	Checksum Password	Response command	Error code	Data	Checksum
Length (Byte)	4	2	4	N(0~n)	1

Figure 5. Response Table

Note:

Total receive packet length: $(11+N)$ bytes. Packet length varies by command, but each instruction has a fixed length.

Checksum Password: secures data transmission. Default is 0x00000000 and can be changed via an application-layer command.

Response Command: corresponds to the command field sent by the host, indicating the command being acknowledged.

Error Code: returns the execution result of the command. 0x00000000 indicates no error; others refer to the error code definition table. Data Content: the response result.

Checksum: sum checksum, byte-wise operation. (See appendix for checksum details.)

5. Command Set Summary

Host					Fingerprint Module			
Command Type	Sub-command	CMD1	CMD2	Data (Len)	ACK1	ACK2	Error Code	Data (Len)
Fingerprint	Fingerprint Enrollment	0x01	0x11	1	0x01	0x11	OK: 0x00000000 Error: returns error code	0
	Query Enrollment Result	0x01	0x12	0	0x01	0x12	OK: 0x00000000 Error: returns error code	3
	Save Fingerprint	0x01	0x13	2	0x01	0x13	OK: 0x00000000 Error: returns error code	0
	Query Save Result	0x01	0x14	0	0x01	0x14	OK: 0x00000000 Error: returns error code	0
	Cancel Enrollment	0x01	0x15	0	0x01	0x15	OK: 0x00000000 Error: returns error code	0
	Update Stored Template	0x01	0x16	2	0x01	0x16	OK: 0x00000000 Error: returns error code	0
	Query Template Update Result	0x01	0x17	0	0x01	0x17	OK: 0x00000000 Error: returns error code	0
	Auto Enrollment	0x01	0x18	4	0x01	0x18	OK: 0x00000000 Error: returns error code	4
	Fingerprint Matching	0x01	0x21	0	0x01	0x21	OK: 0x00000000 Error: returns error code	0
	Query Matching Result	0x01	0x22	0	0x01	0x22	OK: 0x00000000 Error: returns error code	6
	Fingerprint Matching (Synchronous)	0x01	0x23	0	0x01	0x23	OK: 0x00000000 Error: returns error code	6
	Clear Fingerprint Template	0x01	0x31	N	0x01	0x31	OK: 0x00000000 Error: returns error code	0

	Query Fingerprint Clear Result	0x01	0x32	0	0x01	0x32	OK: 0x00000000 Error: returns error code	0
	Query Fingerprint ID Exist Present	0x01	0x33	2	0x01	0x33	OK: 0x00000000 Error: returns error code	3
	Query Template Storage Map	0x01	0x34	0	0x01	0x34	OK: 0x00000000 Error: returns error code	66
	Query Finger Presence	0x01	0x35	0	0x01	0x35	OK: 0x00000000 Error: returns error code	1
	Clear Template (Sync)	0x01	0x36	N	0x01	0x36	OK: 0x00000000 Error: returns error code	0
	Enrollment Confirmation	0x01	0x41	0	0x01	0x41	OK: 0x00000000 Error: returns error code	0
	Query Enrollment Confirm Result	0x01	0x42	0	0x01	0x42	OK: 0x00000000 Error: returns error code	6
	Download Template Info	0x01	0x51	4	0x01	0x51	OK: 0x00000000 Error: returns error code	0
	Download Template Data	0x01	0x52	N	0x01	0x52	OK: 0x00000000 Error: returns error code	0
	Upload Template Info	0x01	0x53	2	0x01	0x53	OK: 0x00000000 Error: returns error code	2
	Upload Template Data	0x01	0x54	2	0x01	0x53	OK: 0x00000000 Error: returns error code	N
System	Set Password	0x02	0x01	4	0x02	0x01	OK: 0x00000000 Error: returns error code	0
	Reset Module	0x02	0x02	0	0x02	0x02	OK: 0x00000000 Error: returns error code	0
	Get Template Count	0x02	0x03	0	0x02	0x03	OK: 0x00000000 Error: returns error code	2
	Get Gain	0x02	0x09	0	0x02	0x09	OK: 0x00000000 Error: returns error code	3

	Get Matching Threshold	0x02	0x0B	0	0x02	0x0B	OK: 0x00000000 Error: returns error code	2
	Sleep Mode	0x02	0x0C	1	0x02	0x0C	OK: 0x00000000 Error: returns error code	0
	Set Max Enrollment Count	0x02	0x0D	1	0x02	0x0D	OK: 0x00000000 Error: returns error code	0
	Set LED Control Info	0x02	0x0F	5	0x02	0x0F	OK: 0x00000000 Error: returns error code	0
	Get System Policy	0x02	0xFB	0	0x02	0xFB	OK: 0x00000000 Error: returns error code	4
	Set System Policy	0x02	0xFC	4	0x02	0xFC	OK: 0x00000000 Error: returns error code	0
Maintenance	Read ID	0x03	0x01	0	0x03	0x01	OK: 0x00000000 Error: returns error code	16
	Heartbeat	0x03	0x03	0	0x03	0x03	OK: 0x00000000 Error: returns error code	0
	Set Baud Rate	0x03	0x04	4	0x03	0x04	OK: 0x00000000 Error: returns error code	0
	Set Communication Password	0x03	0x05	4	0x03	0x05	OK: 0x00000000 Error: returns error code	0

Figure 6. Command Set Summary Table

5.1. Command Description:

The FPMxxx-xx command set is optimized for usage scenarios. Customers only need simple integration to support fingerprint recognition in various applications.

5.2. Fingerprint Commands:

Covers core functions: fingerprint enrollment, matching, and template deletion. If the host uses a different fingerprint algorithm, the fingerprint image capture function can be used to acquire images.

All examples below use the default checksum password 0x00000000. After changing the password via the password command, the new password and checksum must be used.

5.3. Fingerprint Enrollment

Transmit Packet Format (Byte):

REG_IDX indicates the press count during enrollment, starting from 0x01. The response only confirms the command has been received by the module, not that enrollment succeeded. Enrollment success is determined by querying the enrollment result: when PROC exceeds 100, enrollment is complete. This is template stitching, consistent with smart phone fingerprint enrollment. Users are advised to move the finger slightly between presses for better unlock experience. See the operation flow section for details. Recommendation: REG_IDX starts at 0x01 and increments to 0x06 (max 6 presses). If PROC is still not above 100 after 6 presses, continue increase REG_IDX as needed. Note: if REG_IDX reaches the maximum but PROC is still below 100, the template can still be saved, but it will be incomplete with poor user experience. Command 0x11 (enrollment) must be used together with command 0x12 (query enrollment result).

Checksum Password				CMD1	CMD2	REG_IDX	Checksum
0x00	0x00	0x00	0x00	0x01	0x11	0x01	0xED

Transmit Packet Example:

Press Count	Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		REG _IDX	Checksum
1st	F1	1F	E2	2E	B6	6B	A8	8A	00	08	85	00	00	00	00	01	11	01	ED
2nd	F1	1F	E2	2E	B6	6B	A8	8A	00	08	85	00	00	00	00	01	11	02	EC
3rd	F1	1F	E2	2E	B6	6B	A8	8A	00	08	85	00	00	00	00	01	11	03	EB
4th	F1	1F	E2	2E	B6	6B	A8	8A	00	08	85	00	00	00	00	01	11	04	EA
5th	F1	1F	E2	2E	B6	6B	A8	8A	00	08	85	00	00	00	00	01	11	05	E9
6th	F1	1F	E2	2E	B6	6B	A8	8A	00	08	85	00	00	00	00	01	11	06	E8

Figure 7. Enrollment Transmit Packet Format Table

Response Packet Format (Byte):

Checksum Password				CMD1	CMD2	Completion Code				Checksum
0x00	0x00	0x00	0x00	0x01	0x11	0x00	0x00	0x00	0x00	0xEE

Response Packet Example:

Success (identical for all 6 presses):

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Error Code				Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	00	00	00	00	01	11	00	00	00	00	EE

Failure (system busy or other error code):

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Error Code				Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	00	00	00	00	01	11	00	00	00	04	EA

...

5.4. Query Enrollment Result :

Transmit Packet Format (Byte):

Transmit Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	07	86	00	00	00	00	01	12	ED

Figure 8. Query Enrollment Result Transmit Packet Format Table

Response Packet Format (Byte):

When enrollment is incomplete, returns the actual ID and PROC < 100:

Checksum Password (4 Byte)				CMD1	CMD2	Completion Code				ID_H	ID_L	PROC	Checksum
0x00	0x00	0x00	0x00	0x01	0x12	0x00	0x00	0x00	0x00	0x00	0x01	0x14	0xD8

Response Packet Example:

Success (ID1 PROC16):

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		Error Code				ID_H	ID_L	PROC	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0E	7F	00	00	00	00	01	12	00	00	00	00	00	01	10	DC

Success (ID1 PROC32):

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Error Code				ID_H	ID_L	PR OC	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0E	7F	00	00	00	00	01	12	00	00	00	00	00	01	20	CC

Figure 9. Query Enrollment Result — Incomplete Response Packet Format Table

...

Failure (no finger detected, timeout exit, etc.):

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Error Code				ID_H	ID_L	PROC	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0E	7F	00	00	00	00	01	12	00	00	00	08	00	00	00	E5

...

When enrollment is complete, returns the actual ID (the smallest unused ID) and PROC = 100

Checksum Password (4 Byte)				CMD1	CMD2	Completion Code				ID_H	ID_L	PROC	Checksum
0x00	0x00	0x00	0x00	0x01	0x12	0x00	0x00	0x00	0x00	0x00	0x01	0x64	0x88

Response Packet Example:

Success (ID1 PROC100):

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		Error Code				ID_H	ID_L	PROC	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0E	7F	00	00	00	00	01	12	00	00	00	00	00	01	64	88

Success (ID2 PROC100):

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Error Code				ID_H	ID_L	PROC	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0E	7F	00	00	00	00	01	12	00	00	00	00	00	02	64	87

Figure10. Query Enrollment Result — Complete Response Packet Format Table

Enrollment Result Error Codes			
No.	Error Code	Description	Suggested Action
1	0x00000000	OK	Current enrollment step complete; check enrollment progress
2	0x00000003	Invalid command field	Check the enrollment start command and resend it.
3	0x00000004	System busy	System is executing another command; wait briefly and query again.
4	0x00000005	Result queried without sending the corresponding request	Resend the enrollment command, starting from press count 1.
5	0x00000006	System software error	Template data may be too large during stitching; re-enroll this step.
6	0x00000007	Hardware error	SPI communication may have failed; retry several times. If the issue persists, contact us.
7	0x00000008	No finger press detected, timed out	Timeout is currently 10 s. If no finger is detected within 10 s, re-enroll this step.
8	0x00000009	Fingerprint extraction error, possible cause: poor image quality	Indicate extraction error, possibly poor image quality. Re-enroll this step.
9	0x0000000B	Storage full	Indicate storage full. Delete unneeded fingerprints and re-enroll.
10	0x0000000C	Storage write failed	Indicate storage write failure. Diagnose and re-enroll.
11	0x0000000D	Storage read failed	Indicate storage read failure. Diagnose and re-enroll.
12	0x0000000E	Captured fingerprint image quality is poor	Indicate poor image quality. Finger pressure may be too heavy/light or finger too moist; re-enroll.
13	0x0000000F	Duplicate fingerprint template	Indicate duplicate template; this fingerprint is already enrolled. Enroll a new one.
14	0x00000010	Capture area too small	Indicate small capture area. Pressure may be too light or finger too moist; enroll a new one.

15	0x00000011	Movement range too large	Indicate finger movement too large. Re-enroll this step.
16	0x00000012	Movement range too small	Indicate finger movement too small. Re-enroll this step.
17	0x00000015	Forced exit	Resend the enrollment start command.
18	0x000000FF	Other error	Rarely occurs. If it does, try re-enrolling; if it recurs, contact us.

Figure 11. Enrollment Result Error Code Table

5.5. Save Fingerprint Template :

Transmit Packet Format (Byte):

The save ID can be specified. Command 0x13 (save template) must be used together with command 0x14 (query save result).

Checksum Password				CMD1	CMD2	ID_H
0x00	0x00	0x00	0x00	0x01	0x13	0x00
ID_L	Checksum					
0x01	0xEB					

Transmit Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		ID _H	ID _L	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	09	84	00	00	00	00	01	13	00	01	EB

Figure 12. Save Template Transmit Packet Format Table

Response Packet Format (Byte):

Checksum Password				CMD1	CMD2	Completion Code				Checksum
0x00	0x00	0x00	0x00	0x01	0x13	0x00	0x00	0x00	0x00	0xEC

Response Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Error Code				Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	00	00	00	00	01	13	00	00	00	00	EC

Figure 13. Save Template Response Packet Format Table

5.6. Query Save Result:

Transmit Packet Format (Byte):

Transmit Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	07	86	00	00	00	00	01	14	EB

Figure 14. Query Save Result Transmit Packet Format Table

Response Packet Format (Byte):

On success, the returned ID is the saved ID. Error code 0x0F indicates a duplicate ID (duplicate ID transmitted high-byte-first, low-byte-last).

Checksum Password				CMD1	CMD2	Completion Code				ID_H	ID_L	Checksum
0x00	0x00	0x00	0x00	0x01	0x14	0x00	0x00	0x00	0x00	0x00	0x01	0xEA

Response Example — Success:

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		Error Code				ID_H	ID_L	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0D	80	00	00	00	00	01	14	00	00	00	00	00	01	EA

Failure (duplicate fingerprint, ID is the duplicate ID):

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Error Code				ID _H	ID _L	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0D	80	00	00	00	00	01	14	00	00	00	0F	00	00	DC

Figure 15. Query Save Result — Response Packet Format Table

Save Result Error Codes			
No.	Error Code	Description	Suggested Action
1	0x00000000	OK	Save successful
2	0x00000004	System busy	System is executing another command; wait briefly and query again.
3	0x00000005	Result queried without sending the corresponding request	Send the save command first, then query the result; re-save and query again.
4	0x00000006	System software error	Indicate system software error. Re-save.
5	0x00000007	Hardware error	The save ID may exceed the maximum storable value. Re-save.
6	0x00000009	Fingerprint extraction error, possible cause: poor image quality	Indicate extraction error. Re-enroll and save.
7	0x0000000C	Storage write failed	Indicate storage write failure. Diagnose and re-save.
8	0x0000000D	Storage read failed	Indicate storage read failure. Diagnose and re-query save result.
9	0x0000000B	Storage full	Indicate storage full. Delete unneeded fingerprints and re-save.
10	0x0000000E	Captured fingerprint image quality is poor	Indicate poor captured image quality. Re-enroll and save.
11	0x0000000F	Duplicate fingerprint template	Indicate duplicate template; this fingerprint is already enrolled.
12	0x000000FF	Other error	Enrollment may have been forcibly stopped; re-save.

Figure 16. Save Result Error Code Table

5.7. Cancel Enrollment or Matching :

Transmit Packet Format (Byte):

Checksum Password				CMD1	CMD2	Checksum
0x00	0x00	0x00	0x00	0x01	0x15	0xEA

Transmit Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	07	86	00	00	00	00	01	15	EA

Figure 17. Cancel Enrollment Transmit Packet Format Table

Response Packet Format (Byte):

Checksum Password				CMD1	CMD2	Completion Code				Checksum
0x00	0x00	0x00	0x00	0x01	0x15	0x00	0x00	0x00	0x00	0xEA

Response Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Error Code				Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	00	00	00	00	01	15	00	00	00	00	EA

Figure 18. Cancel Enrollment Response Packet Format Table

5.8. Update Stored Template(Self-Learning):

Transmit Packet Format (Byte):

ID is the template ID to update. Self-learning requires this command: send it after each successful match to update the template, Command 0x16 (update template) must be used together with command 0x17 (query template update result).

Checksum Password (4 Byte)				CMD1	CMD2	ID_H	ID_L	Checksum
0x00	0x00	0x00	0x00	0x01	0x16	0x00	0x00	0xE9

Transmit Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		ID_H	ID_L	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	09	84	00	00	00	00	01	16	00	00	E9

Figure 19. Update Template Transmit Packet Format Table

Response Packet Format (Byte):

Checksum Password (4 Byte)				CMD1	CMD2	Completion Code (4 Byte)				Checksum
0x00	0x00	0x00	0x00	0x01	0x16	0x00	0x00	0x00	0x00	0xE9

Response Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Error Code				Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	00	00	00	00	01	16	00	00	00	00	E9

Figure 20. Update Template Response Packet Format Table

5.9. Query Template Update Result:

Transmit Packet Format (Byte):

Checksum Password				CMD1	CMD2	Checksum
0x00	0x00	0x00	0x00	0x01	0x17	0xE8

Transmit Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	07	86	00	00	00	00	01	17	E8

Figure 21. Query Template Update Result Transmit Packet Format Table

Response Packet Format (Byte):

Checksum Password (4 Byte)				CMD1	CMD2	Completion Code (4 Byte)				Checksum
0x00	0x00	0x00	0x00	0x01	0x17	0x00	0x00	0x00	0x00	0xE8

Response Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Error Code				Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	00	00	00	00	01	17	00	00	00	00	E8

Figure 22. Query Template Update Result Response Packet Format Table

5.10. Auto Enrollment:

Transmit Packet Format (Byte):

By passing the total press count and fingerprint ID, the module automatically performs image capture, extraction, stitching, and saving. Command response timeout is 10 s. The wait-finger flag controls whether to wait for finger lift after each press: 1 = wait for finger lift before the next press, 0 = do not detect finger lift and continue enrolling until progress reaches 100. Press count can be set from 1 to 6. The user can specify the enrollment ID (an existing ID can be specified and overwritten); if ID is 0xFFFF, the system automatically assigns an unused ID.

Checksum Password				CMD1	CMD2	Wait Finger	Press Count	ID_H	ID_L	Checksum
0x00	0x00	0x00	0x00	0x01	0x18	0x00	0x03	0xFF	0xFF	0xE6

Transmit Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Wait Finger	Press Count	ID_H	ID_L	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	00	00	00	00	01	18	00	03	FF	FF	E6

Response Packet Format (Byte):

On success, returns the fingerprint ID. The response is returned in multiple packets: one after each capture, and when the wait-finger flag is 1, the result is returned first then the module waits for finger lift; one final response after saving. In the response, 'count' is the current capture number; 0xFF indicates the save response. If the transmit ID was 0xFFFF, the response ID is the smallest available ID; otherwise it is the user-specified ID. Progress is the enrollment progress, max 100.

Checksum Password				CMD1	Completion Code (4 byte)				Count	ID_H	ID_L	Progress	Checksum
0x00	0x00	0x00	0x00	0x01	0x00	0x00	0x00	0x00	0x01	0x00	0x00	0x21	0xC5

Response Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Error Code				Count	ID_H	ID_L	Progress	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0F	7E	00	00	00	00	01	18	00	00	00	00	01	00	00	21	C5
F1	1F	E2	2E	B6	6B	A8	8A	00	0F	7E	00	00	00	00	01	18	00	00	00	00	02	00	00	42	A3
F1	1F	E2	2E	B6	6B	A8	8A	00	0F	7E	00	00	00	00	01	18	00	00	00	00	03	00	00	64	80
F1	1F	E2	2E	B6	6B	A8	8A	00	0F	7E	00	00	00	00	01	18	00	00	00	00	FF	00	00	64	84

5.11. Fingerprint Matching

Transmit Packet Format (Byte):

Command 0x21 (matching) must be used together with command 0x22 (query matching result):

Checksum Password				CMD1		CMD2		Checksum	
0x00	0x00	0x00	0x00	0x01	0x21	0x01	0x21	0x01	0xDE

Transmit Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Checksum	
F1	1F	E2	2E	B6	6B	A8	8A	00	07	86	00	00	00	00	01	21	DE	

Figure 23. Matching Transmit Packet Format Table

Response Packet Format (Byte):

Checksum Password				CMD1		CMD2		Completion Code				Checksum	
0x00	0x00	0x00	0x00	0x01	0x21	0x01	0x21	0x00	0x00	0x00	0x00	0x01	0xDE

Response Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Error Code				Checksum	
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	00	00	00	00	01	21	00	00	00	00	DE	

Figure 24. Matching Response Packet Format Table

5.12. Query Matching Result

Transmit Packet Format (Byte):

Checksum Password				CMD1		CMD2		Checksum			
0x00	0x00	0x00	0x00	0x01	0x22	0x01	0x22	0x01	0x22	0x01	0xDD

Transmit Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Checksum	
F1	1F	E2	2E	B6	6B	A8	8A	00	07	86	00	00	00	00	01	22	DD	

Figure 25. Query Matching Result Transmit Packet Format Table

Response Packet Format (Byte):

Contains three fields: match result, match score, and matched fingerprint ID:

Checksum Password (4 Byte)				CMD1		CMD2		Completion Code (4 Byte)			
0x00	0x00	0x00	0x00	0x01	0x22	0x01	0x22	0x00	0x00	0x00	0x00
Match Result		Match Score		Matched Fingerprint ID		Checksum					
0x00	0x01	0x27	0x0F	0x00	0x03	0xA3					

Response Example

Success:

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		Error Code				Match Result		Match Score		Match ID		Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	11	7C	00	00	00	00	01	22	00	00	00	00	00	01	27	0F	00	03	A3

Failure (matching error, possible cause: poor image quality, etc.):

Frame Header								Data (Len)	Header Checksum	Checksum Password				Command				Error Code				Match Result		Match Score		Match ID		Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	11	7C	00	00	00	00	01	22	00	00	00	0A	00	00	00	00	00	00	00	D3

Failure (system busy or other error):

Frame Header								Data (Len)	Frame Header Check	Checksum Password				Command				Error Code				Checksum			
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	00	00	00	00	01	22	00	00	00	04	00	00	00	04	D9

Figure 26. Query Matching Result Response Packet Format Table

...

Matching Result Error Codes			
No.	Error Code	Description	Suggested Action
1	0x00000000	OK	Match successful; view matched ID and score
2	0x00000004	System busy	System is executing another command; wait briefly and query again.
3	0x00000005	Result queried without sending the corresponding request	Send the match command first, then query the result.
4	0x00000006	System software error	Indicate system software error. Re-match.
5	0x00000007	Hardware error	Indicate hardware error. Re-match.
6	0x00000008	No finger press detected, timed out	Indicate no finger pressed. Re-match.
7	0x00000009	Fingerprint extraction error, possible cause: poor image quality	Indicate extraction error, possibly poor image quality. Re-match.
8	0x0000000A	Template library is empty	Enroll a fingerprint before matching.
9	0x0000000E	Captured fingerprint image quality is poor	Indicate poor image quality. Pressure may be too heavy/light or finger too moist; re-match.

10	0x00000010	Capture area too small	Indicate small capture area. Pressure too light or incorrect placement; re-match.
11	0x000000FF	Other error	Re-match.

Figure 27.

Matching Result Error Code Table

5.13. Fingerprint Matching (Synchronous) :

Transmit Packet Format (Byte):

Checksum Password				CMD1	CMD2	Checksum
0x00	0x00	0x00	0x00	0x01	0x23	0xDC

Transmit Packet Example:

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	07	86	00	00	00	00	01	23	DC

Figure 28. Query Matching Result Transmit Packet Format Table

Response Packet Format (Byte):

Checksum Password (4 Byte)				CMD1	CMD2	Completion Code (4 Byte)			
0x00	0x00	0x00	0x00	0x01	0x23	0x00	0x00	0x00	0x00
Match Result		Match Score		Matched Fingerprint ID		Checksum			
0x00	0x01	0x27	0x0F	0x00	0x03	0xA2			

Response Example:

Success:

Frame Header								Data (Len)		Frame Header Check		Checksum Password				Command		Error Code				Match Result		Match Score		Match ID		Checksum	
F1	1F	E2	2E	B6	6B	A8	8A	00	11	7C	00	00	00	00	01	23	00	00	00	00	00	01	27	0F	00	03	A2		

Failure (matching error, possible cause: poor image quality, etc.):

Frame Header								Data (Len)	Header Checksum	Checksum Password				Command				Error Code				Match Result		Match Score		Match ID		Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	11	7C	00	00	00	00	01	23	00	00	00	0A	00	00	00	00	00	00	00	D2

5.14. Clear Fingerprint Template :

Transmit Packet Format (Byte):

CL_FLAG is the clear mode flag: 0x01 = clear all fingerprints, 0x02 = clear multiple fingerprints,

0x03 = clear a block of fingerprints, default 0x00 = clear a single fingerprint. ID_H, ID_L are the fingerprint ID to delete

(IDs are assigned starting from 0). Command 0x31 (clear template) must be used together with command 0x32 (query clear result):

Clear All or Single Template — Transmit Packet Format:

Checksum Password (4 Byte)				CMD1	CMD2	CL_FLAG	ID_H	ID_L	Checksum
0x00	0x00	0x00	0x00	0x01	0x31	0x00	0x00	0x01	0xCD

Transmit Packet Example:

Clear All Fingerprints:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		CL_FLAG	ID_H	ID_L	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0A	83	00	00	00	00	01	31	01	00	01	CC

Clear Single Fingerprint:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		CL_FLAG	ID_H	ID_L	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0A	83	00	00	00	00	01	31	00	00	01	CD

Clear Multiple Templates — Transmit Packet Format:

COUNT is the number of fingerprints to delete; IDn are the template IDs (assigned from 0).

Checksum Password (4 Byte)				CMD1	CMD2	CL_FLAG	COUNT_H	COUNT_L
0x00	0x00	0x00	0x00	0x01	0x31	0x02	0xXX	0xXX
ID1_H	ID1_L	ID2_H	ID2_L			IDn_H	IDn_L	Checksum
0x00	0x01	0x00	0x02			0x00	0xXX	0xXX

Transmit Packet Example:

Clear Multiple Fingerprints:

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		CL_FLAG	COUNT_H	COUNT_L
F1	1F	E2	2E	B6	6B	A8	8A	00	0E	7F	00	00	00	00	01	31	02	00	02
ID1_H	ID1_L	ID2_H	ID2_L	Checksum															
00	01	00	02	C7															

Block clear: CL_FLAG = 0x03, pass the first and last ID to delete.

Checksum Password (4 Byte)				CMD1	CMD2	CL_FLAG	FIRST_ID_H	FIRST_ID_L
0x00	0x00	0x00	0x00	0x01	0x31	0x03	00	02
LAST_ID_H	LAST_ID_L	Checksum						
0x00	0x06	0xC3						

Transmit Packet Example:

Clear Fingerprint Template (Block Delete):

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		CL_FLAG	FIRST_ID_H	FIRST_ID_L
F1	1F	E2	2E	B6	6B	A8	8A	00	0C	81	00	00	00	00	01	31	03	00	02
LAST_ID_H	LAST_ID_L	Checksum																	
00	06	C3																	

Figure 28. Clear Template Transmit Packet Format Table

Response Packet Format (Byte):

Checksum Password (4 Byte)				CMD1	CMD2	Completion Code (4 Byte)				Checksum
0x00	0x00	0x00	0x00	0x01	0x31	0x00	0x00	0x00	0x00	0xCE

Response Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Error Code				Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	00	00	00	00	01	31	00	00	00	00	CE

Figure 29. Clear Template Response Packet Format Table

5.15. Query Clear Result:

Transmit Packet Format (Byte), no data field:

Checksum Password				CMD1		CMD2		Checksum
0x00	0x00	0x00	0x00	0x01		0x32		0xCD

Transmit Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Error Code				Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	07	86	00	00	00	00	01	32					CD

Figure 30. Query Clear Result Transmit Packet Format Table

Response Packet Format (Byte):

Checksum Password (4 Byte)				CMD1		CMD2		Completion Code (4 Byte)				Checksum
0x00	0x00	0x00	0x00	0x01		0x32		0x00	0x00	0x00	0x00	0xCD

Response Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Error Code				Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	00	00	00	00	01	32	00	00	00	00	CD

Figure 31. Query Clear Result Response Packet Format Table

Deletion Result Error Codes			
No.	Error Code	Description	Suggested Action
1	0x00000000	OK	Deletion successful
2	0x00000001	Unrecognized command	Indicate unrecognized command. Verify the command and resend.
3	0x00000002	Invalid command field length	Indicate invalid command field length. Verify and resend.
4	0x00000003	Invalid command field	Indicate invalid command field. Verify and resend.
5	0x00000004	System busy	System is executing another command; wait briefly and query again.

Figure 32. Deletion Result Error Code Table

5.16. Query Fingerprint ID Existence :

Transmit Packet Format (Byte):

ID_H, ID_L are the fingerprint ID to query:

Checksum Password (4 Byte)				CMD1		CMD2		ID_H	ID_L	Checksum
0x00	0x00	0x00	0x00	0x01		0x33		0x00	0x01	0xCB

Transmit Packet Example:

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		ID_H	ID_L	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	09	84	00	00	00	00	01	33	00	01	CB

Figure 33. Query Fingerprint ID Existence Transmit Packet Format Table

Response Packet Format (Byte):

On success the completion code is 0x00000000; otherwise the query failed. If the queried fingerprint exists, STATE returns 1, otherwise STATE = 0; ID_H, ID_L in the data are the queried fingerprint ID:

Checksum Password (4 Byte)				CMD1		CMD2		Completion Code (4 Byte)				STATE	ID_H	ID_L	Check
0x00	0x00	0x00	0x00	0x01		0x33		0x00	0x00	0x00	0x00	0x01	0x00	0x01	0xCA

Response Example :

Exists:

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		Error Code				STA TE	ID _H	ID _L	Check sum
F1	1F	E2	2E	B6	6B	A8	8A	00	0E	7F	00	00	00	00	01	33	00	00	00	00	01	00	01	CA

Does not exist:

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		Error Code				STA TE	ID_H	ID_L	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0E	7F	00	00	00	00	01	33	00	00	00	00	00	00	01	CB

Figure 34. Query Fingerprint ID Existence Response Packet Format Table

5.17. Query Template Storage Map:

Transmit Packet Format (Byte):

Checksum Password (4 Byte)				CMD1		CMD2		Checksum	
0x00	0x00	0x00	0x00	0x01		0x34		0xCB	

Transmit Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Checksum	
F1	1F	E2	2E	B6	6B	A8	8A	00	07	86	00	00	00	00	01	34	CB	

Figure 35. Query Template Storage Map Transmit Packet Format Table

Response Packet Format (Byte):

On success the completion code is 0x00000000; otherwise the query failed. NUM_H, NUM_L are the total number of stored templates. INFO_0 ~ INFO_63 indicate storage status: fingerprints 0 ~ 7 correspond to bits 0 ~ 7 of INFO_0, fingerprints 8 ~ 15 to bits 0 ~ 7 of INFO_1, and so on. A bit is set to 1 if the fingerprint exists, otherwise 0.

Checksum Password (4 Byte)				CMD1	CMD2	Completion Code (4 Byte)				NUM_H	NUM_L
0x00	0x00	0x00	0x00	0x01	0x34	0x00	0x00	0x00	0x00	0x20	0x00
INFO_0	INFO_1	INFO_2	...							INFO_63	Checksum
0xFF	0x00	0x00	...							0x00	chksum

Response Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Error Code			
F1	1F	E2	2E	B6	6B	A8	8A	00	4D	40	00	00	00	00	01	34	00	00	00	
NUM_H	NUM_L	INFO_0		INFO_1		INFO_2										INFO_63		Checksum		
20	00	01		00		00										00		C8		

Figure 36. Query Template Storage Map Response Packet Format Table

5.18. Query Finger Presence :

Transmit Packet Format (Byte):

Checksum Password (4 Byte)				CMD1		CMD2		Checksum	
0x00	0x00	0x00	0x00	0x01		0x35		0xCA	

Transmit Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Checksum	
F1	1F	E2	2E	B6	6B	A8	8A	00	07	86	00	00	00	00	01	35	CA	

Figure 37. Query Finger Presence Transmit Packet Format Table

Response Packet Format (Byte):

On success the completion code is 0x00000000; otherwise the query failed. If a finger is present, STATE = 1, otherwise STATE = 0:

Checksum Password (4 Byte)				CMD1	CMD2	Completion Code (4 Byte)				STATE	Check
0x00	0x00	0x00	0x00	0x01	0x35	0x00	0x00	0x00	0x00	0x01	0xC9

Response Example:

Present:

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		Error Code				STATE	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0C	81	00	00	00	00	01	35	00	00	00	00	01	C9

Not present:

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		Error Code				STATE	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0C	81	00	00	00	00	01	35	00	00	00	00	00	CA

Figure 38. Query Finger Presence Response Packet Format Table

5.19. Clear Fingerprint Template (Synchronous):

Transmit Packet Format (Byte):

CL_FLAG is the clear mode flag: 0x01 = clear all fingerprints, 0x02 = clear multiple fingerprints, 0x03 = clear a block of fingerprints, default 0x00 = clear a single fingerprint. ID_H, ID_L are the fingerprint ID to delete (IDs assigned from 0):

Clear All or Single Template — Transmit Packet Format:

Checksum Password (4 Byte)				CMD1	CMD2	CL_FLAG	ID_H	ID_L	Checksum
0x00	0x00	0x00	0x00	0x01	0x36	0x00	0x00	0x01	0xC8

Transmit Packet Example:

Clear All Fingerprints:

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		CL_FLAG	ID_H	ID_L	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0A	83	00	00	00	00	01	36	01	00	01	C7

Clear Single Fingerprint:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		CL_FLAG	ID_H	ID_L	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0A	83	00	00	00	00	01	36	00	00	01	C8

Clear Multiple Templates — Transmit Packet Format:

COUNT is the number of fingerprints to delete; IDn are the template IDs (assigned from 0).

Checksum Password (4 Byte)				CMD1	CMD2	CL_FLAG	COUNT_H	COUNT_L
0x00	0x00	0x00	0x00	0x01	0x36	0x02	0xXX	0xXX
ID1_H	ID1_L	ID2_H	ID2_L			IDn_H	IDn_L	Checksum
0x00	0x01	0x00	0x02			0x00	0xXX	0xXX

Transmit Packet Example:

Clear Multiple Fingerprints:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		CL_FLAG	COUNT_H	COUNT_L
F1	1F	E2	2E	B6	6B	A8	8A	00	0E	7F	00	00	00	00	01	36	02	00	02
ID1_H	ID1_L	ID2_H	ID2_L	Checksum															
00	01	00	02	C2															

Block clear: CL_FLAG = 0x03, pass the first and last ID to delete.

Checksum Password (4 Byte)				CMD1	CMD2	CL_FLAG	FIRST_ID_H	FIRST_ID_L
0x00	0x00	0x00	0x00	0x01	0x36	0x03	0x00	0x02
LAST_ID_H	LAST_ID_L	Checksum						
0x00	0x06	0xBE						

Transmit Packet Example:

Clear Fingerprint Template (Block Delete):

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		CL_FLAG	FIRST_ID_H	FIRST_ID_L
F1	1F	E2	2E	B6	6B	A8	8A	00	0C	81	00	00	00	00	01	36	03	00	02
LAST_ID_H	LAST_ID_L	Checksum																	
00	06	BE																	

Figure 39. Clear Template Transmit Packet Format Table

Response Packet Format (Byte):

Checksum Password (4 Byte)				CMD1	CMD2	Completion Code (4 Byte)				Checksum
0x00	0x00	0x00	0x00	0x01	0x36	0x00	0x00	0x00	0x00	0xC9

Response Packet Example:

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		Error Code				Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	00	00	00	00	01	36	00	00	00	00	C9

Figure 40. Query Clear Result Response Packet Format Table

5.20. Enrollment Confirmation:

This command is used after successful enrollment to require the user to confirm enrollment success, per industry application requirements.

Transmit Packet Format (Byte):

Command 0x41 (enrollment confirmation) must be used together with command 0x42 (query confirmation result):

Checksum Password (4 Byte)				CMD1		CMD2		Checksum	
0x00	0x00	0x00	0x00	0x01	0x41	0x00	0x41	0xBE	

Transmit Packet Example:

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	07	86	00	00	00	00	01	41	BE

Figure 41. Enrollment Confirmation Transmit Packet Format Table

Response Packet Format (Byte):

Checksum Password (4 Byte)				CMD1		CMD2		Completion Code (4 Byte)				Checksum
0x00	0x00	0x00	0x00	0x01	0x41	0x00	0x00	0x00	0x00	0x00	0x00	0xBE

Response Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Error Code				Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	00	00	00	00	01	41	00	00	00	00	BE

Figure 42. Enrollment Confirmation Response Packet Format Table

5.21. Query Enrollment Confirmation Result :

Transmit Packet Format (Byte):

Checksum Password (4 Byte)				CMD1		CMD2		Checksum	
0x00	0x00	0x00	0x00	0x01	0x42	0x00	0x42	0xBD	

Transmit Packet Example:

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	07	86	00	00	00	00	01	42	BD

Figure 43. Query Enrollment Confirmation Transmit Packet Format Table

Response Packet Format (Byte):

Contains three fields: match result, match score, and match ID (reserved):

Checksum Password (4 Byte)				CMD1		CMD2		Completion Code (4 Byte)				Checksum	
0x00	0x00	0x00	0x00	0x01	0x42	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
Match Result		Match Score		ID (Reserved)		Checksum							
0x00	0x01	0x27	0x0F	0x00	0x00	0x86							

Response Packet Example:

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		Error Code				
F1	1F	E2	2E	B6	6B	A8	8A	00	11	7C	00	00	00	00	01	42	00	00	00	00	
Match Result		Match Score		ID(Reserved)		Checksum															
00	01	27	0F	00	00	86															

Figure 44. Query Enrollment Confirmation Response Packet Format Table

5.22. Download Template Info :

Transmit Packet Format (Byte):

ID_H, ID_L are the fingerprint ID to download (assigned from 0); INFO is the total data size to download (max 8192 bytes).

Checksum Password (4 Byte)				CMD1	CMD2	ID_H	ID_L	INFO_H	INFO_L	Checksum
0x00	0x00	0x00	0x00	0x01	0x51	0x00	0x00	0x07	0xEC	0xBB

Transmit Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command	ID_H	ID_L	INF_O_H	INF_O_L	Checksum	
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	00	00	00	00	01	51	00	00	07	EC	BB

Figure 45. Download Template Info Transmit Packet Format Table

Response Packet Format (Byte):

Checksum Password (4 Byte)				CMD1	CMD2	Completion Code (4 Byte)				Checksum
0x00	0x00	0x00	0x00	0x01	0x51	0x00	0x00	0x00	0x00	0xAE

Response Packet Example:

Frame Header								Data (Len)		Header Checksum		Checksum Password				Command		Error Code				Checksum	
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	00	00	00	00	01	51	00	00	00	00	AE		

Figure 46. Download Template Info Response Packet Format Table

5.23. Download Template Data :

Transmit Packet Format (Byte):

FRAME is the current frame number (first frame starts at 0). The DATA length is variable, determined by the data length sent by the user (len-9, max 128 bytes). Download all data specified by INFO.

Checksum Password				CMD1	CMD2	FRAME_H	FRAME_L	DATA1	...	DATAn	Checksum
0x00	0x00	0x00	0x00	0x01	0x52	0xXX	0xXX	0xXX	...	0xXX	0xXX

Transmit Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		FRAME_H	FRAME_L	DATA1	...	DATAn	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	89	04	00	00	00	00	01	52	0xXX	0xXX	0xXX	...	0xXX	0xXX

Figure 47. Download Template Data Transmit Packet Format Table

Response Packet Format (Byte)

Checksum Password (4 Byte)				CMD1	CMD2	Completion Code (4 Byte)				Checksum
0x00	0x00	0x00	0x00	0x01	0x52	0x00	0x00	0x00	0x00	0xAD

Response Packet Example:

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		Error Code				Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	00	00	00	00	01	52	00	00	00	00	AD

Figure 48. Download Template Data Response Packet Format Table

5.24. Upload Template Info :

Transmit Packet Format (Byte):

ID_H, ID_L are the fingerprint ID to upload:

Checksum Password (4 Byte)				CMD1	CMD2	ID_H	ID_L	Checksum
0x00	0x00	0x00	0x00	0x01	0x53	0x00	0x00	0xAC

Transmit Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		ID_H	ID_L	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	09	84	00	00	00	00	01	53	00	00	AC

Figure 49. Upload Template Info Transmit Packet Format Table

Response Packet Format (Byte):

INFO: uploaded data size:

Checksum Password (4 Byte)				CMD1	CMD2	Completion Code (4 Byte)				INFO_H	INFO_L	Checksum
0x00	0x00	0x00	0x00	0x01	0x53	0x00	0x00	0x00	0x00	0x07	0xEC	0xB9

Response Packet Example:

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		Error Code				INFO_H	INFO_L	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0D	80	00	00	00	00	01	53	00	00	00	00	07	EC	B9

Figure 50. Upload Template Info Response Packet Format Table

5.25. Upload Template Data :

Transmit Packet Format (Byte):

NUM_H, NUM_L are the upload count or frame number (first frame starts at 0):

Checksum Password (4 Byte)				CMD1	CMD2	NUM_H	NUM_L	Checksum
0x00	0x00	0x00	0x00	0x01	0x54	0x00	0x00	0xAB

Transmit Packet Example:

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		NUM_H	NUM_L	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	09	84	00	00	00	00	01	54	00	00	AB

Figure 51. Upload Template Data Transmit Packet Format Table

Response Packet Format (Byte):

The host should accumulate the length of each frame of template data and read exactly the INFO length obtained from command 0x53:

Checksum Password (4 Byte)				CMD1	CMD2	Completion Code (4 Byte)				NUM_H	NUM_L	DATA1	...	DATA 128	Checksum
0x00	0x00	0x00	0x00	0x01	0x54	0x00	0x00	0x00	0x00	0x00	0x00	0xXX	...	0xXX	0xXX

Response Packet Example:

Frame Header								Data (Len)	Header Checksum	Checksum Password	Command	Error Code				NUM_H	NUM_L	DATA1	...	DATA 128	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	8D	00	00	00	00	00	00	00	00	XX	...	XX	XX

Figure 52. Upload Template Data Response Packet Format Table

5.26. Set Password Command :

Transmit Packet Format (Byte):

If setting password to 0x12345678, PSW0~PSW3 are 0x12, 0x34, 0x56, 0x78 respectively. Not written to flash; lost on power-down.

Checksum Password				CMD1	CMD2	PSW0
0x00	0x00	0x00	0x00	0x02	0x01	0x12
PSW1	PSW2	PSW3	Checksum			
0x34	0x56	0x78	0xE9			

Transmit Packet Example:

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		PSW0	PSW1	PSW2	PSW3	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	00	00	00	00	02	01	12	34	56	78	E9

Figure 53. Set Password Command Transmit Packet Format Table

Response Packet Format (Byte):

Checksum Password				CMD1	CMD2	Completion Code				Checksum
0x12	0x34	0x56	0x78	0x02	0x01	0x00	0x00	0x00	0x00	0xE9

Response Packet Example:

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		Error Code				Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	12	34	56	78	02	01	00	00	00	00	E9

Figure 54. Set Password Command Response Packet Format Table

Password set successfully. If the Get Module ID command is sent again, the format is as follows.

Checksum Password				CMD1	CMD2	Checksum
0x12	0x34	0x56	0x78	0x03	0x01	0xE8

Transmit Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	07	86	12	34	56	78	03	01	E8

Figure 55. Password Set Success Verification Packet Format Table

5.27. Reset Fingerprint Module :

Transmit Packet Format (Byte):

Checksum Password (4 Byte)				CMD1	CMD2	Checksum
0x00	0x00	0x00	0x00	0x02	0x02	0xFC

Transmit Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	07	86	00	00	00	00	02	02	FC

Figure 49. Reset Module Transmit Packet Format Table

Response Packet Format (Byte):

Checksum Password				CMD1	CMD2	Completion Code				Checksum
0x00	0x00	0x00	0x00	0x02	0x02	0x00	0x00	0x00	0x00	0xFC

Response Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Error Code				Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	00	00	00	00	02	02	00	00	00	00	FC

Figure 56. Reset Module Response Packet Format Table

5.28. Get Fingerprint Template Count:

Transmit Packet Format (Byte):

Checksum Password (4 Byte)				CMD1	CMD2	Checksum
0x00	0x00	0x00	0x00	0x02	0x03	0xFB

Transmit Packet Example:

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	07	86	00	00	00	00	02	03	FB

Figure 57. Get Template Count Transmit Packet Format Table

Response Packet Format (Byte):

Returns the number of enrolled templates:

Checksum Password (4 Byte)				CMD1	CMD2	Completion Code (4 Byte)				NUM_H	NUM_L	Checksum
0x00	0x00	0x00	0x00	0x02	0x03	0x00	0x00	0x00	0x00	0x00	0x04	0xF7

Response Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Error Code				NUM_H	NUM_L	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0D	80	00	00	00	00	02	03	00	00	00	00	00	04	F7

Figure 58. Get Template Count Response Packet Format Table

5.29. Get Gain:

Transmit Packet Format (Byte):

Checksum Password				CMD1	CMD2	Checksum
0x00	0x00	0x00	0x00	0x02	0x09	0xF5

Transmit Packet Example:

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	07	86	00	00	00	00	02	09	F5

Figure 59. Get Gain Transmit Packet Format Table

Response Packet Format (Byte):

Checksum Password				CMD1	CMD2	Completion Code			
0x00	0x00	0x00	0x00	0x02	0x09	0x00	0x00	0x00	0x00
Shift	Gain	PxlCtrl	Checksum						
0x1F	0x00	0x04	0xD2						

Response Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Error Code			
F1	1F	E2	2E	B6	6B	A8	8A	00	0E	7F	00	00	00	00	02	09	00	00	00	00
Shift		Gain		Pxl Ctrl		Checksum														
1F		00		04		D2														

Figure 60. Get Gain Response Packet Format Table

5.30. Get Matching Threshold:

Transmit Packet Format (Byte):

Checksum Password				CMD1	CMD2	Checksum
0x00	0x00	0x00	0x00	0x02	0x0B	0xF3

Transmit Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	07	86	00	00	00	00	02	0B	F3

Figure 61. Get Matching Threshold Transmit Packet Format Table

Response Packet Format (Byte):

Checksum Password				CMD1	CMD2	Completion Code			
0x00	0x00	0x00	0x00	0x02	0x0B	0x00	0x00	0x00	0x00
Threshold_H		Threshold_L		Checksum					
0x21		0x34		0x9E					

Response Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Error Code				Thresh old_H	Thresh old_L	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0D	80	00	00	00	00	02	0B	00	00	00	00	21	34	9E

Figure 62. Get Matching Threshold Response Packet Format Table

5.31. Set Sleep Mode :

Transmit Packet Format (Byte):

MODE field is 1 byte: 0x00 = normal sleep mode, 0x01 = deep sleep mode

Checksum Password				CMD1	CMD2	MODE	Checksum
0x00	0x00	0x00	0x00	0x02	0x0C	0x00	0xF2

Transmit Packet Example:

Normal Sleep:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		MODE	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	08	85	00	00	00	00	02	0C	00	F2

Deep Sleep:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		MODE	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	08	85	00	00	00	00	02	0C	01	F1

Figure 63. Set Sleep Mode Transmit Packet Format Table

Response Packet Format (Byte):

Checksum Password				CMD1	CMD2	Completion Code				Checksum
0x00	0x00	0x00	0x00	0x02	0x0C	0x00	0x00	0x00	0x00	0xF2

Response Packet Example:

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		Error Code				Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	00	00	00	00	02	0C	00	00	00	00	F2

Figure 64. Set Sleep Mode Response Packet Format Table

Note: In normal sleep mode, the module can be woken by pressing the fingerprint sensor; the UART is unresponsive until wake-up. In deep sleep mode, an external reset signal is required to wake the module; the UART is unresponsive.

5.32. Set Enrollment Stitch Count:

Transmit Packet Format (Byte):

ENROLL_NUM range (1 - 6):

Checksum Password				CMD1	CMD2	ENROLL_NUM	Checksum
0x00	0x00	0x00	0x00	0x02	0x0D	0x06	0xEB

Transmit Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		ENROLL_NUM	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	08	85	00	00	00	00	02	0D	06	EB

Figure 65. Set Enrollment Stitch Count Transmit Packet Format Table

Response Packet Format (Byte):

Checksum Password				CMD1	CMD2	Completion Code				Checksum
0x00	0x00	0x00	0x00	0x02	0x0D	0x00	0x00	0x00	0x00	0xF1

Response Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Error Code				Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	00	00	00	00	02	0D	00	00	00	00	F1

Figure 66. Set Enrollment Stitch Count Response Packet Format Table

5.33. Set LED Control Info:

Transmit Packet Format (Byte):

Checksum Password				CMD1	CMD2	Control Mode
0x00	0x00	0x00	0x00	0x02	0x0F	0x03
LED Color	Param 1	Param 2	Param 3	Checksum		
0x03	0x64	0x00	0x32	0x53		

Parameter Description:

Parameter	Control Mode	LED Color		Param 1	Param 2	Param 3
Parameter Description	0: LED off 1: LED on 2: Auto-on when finger touches	0: No color control 1: Green 2: Red 3: Red + Green	PWM Mode	Max duty cycle (range: 0 - 100)	Min duty cycle (range: 0 - 100)	Duty cycle change rate (unit: 1%/s, range: 0 - 100)
	3: PWM LED (breathing) 4: Blinking LED	4: Blue 5: Red + Blue 6: Green + Blue 7: Red+Blue+Green	Blink Mode	LED on duration (unit: 10 ms)	LED off duration (unit: 10 ms)	Number of blink cycles

Transmit Example:

LED Off:

Set to breathing mode (red+green on simultaneously, max duty 100, min duty 0, duty change rate 50%/s):

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Control Mode	LED Color	Param 1	Param 2	Param 3	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0C	81	00	00	00	00	02	0F	03	03	64	00	32	53

Set to blinking mode (green blink, on 200 ms, off 200 ms, 5 cycles):

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Control Mode	LED Color	Param 1	Param 2	Param 3	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0C	81	00	00	00	00	02	0F	04	01	14	14	05	BD

Figure 67. Set LED Control Info Transmit Packet Format Table

Response Packet Format (Byte):

Checksum Password				CMD1	CMD2	Completion Code				Checksum
0x00	0x00	0x00	0x00	0x02	0x0F	0x00	0x00	0x00	0x00	0xEF

Response Packet Example:

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		Error Code				Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	00	00	00	00	02	0F	00	00	00	00	EF

Figure 68. Set LED Control Info Response Packet Format Table

5.34. Get System Policy:

Transmit Packet Format (4 Byte):

Checksum Password				CMD1	CMD2	Completion Code				Checksum
0x00	0x00	0x00	0x00	0x02	0xFB	0x00	0x00	0x00	0x00	0x03

Transmit Packet Example:

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	07	86	00	00	00	00	02	FB	03

Figure 69. Get System Policy Transmit Format Table

Response Packet Format (Byte):

Bit0: Internal use. Bit1: Duplicate fingerprint check. Bit2: Self-learning. Bit3: Internal use. Bit4: 360-degree recognition.

Bit5: Internal use. Others reserved.

Checksum Password				CMD1	CMD2	Policy0
0x00	0x00	0x00	0x00	0x02	0xFB	0x00
Policy1	Policy2	Policy3	Checksum			
0x00	0x00	0x16	0xEC			

Response Example: policy 0x00000016 = duplicate check, self-learning, 360-degree

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Error Code				Policy0	Policy1	Policy2	Policy3	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0F	7E	00	00	00	00	02	FB	00	00	00	00	00	00	00	16	ED

Figure 70. Get System Policy Response Format Table

5.35. Set System Policy:

Transmit Packet Format (4 Byte):

Bit0: Internal use. Bit1: Duplicate fingerprint check. Bit2: Self-learning. Bit3: Internal use. Bit4: 360-degree recognition.

Bit5: Internal use. Others reserved.

Checksum Password				CMD1	CMD2	Policy0
0x00	0x00	0x00	0x00	0x02	0xFC	0x00
Policy1	Policy2	Policy3	Checksum			
0x00	0x00	0x16	0xEC			

Transmit Example: policy 0x00000016 = duplicate check, self-learning, 360-degree

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		Policy0	Policy1	Policy2	Policy3	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	00	00	00	00	02	FC	00	00	00	16	EC

Figure 71. Set System Policy Transmit Format Table

Response Packet Format (Byte):

Checksum Password				CMD1	CMD2	Completion Code				Checksum
0x00	0x00	0x00	0x00	0x02	0xFC	0x00	0x00	0x00	0x00	0x02

Response Packet Example:

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		Error Code				Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	00	00	00	00	02	FC	00	00	00	00	02

Figure 72. Set System Policy Response Format Table

5.36. Get Fingerprint Module ID:

Transmit Packet Format (Byte):

Checksum Password				CMD1	CMD2	Checksum	
0x00	0x00	0x00	0x00	0x03	0x01	0xFC	

Transmit Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	07	86	00	00	00	00	03	01	FC

Figure 73. Get Module ID Transmit Packet Format Table

Response Packet Format (Byte):

Return field length is 16.

Checksum Password				CMD1	CMD2	Completion Code				ID0
0x00	0x00	0x00	0x00	0x03	0x01	0x00	0x00	0x00	0x00	0x4D
ID1	ID2	ID3	ID4	ID5	ID6	ID7	ID8	ID9	ID10	ID11
0x4C	0x2D	0x46	0x50	0x4D	0x30	0x30	0x31	0x2D	0x30	0x31
ID12	ID13	ID14	ID15	Checksum						
0x2D	0x31	0x30	0x31	0x75						

Response Packet Example:

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		Error Code				ID0
F1	1F	E2	2E	B6	6B	A8	8A	00	1B	72	00	00	00	00	03	01	00	00	00	00	4D
ID1	ID2	ID3	ID4	ID5	ID6	ID7	ID8	ID9	ID10	ID11	ID12	ID13	ID14	ID15	Checksum						
4C	2D	46	50	4D	30	30	31	2D	30	31	2D	31	30	31	75						

Figure 74. Get Module ID Response Packet Format Table

5.37. Heartbeat Command:

Transmit Packet Format (Byte):

Generally used to test communication with the module.

Checksum Password				CMD1	CMD2	Checksum
0x00	0x00	0x00	0x00	0x03	0x03	0xFA

Transmit Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	07	86	00	00	00	00	03	03	FA

Figure 75. Heartbeat Command Transmit Packet Format Table

Response Packet Format (Byte):

Checksum Password				CMD1	CMD2	Completion Code				Checksum
0x00	0x00	0x00	0x00	0x03	0x03	0x00	0x00	0x00	0x00	0xFA

Response Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Error Code				Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	00	00	00	00	03	03	00	00	00	00	FA

Figure 76. Heartbeat Command Response Packet Format Table

5.38. Set Baud Rate:

Transmit Packet Format (4 Byte):

If setting baud rate to 9600, Baud0~Baud3 are 0x00002580.

Checksum Password				CMD1	CMD2	Baud0
0x00	0x00	0x00	0x00	0x03	0x04	0x00
Baud1	Baud2	Baud3	Checksum			
0x00	0x25	0x80	0x54			

Transmit Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Baud 0	Baud 1	Baud 2	Baud 3	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	00	00	00	00	03	04	00	00	25	80	54

Baud rate 115200:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Baud 0	Baud 1	Baud 2	Baud 3	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	00	00	00	00	03	04	00	01	C2	00	36

Figure 77. Set Baud Rate Transmit Packet Format Table

Response Packet Format (Byte):

Checksum Password				CMD1	CMD2	Completion Code				Checksum
0x00	0x00	0x00	0x00	0x03	0x03	0x00	0x00	0x00	0x00	0xFA

Response Packet Example:

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		Error Code				Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	00	00	00	00	03	04	00	00	00	00	F9

Figure 78. Set Baud Rate Response Packet Format Table

5.39. Set Communication Password :

Transmit Packet Format (Byte):

If setting password to 0x12345678, PSW0~PSW3 are 0x12, 0x34, 0x56, 0x78 (written to flash)

Checksum Password				CMD1	CMD2	PSW0
0x00	0x00	0x00	0x00	0x03	0x05	0x12
PSW1	PSW2	PSW3	Checksum			
0x34	0x56	0x78	0xE4			

Transmit Packet Example:

Set password to: 0x12345678

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		PSW 0	PSW 1	PSW 2	PSW 3	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	00	00	00	00	03	05	12	34	56	78	E4

Restore default password 0x00000000:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		PSW 0	PSW 1	PSW 2	PSW 3	Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	12	34	56	78	03	05	00	00	00	00	E4

Figure 79. Set Communication Password Transmit Packet Format Table

Response Packet Format (Byte):

Checksum Password				CMD1	CMD2	Completion Code				Checksum
0x12	0x34	0x56	0x78	0x03	0x05	0x00	0x00	0x00	0x00	0xE4

Response Packet Example:

Password: 0x12345678

Frame Header								Data (Len)		Frame Header Check	Checksum Password				Command		Error Code				Check sum
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	12	34	56	78	03	05	00	00	00	00	E4

Password: 0x00000000

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Error Code				Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	0B	82	00	00	00	00	03	05	00	00	00	00	F8

Figure 80. Set Communication Password Response Packet Format Table

If the Get Module ID command is sent again, the format is as follows.

Checksum Password				CMD1	CMD2	Checksum
0x78	0x56	0x34	0x12	0x03	0x01	0xE8

Transmit Packet Example:

Frame Header								Data (Len)		Header Checksum	Checksum Password				Command		Error Code				Checksum
F1	1F	E2	2E	B6	6B	A8	8A	00	07	86	12	34	56	78	03	01					E8

Figure 81. Password Set Success Verification Packet Format Table

6. Appendix

6.1. Error Code Definition Table:

No.	Error Code	Description
1	0x00000000	Completed normally
2	0x00000001	Unrecognized command
3	0x00000002	Invalid command data length
4	0x00000003	Invalid command field data
5	0x00000004	System busy, cannot execute current command
6	0x00000005	Result queried without sending the corresponding request
7	0x00000006	System software error
8	0x00000007	Hardware error
9	0x00000008	No finger press detected, timed out
10	0x00000009	Fingerprint extraction error
11	0x0000000A	Fingerprint matching error (template library empty)
12	0x0000000B	Fingerprint data storage full
13	0x0000000C	Storage write failed
14	0x0000000D	Storage read failed
15	0x0000000E	Captured fingerprint image quality is poor
16	0x0000000F	Duplicate fingerprint
17	0x00000010	Capture area too small, finger-sensor contact area too small
18	0x00000011	Finger movement too large during capture
19	0x00000012	Finger movement too small during capture
20	0x00000013	Fingerprint ID already in use
21	0x00000014	Module image capture failed
22	0x00000015	Command forcibly interrupted
23	0x00000016	Template data does not need updating
24	0x00000017	Invalid fingerprint ID
25	0x00000018	Gain adjustment failed
26	0x00000019	Data buffer overflow
27	0x0000001A	Error returned when capture-related message received while sensor is sleeping
28	0x0000001C	Checksum error
29	0x00000022	Flash write failed during enrollment storage
30	0x000000FF	Other error

Figure 82 Error Code Definition Table

6.2. Power-on Operation Flow:

As shown below, after power-on the module executes the following flow. The host should wait approximately 100 ms after power-on or reset for the module to complete initialization.

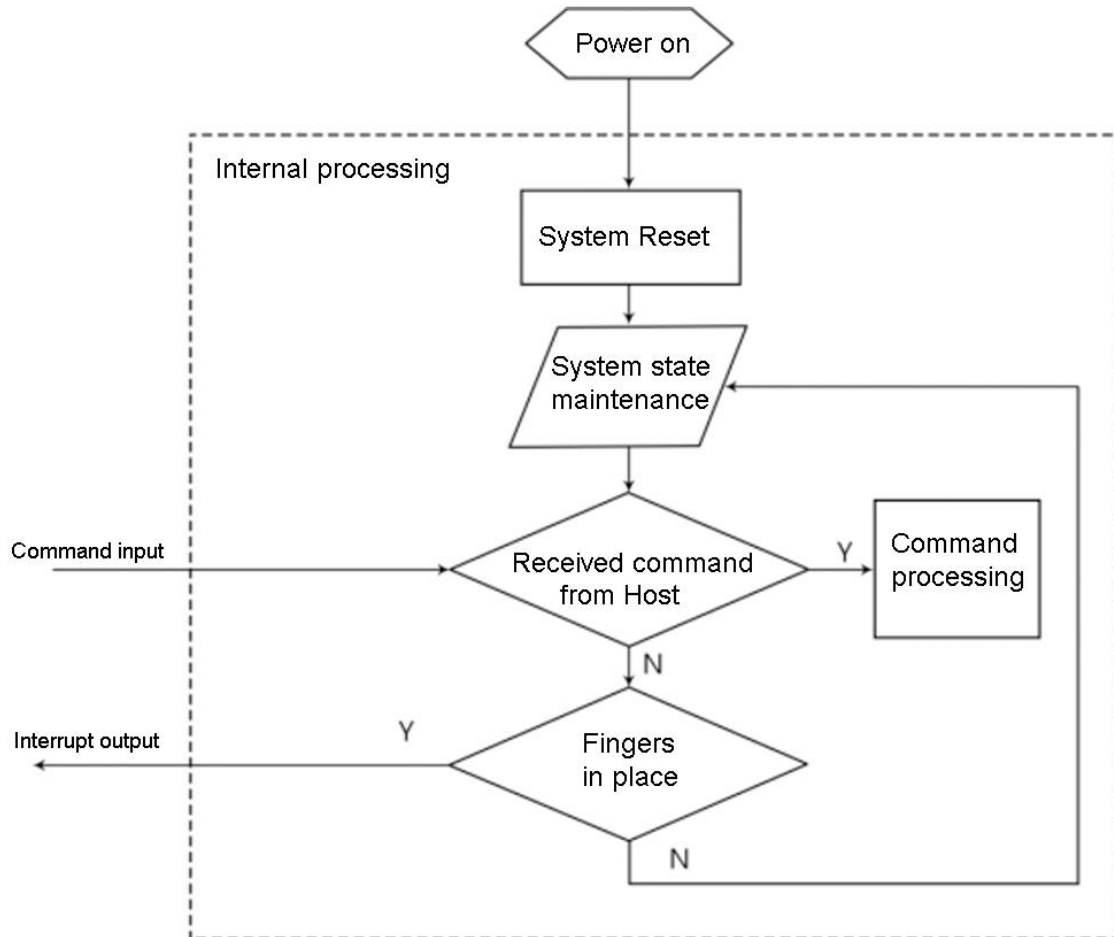


Figure 83. Power-on Operation Flowchart

Note:

A finger pressing the sensor during initialization does not affect it. After initialization completes, the module immediately outputs a finger-present interrupt signal.

6.3. Enrollment Flow:

The enrollment flow adds user fingerprint template information for subsequent verification. The enrollment process is shown below:

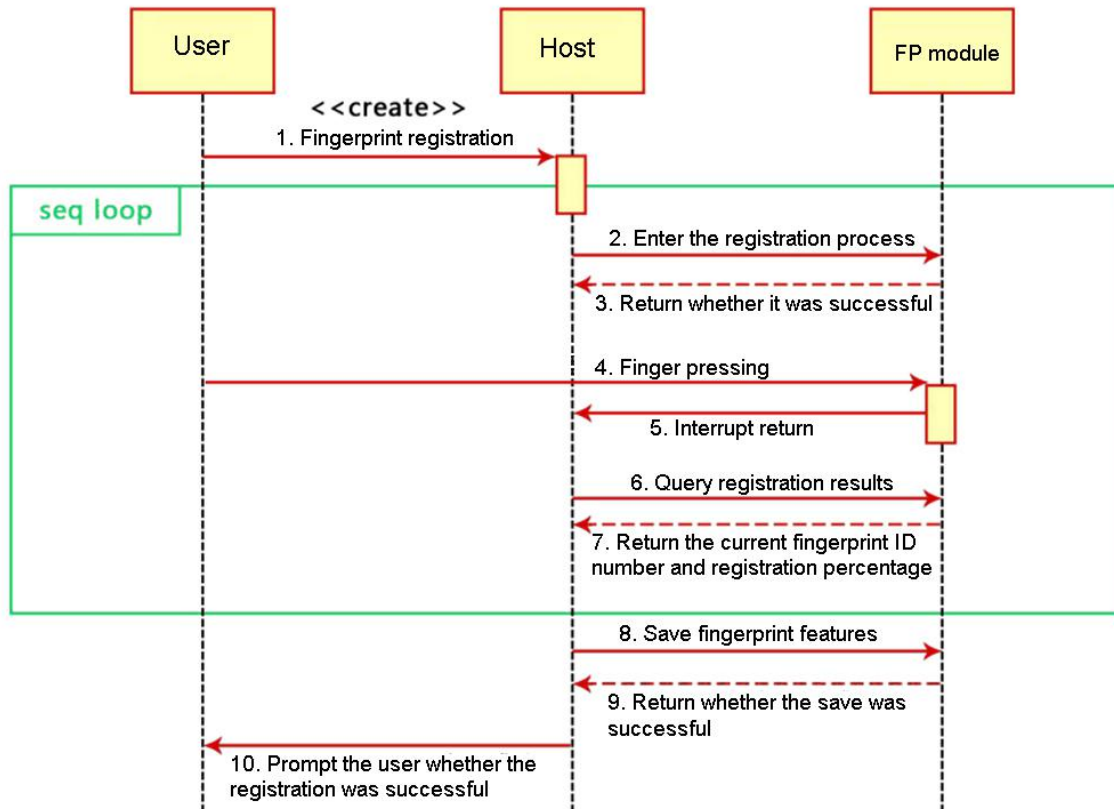


Figure 84. Enrollment Flowchart

The user first initiates enrollment on the host. The host then sends the enter-enrollment command to the module, which responds and enters enrollment mode. If no query-enrollment-result command is sent within 10 s after entering enrollment, the enter-enrollment command must be resent. The interval between consecutive query commands must also not exceed 10 s, otherwise the enter-enrollment command must be resent.

Enrollment Flow Description:

- ① Send enrollment command (0x01 0x11). (Recommendation: REG_IDX starts at 0x01, increments to 0x06.)
- ② Query enrollment result (0x01 0x12). If completion code is 0x00000000, the current enrollment step is complete. At this point check whether PROC has increased — if so, this step succeeded. Then check PROC: if PROC < 100, go to step ③; if PROC ≥ 100, the entire enrollment flow is successfully complete and go to step ④. If completion code is 0x00000004, the sensor has not finished processing fingerprint data — wait briefly (recommended 200 ms) and query again. For other errors (e.g., poor image quality, pressure too heavy/light, finger too moist), continue enrolling and go to step ①.
- ③ Query finger presence (0x01 0x35). If completion code is not 0x00000000 or STATE = 1, a query error occurred or the finger is still on the sensor, prompt the user to lift and re-press, wait (recommended 200 ms), and query again. When completion code is 0x00000000 and STATE = 0, go to step ①.
- ④ Save fingerprint (0x01 0x13).
- ⑤ Query save result (0x01 0x14). If completion code is 0x00000000, save is complete. If 0x00000004, save is not complete, wait briefly (recommended 200 ms) and query again. If 0x0000000B, flash storage may be full. Refer to the error code table for specific causes.

Recommendation: enrollment count varies by customer requirement. For uniform shipping, it is recommended to send the set-enrollment-count command (0x02 0x0D) before each enrollment to set the desired count.

6.4. Verification Flow:

The verification flow authenticates the user by confirming whether the fingerprint has been enrolled. The interval between entering matching mode and querying the result must not exceed 10 s, otherwise the command must be resent.

Verification Flow:

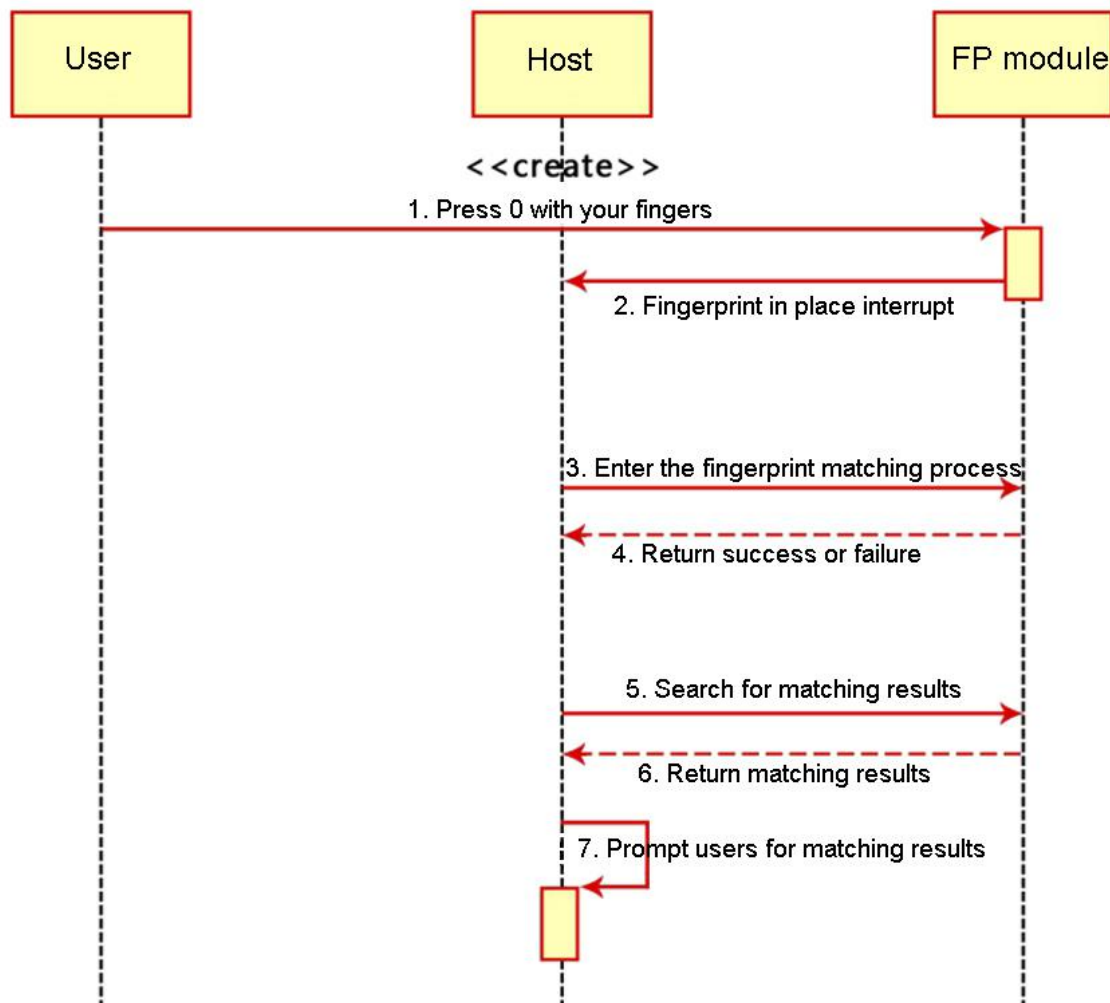


Figure 85. Verification Flowchart

Verification Flow Description:

Send matching command (0x01 0x21). Prompt the user to press a finger before sending.

Query matching result (0x01 0x22). If completion code is 0x00000000, matching is complete. At this point view the matchresult, match score, and matched fingerprint ID. If match result is 1, a fingerprint was matched; otherwise no match. If completion code is 0x00000004, matching is not complete — wait briefly (recommended 200 ms) and query again. For other errors (e.g., poor image quality 0x0000000A, pressure too heavy/light, finger too moist), re-match and go to step ①.

6.5. Deletion Flow:

The deletion flow removes unneeded fingerprints. Deletion flow:

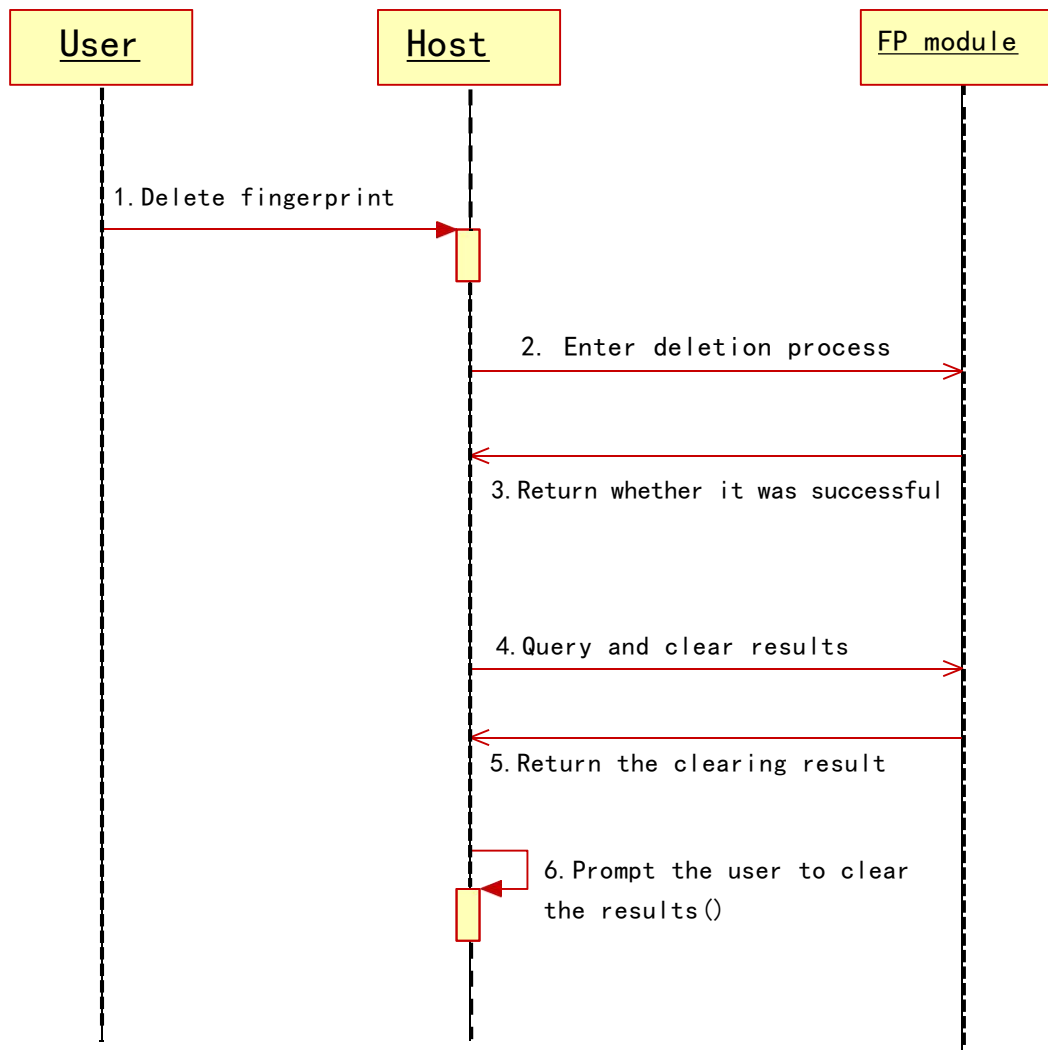


Figure 86. Deletion Flowchart

Deletion Flow Description:

Send clear-template command (0x01 0x31). CL_ALL flag: 0x01 = clear all fingerprints, default 0x00 = clear a single fingerprint. ID_H, ID_L are the ID to delete. Users can clear all, single, or multiple fingerprints as needed (see the clear-template transmit examples for formats). Query clear result (0x01 0x32). If completion code is 0x00000000, deletion succeeded. If 0x00000004, deletion is not complete — wait briefly (recommended 200 ms) and query again. For other errors (e.g., fingerprint ID does not exist, flash hardware error), refer to the error code table and resend the correct clear command.

6.6. Checksum Function Code Example:

```

Header checksum = ~(frame header + app layer data
length checksum) + 1
Transmit checksum = ~(checksum password + command +
data content) + 1
Receive checksum = ~(checksum password + response
command + error code + data content) + 1

static U8Bit FP_protocol_get_checksum(U8Bit * data, U32Bit length)
{
    U32Bit i = 0;
    S8Bit sum = 0;

    for(i = 0; i < length;
        i++) { sum += data[i];
    }

    return (U8Bit)((~sum)+1);
}

```

6.7. Reset Function (Single-Supply Module):

Purpose: wake-up function (reset can wake the module without touching the sensor). Operation: pull the RESET pin low for 10 ms, then release it high.

6.8. Self-Learning Function:

Self-learning updates a small portion of unenrolled template features after a successful match, thereby improving match success rate.

Flow Description:

Send matching command (0x01 0x21). Press a finger.

Query matching result (0x01 0x22). If completion code is 0x00000000 and match result is 0x0001, matching succeeded go to step ③. Otherwise it failed and no template update is needed.

Update stored template (0x01 0x16). Pass the matched ID.

Query update result (0x01 0x17). Check whether the update succeeded or was unnecessary.

7. Revision History

Version	Revisions	Date	Modified by
V1.0.1	1.First draft.	2016-01-13	Roy
V1.0.2	1.Modify Command	2016-04-26	Roy
V1.0.3	1.Modify Query Fingerprint ID	2016-04-29	Roy
V1.0.5	Fix bug	2016-05-11	Roy/Jack
V1.0.6	Add Low Power Model	2016-06-02	Jack/lidong
V1.0.7	optimize	2017-03-20	lidong/Alex
V1.0.8	optimize	2017-07-03	lidong/Alex
V1.0.9	Add Command/optimize	2017-09-01	Alex
V1.1.0	Add Command	2017-11-27	lidong
V1.1.1	Add Command/optimize	2018-01-10	Alex
V1.1.2	Add Command/optimize	2018-03-26	Alex
V1.1.3	optimize	2018-04-25	Alex
V1.1.5	Add Command/optimize	2018-06-21	Alex
V1.1.6	Add Command	2018-08-20	Lance
V1.1.7	Add Command	2018-10-08	Lance
V1.1.8	Optimize	2018-11-16	Lance
V1.1.9	Fix bug	2019-12-25	Phager
V1.1.10	1.Definitions of hardware interface removed (please be referred to the module SPEC for them). 2. Layout optimized.	2020-09-11	lidong