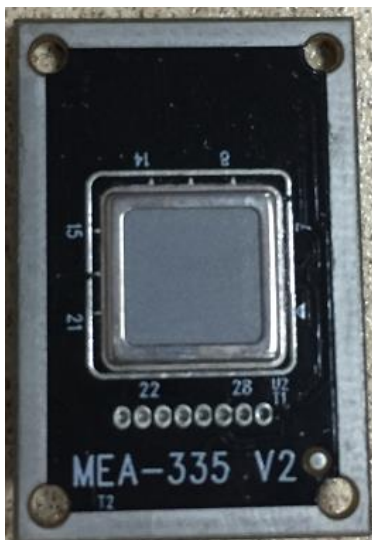


MEA-335/MEA-336/MEA536

Fingerprint Identification Module

User's Guide



MEA-335 Top View



MEA-536 Top View



MEA-336 Top view



Bottom View

BRK co.,ltd

2016/9

CONTENTS

1	SUMMARY	4
2	COMMUNICATION PROTOCOL	5
2.1	THE PROCESS OF COMMUNICATION	5
2.2	CLASSIFY OF COMMUNICATION PACKET	6
2.2.1	Command packet	6
2.2.2	Response packet	6
2.3	PACKET STRUCTURE.....	7
2.3.1	Packet Identify Code	7
2.3.2	Structure of Command packet	7
2.3.3	Structure of Response packet	7
2.4	COMMAND LIST.....	9
2.5	COMMAND PACKET DESCRIPTION	9
2.5.1	CMD_TEST_CONNECTION	9
2.5.2	Set parameter to device (CMD_SET_PARAM)	10
2.5.3	Get device parameter (CMD_GET_PARAM)	13
2.5.4	Capture fingerprint image to ImageBuffer (CMD_GET_IMAGE).....	14
2.5.5	Save template from Ram Buffer to DB in flash memory (CMD_STORE_CHAR)	15
2.5.6	Read an template from DB in FLASH to Ram Buffer (CMD_LOAD_CHAR).....	16
2.5.7	Delete all template that appointed ID range (CMD_DEL_CHAR).....	17
2.5.8	Get first valid ID for enroll in appointed range (CMD_GET_EMPTY_ID).....	18
2.5.9	Get an appointed ID No. and check whether enrolled (CMD_GET_STATUS).....	19
2.5.10	Get the total count of have been enrolled (CMD_GET_ENROLL_COUNT).....	20
2.5.11	Creat template data from ImageBuffer to RamBuffer (CMD_GENERATE).....	21
2.5.12	Merge 2 or 3 templates in RamBuffer to one template (CMD_MERGE).....	22
2.5.13	Verify 2 Templates in appointed RamBuffer (CMD_MATCH)	23
2.5.14	Identify (1:N) an RamBffer with an range in DB of device (CMD_SEARCH).....	24
2.5.15	Verify an RamBuffer with an appointed ID in DB (CMD_VERIFY).....	25
2.5.16	Communication error (Incorrect Command)	26
2.6	NOTE	26
3	APPENDIX	27
3.1	RESPONSE AND ERROR CODE	27
3.2	ENROLL AND MATCH FLOW CHART	27
3.2.1	Enroll	28
3.2.2	Verify & Identify	29

4	HARDWARE	30
5.1	DIMENSION OF BOARD.....	30
4.2	CONNECTOR	31
4.3	TECHNOLOGY PARAMETERS.....	33

1 SUMMERY

MEA-335/MEA-336/MEA-536 as single chip fingerprint identity OEM modules, with the advantages of small volume, low power consumption, simple interface, high reliability, small fingerprint template (496 bytes), large capacity fingerprint recognition (3000 fingerprints recognition and response time is less than 1.5 second) etc, can be very convenient used for the your application system to meet the requirement for fingerprint identification products.

Especially with auto-learn function, during the process of fingerprint identification, it collect and save the newest fingerprint feature data to fingerprint database, make identification better and better in the future use.

MEA-335/MEA-336/MEA-536 communication interface is UART or USB, it is work as a slave device, the host device will send commands to control it.

The module MEA-335/MEA-336/MEA-536 has the adjustable security level function, fingerprint feature data read / write functions and fingerprint image read / write function, Match methods is 1:N or 1:1.

Main functions as following :

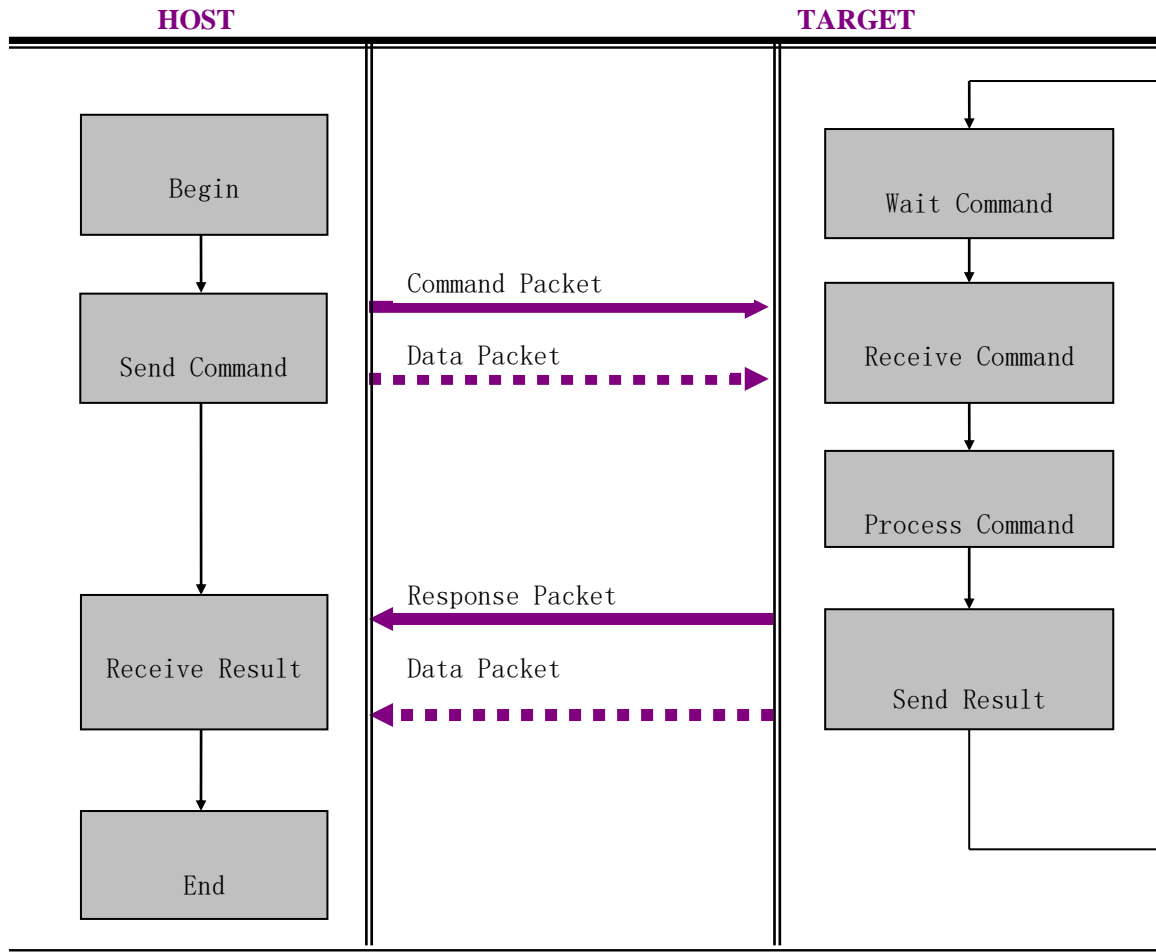
- Big fingerprint capacity: 3000 fingerprints (identification time is less than 1.5 second)
- small fingerprint template (496 bytes)
- Identify mode: 1 : N identification or 1 : 1 verification
- Auto Learn function
- fingerprint feature data read / write functions
- fingerprint image read / write function
- adjustable security level function
- communication interface is UART or USB
- Support OS: Windows or Android

APPLICATION :

- Fingerprint access control
- Fingerprint time attendance
- Fingerprint lock, fingerprint safe box
- Fingerprint POS terminal (support Android OS)

2 COMMUNICATION PROTOCOL

2.1 The process of communication



Picture 2-1 process of communication

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

2.2 Classify of communication packet

2.2.1 Command packet

- Command Packet is the instruction from Host to Target ;
- Total length of Command packet is **26 bytes**;

2.2.2 Response packet

- Response packet is result of execute command packet;
- from Target to Host,;
- Response packet length is **26 byte**;

2.3 Packet Structure

2.3.1 Packet Identify Code

Section of start 2byte prefix define type of packet

Type of Packet	Code
Command packet	0x55AA
Response packet	0xAA55

2.3.2 Structure of Command packet

PREFIX		SID	DID	CMD		LEN		DATA				CKS	
0x55	0xAA			L	H	L	H	D0	D1	...	D15	L	H
0	1	2	3	4	5	6	7	8	9	...	23	24	25

OFFSET	FIELD	TYPE	SIZE	DESCRIPTION
0	PREFIX	WORD	2byte	Packet Identify code
2	SID	BYTE	1byte	Soruce Device ID
3	DID	BYTE	1byte	Destination Device ID
4	CMD	WORD	2byte	Command Code
6	LEN	WORD	2byte (=n, n < 16)	Length of DATA
8	DATA	Byte Array	16byte	Command Parameter
24	CKS	WORD	2byte	Check Sum is the low word of Arithmetic Sum (From PREFIX to DATA), Sum=offset[0] + offset[1] + ...+offset[23]

Table 2-2 Structure of Command packet

2.3.3 Structure of Response packet

PREFIX	SID	DID	RCM	LEN	RET	DATA	CKS
--------	-----	-----	-----	-----	-----	------	-----

0xAA	0x55			L	H	L	H	L	H	D0	D1	...	D15	L	H
0	1	2	3	4	5	6	7	8	9	10	11	...	23	24	25

OFFSET	FIELD	TYPE	SIZE	DESCRIPTION
0	PREFIX	WORD	2byte	Packet Identify code
2	SID	BYTE	1byte	Source Device ID
3	DID	BYTE	1byte	Destination Device ID
4	RCM	WORD	2byte	Response Code
6	LEN	WORD	2byte(=n, n < 16)	Length of RET and DATA
8	RET	WORD	2byte	Result Code(0 :success , 1 :fault)
10	DATA	Byte Array	14byte	Response Data
24	CKS	WORD	2byte	Check Sum is the low word of Arithmetic Sum (From PREFIX to DATA), Sum=offset[0] + offset[1] + ...+offset[23]

Table 2-3 structure Response packet

Before send Command Data Packet, Host first send Command packet which set the length of next command data packet in Data Field, the length of Command Data Packet which to be transmitted is required.

2.4 Command List

No	Command Name	Code	Function
1	CMD_TEST_CONNECTION	0x0001	For test the communication to device
2	CMD_SET_PARAM	0x0002	Set parameter (Device ID, Security Level, BPS, Duplication Check ,Auto Learn)
3	CMD_GET_PARAM	0x0003	Get parameter (Device ID, Security Level, BPS, Duplication Check, Auto Learn)
4	CMD_GET_DEVICE_INFO	0x0004	Get device information
5	CMD_GET_IMAGE	0x0020	Capture the fingerprint image and save to ImageBuffer
6	CMD_STORE_CHAR	0x0040	Save the template data that storage in Ram Buffer into Flash Memory
7	CMD_LOAD_CHAR	0x0041	Save the template data that storage in flash memory into Ram Buffer
8	CMD_DEL_CHAR	0x0044	Appointed an range in DB and then delete all template data in the range
9	CMD_GET_EMPTY_ID	0x0045	appointed an range and get the first ID number which is unappropriated in the range
10	CMD_GET_STATUS	0x0046	Get the appointed ID number whether is available
11	CMD_GET_ENROLL_COUNT	0x0048	Get the count of enrolled template in appointed range
12	CMD_GENERATE	0x0060	Generate template data from fingerprint image that saved in ImageBuffer and then save to appointed Ram Buffer
13	CMD_MERGE	0x0061	Compose two or three template that saved in Ram Buffer and then generate ultimate template data
14	CMD_MATCH	0x0062	1:1 match, match between two templates that appointed in Ram Buffer
15	CMD_SEARCH	0x0063	1:N identify ,an template in Ram Buffer match to all template that appointed range in DB
16	CMD_VERIFY	0x0064	1:1 match, an template in Ram Buffer match with an template in DB

2.5 Command Packet Description

ImageBuffer and Ram Buffer used for command communication.

ImageBuffer used for save fingerprint image.

Ram Buffer used for save fingerprint feature data.

Total three Ram Buffer: Ram Buffer0, Ram Buffer1 and Ram Buffer2 .

The data in ImageBuffer and Ram Buffer will lost when the module is power down.

2.5.1 CMD_TEST_CONNECTION

- [Function]

Check the connect state for Target to Host .

Host send the command first connect to Target .

Failed is for connect: working abnormal or baud rate setting wrong .

- [Sequence]

If connection is successful, target response is ERR_SUCCESS

- [Command and Response]

PREFIX	0x55AA
SID	Source Device ID
DID	Destination Device ID
CMD	0x0001
LEN	0
DATA	null
PREFIX	0xAA55
SID	Source Device ID
DID	Destination Device ID
RCM	0x0001
LEN	2
RET	Result Code
DATA	null

Table 3-1 CMD_TEST_CONNECTION command

e.g. Test Connect

HOST command: 55 AA 00 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01

Target response: AA 55 01 00 01 00 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 03 01

2.5.2 Set parameter to device (CMD_SET_PARAM)

- [Function]

According to appointed Parameter Type, setting device parameters (Device ID, Security Level, Baudrate, Duplication Check, Auto Learn) response result.

- [Sequence]

1. If the Parameter Type is invalid, return ERR_INVALID_PARAM

2. Else if the Parameter Value is invalid, return ERR_INVALID_PARAM
3. Else according to Parameter Type, setting Parameter Value and return result.

● [Command and Response]

PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0002	
LEN	5	
DATA	1bytes	Parameter Type
	4bytes	Parameter Value
PREFIX	0xAA55	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0002	
LEN	2	
RET	Result Code	
DATA	null	

Table 3-2 CMD_SET_PARAM command

● [Parameter Type]

Parameter Type	Description
0	Device ID: setting range 1 ~ 255
1	Security Level: setting range 1 ~ 5 , Default setting 3 Security Level :

	Security Level	Recognition rate	
	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%
2	Duplication Check: can setting 0 or 1 (NO /YES) Setting to 1 : CMD_STORE_CHAR will duplication check . Setting to 0 : will not duplication check.		
3	Baudrate: setting range 1 ~ 8 (1:9600bps; 2:19200bps; 3:38400bps; 4:57600bps; 5:115200bps; 6:230400bps; 7:460800bps, 8:921600bps		
4	Auto Learn: can setting 0 or 1 (NO /YES) Setting to 1: CMD_SEARCH, CMD_VERIFY will update fingerprint template data Setting to 0: not update template data		

e.g. setting baud rate to 921600BPS

HOST command: 55 AA 00 00 02 00 05 00 03 08 00 00 00 00 00 00 00 00 00 00 11 01
 Target response: AA 55 01 00 02 00 02 00 00 00 00 00 00 00 00 00 00 00 00 00 04 01

2.5.3 Get device parameter (CMD_GET_PARAM)

- **[Function]**

According to appointed Parameter Type, get device parameters as following:
Device ID, Security Level, Baudrate, Duplication Check, Auto Learn.
For Parameter Type, please refer to CMD_SET_PARAM .

- **[Sequence]**

1. If command Parameter Type invalid , return ERR_INVALID_PARAM .
2. Else return command Parameter Type reference device parameters.

- **[Command and Response]**

PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0003	
LEN	1	
DATA	1byte	Parameter Type
PREFIX	0xAA55	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0003	
LEN	success : 6, failed : 2	
RET	Result Code	
DATA	4bytes	success: Parameter Value

Table 3-3 CMD_GET_PARAM command

e.g. read the security level

Host command: 55 AA 00 00 03 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 04 01
Target response: AA 55 01 00 03 00 06 00 00 00 03 00 00 00 00 00 00 00 00 00 0C 01

2.5.4 Capture fingerprint image to ImageBuffer (CMD_GET_IMAGE)

- **[Function]**

Scan the fingerprint image and save to ImageBuffer .

- [Sequence]

scan the fingerprint image.

If scan image success , return ERR_SUCCESS

Else return error code.

- **[Command and Response]**

PREFIX	0x55AA
SID	Source Device ID
DID	Destination Device ID
CMD	0x0020
LEN	0
DATA	null

PREFIX	0xAA55
SID	Source Device ID
DID	Destination Device ID
RCM	0x0020
LEN	2
RET	Result Code
DATA	0

Table 3-6 CMD_IMAGE command

e.g.2 after send CMD_GET_IMAGE, target captured fingerprint image and the response

```
Host command: 55 AA 00 00 20 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 1F 01
```

Target response: AA 55 01 00 20 00 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 22 01

2.5.5 Save template from Ram Buffer to DB in flash memory (CMD_STORE_CHAR)

- [Function]

Save the appointed template in Ram Buffer to DB that appointed ID number

- [Sequence]

1. If ID number is invalid, return ERR_INVALID_TMPL_NO
2. If Ram Buffer number is invalid, return ERR_INVALID_BUFFER_ID
3. If Duplication Check set to OFF, will save the template to DB and return result.
4. If Duplication Check set to ON, will proceed 1:N match between the template and DB,
Match success: Result code=ERR_DUPLICATION_ID and DATA=ID number in DB.
No match: Result code=ERR_SUCCESS and DATA=0

5. [Command and Response]

PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0040	
LEN	4	
DATA	2bytes	Template ID
	2bytes	Ram Buffer ID (0 or 1 or 2)
PREFIX	0XAA55	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0040	
LEN	If result Code is ERR_DUPLICATION_ID, then LEN=4,else LEN= 2	
RET	Result Code	
DATA	2bytes	If result Code=ERR_DUPLICATION_ID, DATA=ID number If result Code=ERR_SUCCESS,DATA=0

Table 3-11 CMD_STORE_CHAR command

E.g. : Save template data in RamBuffer0 to DB which ID=1

Host command: 55 AA 00 00 40 00 04 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 44 01

Target response: AA 55 01 00 40 00 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 42 01

2.5.6 Read an template from DB in FLASH to Ram Buffer (CMD_LOAD_CHAR)

- **[Function]**

Read appointed ID Template from DB and save to appointed Ram Buffer .

- **[Sequence]**

- If appointed Template ID is invalid, return ERR_INVALID_TMPL_NO
- Else if appointed Template ID without enroll data, return ERR_TMPL_EMPTY
- Else if appointed Ram Buffer ID is invalid, return ERR_INVALID_BUFFER_ID
- Else save the template data to appointed Ram Buffer and return ERR_SUCCESS

- **[Command and Response]**

PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0041	
LEN	4	
DATA	2bytes	Template ID
	2bytes	Ram Buffer ID

PREFIX	0Xaa55	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0041	
LEN	2	
RET	Result Code	
DATA	0	

Table 3-12 CMD_LOAD_CHAR command

E.g. : read the template data that ID=1 and save to RamBuffer0

Host command: 55 AA 00 00 41 00 04 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 45 01

Target response: AA 55 01 00 41 00 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 43 01

2.5.7 Delete all template that appointed ID range (CMD_DEL_CHAR)

- **[Function]**

Delete all template that appointed range (from start Template ID to end Template ID).

- **[Sequence]**

- If appointed range is invalid, return `ERR_INVALID_PARAM`.
- Else if in the appointed range there is no any enroll template, return `ERR_TMPL_EMPTY`.
- Else delete all enrolled template in the appointed range and return result.

- **[Command and Response]**

PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0044	
LEN	4	
DATA	2bytes	Begin Template ID No.
	2bytes	END Template ID No.

PREFIX	0XAA55	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0044	
LEN	2	
RET	Result Code	
DATA	0	

Table 3-15 CMD_DEL_CHAR command

E.g.: delete all template in database (ID from 1 to 2000)

Host command: 55 AA 00 00 44 00 04 00 01 00 D0 07 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 1F 02

Target response: AA 55 01 00 44 00 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 46 01

2.5.8 Get first valid ID for enroll in appointed range (CMD_GET_EMPTY_ID)

- [Function]

Get the first ID No. which is usable (no any template data) in the appointed range .

- [Sequence]

1. if the appointed range is invalid , return ERR_INVALID_PARAM
2. else search the first ID No. of valid in the appointed range ,
if the ID is exist return the ID number, else return ERR_EMPTY_ID_NOEXIST.

- [Command and Response]

PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0045	
LEN	4	
DATA	2bytes	Begin Template ID No.
	2bytes	END Template ID No.
PREFIX	0xAA55	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0045	
LEN	success : 4, failed : 2	
RET	Result Code	
DATA	2bytes	success: ID number

Table 3-16 CMD_GET_EMPTY_ID command

E.g. : get the first ID number in the range of 1-2000 (0x0001-0x07D0) , the ID is 11 (0x000B)

Host command : 55 AA 00 00 45 00 04 00 01 00 D0 07 00 00 00 00 00 00 00 00 00 00 20 02

Target response : AA 55 01 00 45 00 04 00 00 00 0B 00 00 00 00 00 00 00 00 00 00 00 54 01

2.5.9 Get an appointed ID No. and check whether enrolled (CMD_GET_STATUS)

- [Function]

Appointed an ID No. and check whether the ID is enrolled fingerprint template data

- [Sequence]

1. If the appointed ID number is invalid, return ERR_INVALID_TMPL_NO .
2. Else return enroll state (1: enrolled template , 0: without template)

- [Command and Response]

PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0046	
LEN	2	
DATA	Template ID	
PREFIX	0xAA55	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0046	
LEN	success : 3, failed : 2	
RET	Result Code	
DATA	1byte	success: enroll state (1: enrolled template, 0: without template)

Table 3-17 CMD_GET_STATUS command

E.g. 1: The ID number (ID=1) is usable (no any template data) to enroll

Host command : 55 AA 00 00 46 00 02 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 48 01
 Target response : AA 55 01 00 46 00 03 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 49 01

E.g. : The ID number (ID=1) have been enrolled

Host command : 55 AA 00 00 46 00 02 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 48 01
 Target response : AA 55 01 00 46 00 03 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 4A 01

2.5.10 Get the total count of have been enrolled (CMD_GET_ENROLL_COUNT)

- [Function]

Get the total count of enrolled in appointed range (begin Template ID ~ end Template ID).

- [Sequence]

1. If the appointed range is invalid, return ERR_INVALID_PARAM
2. Else return the total count of enrolled template .

- [Command and Response]

PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0048	
LEN	4	
DATA	2bytes	begin Template ID No.
	2bytes	END Template ID No.
PREFIX	0xAA55	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0048	
LEN	success : 4, failed : 2	
RET	Result Code	
DATA	2bytes	Total Count of Enrolled Template

Table 3-19 CMD_GET_ENROLL_COUNT command

e.g. : get the count of enrolled, ID range is 1-2000 (0x0001~0x07D0), total count is 10 (0x000A)

Host command : 55 AA 00 00 48 00 04 00 01 00 D0 07 00 00 00 00 00 00 00 00 00 00 00 23 02

Target response : AA 55 01 00 48 00 04 00 00 00 0A 00 00 00 00 00 00 00 00 00 00 00 00 56 01

2.5.11 Creat template data from ImageBuffer to RamBuffer (CMD_GENERATE)

- [Function]

Creat template data from image in ImageBuffer and save the template data to appointed Ram Buffer temporary.

- [Sequence]

1. If the appointed Ram Buffer ID is invalid , return ERR_INVALID_BUFFER_ID .
2. Else check the image in ImageBuffer, if wrong, return ERR_BAD_QUALITY .
3. Ele save the template data to appointed Ram Buffer and return ERR_SUCCESS.

- [Command and Response]

PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0060	
LEN	2	
DATA	2bytes	Ram Buffer ID

PREFIX	0xAA55	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0060	
LEN	2	
RET	Result Code	
DATA	0	

Table 3-20 CMD_GENERATE command

E.g. 1 : save the template data from ImageBuffer to RamBuffer0

Host command : 55 AA 00 00 60 00 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 61 01
Target response : AA 55 01 00 60 00 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 62 01

E.g. 2 : create the template data from ImageBuffer and save to RamBuffer1

Host command : 55 AA 00 00 60 00 02 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 62 01
Target response : AA 55 01 00 60 00 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 62 01

E.g. 3 : create the template data from ImageBuffer and save to RamBuffer2

Host command : 55 AA 00 00 60 00 02 00 02 00 00 00 00 00 00 00 00 00 00 00 00 00 63 01
Target response : AA 55 01 00 60 00 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 62 01

2.5.12 Merge 2 or 3 templates in RamBuffer to one template (CMD_MERGE)

- **[Function]**

Merge the templates in Ram Buffer to one Template, templates can be 2 or 3.

If 2 : Merge RamBuffer0 and RamBuffer1 .

If 3 : Merge RamBuffer0 and RamBuffer1 and RamBuffer2 .

3. [Sequence]

1. If the appointed Ram Buffer ID is invalid, return ERR_INVALID_BUFFER_ID.
2. Else if merge counter is invalid , return ERR_GEN_COUNT .
3. Else merge the template data, if merge failed return error code.
4. Else storage the new Template to appointed Ram Buffer and return ERR_SUCCESS

4. [Command and Response]

PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0061	
LEN	3	
DATA	2bytes	Ram Buffer ID to storage
	1byte	merge count (2 or 3)
PREFIX	0xAA55	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0061	
LEN	2	
RET	Result Code	
DATA	0	

Table 3-21 CMD_MERGE command

E.g. : Merge 3 templates in RamBuffer0

Host command : 55 AA 00 00 61 00 03 00 00 00 03 00 66 01

Target response : AA 55 01 00 61 00 02 00 63 01

2.5.13 Verify 2 Templates in appointed RamBuffer (CMD_MATCH)

- **[Function]**

Verify two fingerprint templates in appointed Ram Buffer

- **[Sequence]**

1. If the appointed two Ram Buffer ID is invalid, return ERR_INVALID_BUFFER_ID
2. Else if verify the two template in Ram Buffer and return verify result.
3. If verify success return ERR_SUCCESS , else return ERR_VERIFY

- **[Command and Response]**

PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0062	
LEN	4	
DATA	2bytes	The first Ram Buffer ID for match
	2bytes	The second Ram Buffer ID for match
PREFIX	0xAA55	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0062	
LEN	2	
RET	Result Code	
DATA	No data	

Table 3-22 CMD_MATCH command

E.g. : Verify the two templates in RamBuffer0 and RamBuffer1

Host command : 55 AA 00 00 62 00 04 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 66 01

Target response : AA 55 01 00 62 00 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 64 01

2.5.14 Identify (1:N) an RamBffer with an range in DB of device (CMD_SEARCH)

- [Function]

Identify (1:N match) the appointed RamBuffer with an appointed range in DB of device and return identify result .

- [Sequence]

1. If the appointed Ram Buffer ID is invalid, return ERR_INVALID_BUFFER_ID.
2. else if the appointed range is invalid, return ERR_INVALID_BUFFER_ID.
3. If there is no enrolled template in the range , return ERR_ALL_TMPL_EMPTY.
4. Identify and return result
5. If search success , return ERR_SUCCESS and DATA field is the searched ID and intelligence update result , else return ERR_IDENTIFY.

- [Command and Response]

PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0063	
LEN	6	
DATA	2bytes	Ram Buffer
	2bytes	Begin Template ID for search
	2bytes	End template ID for search
PREFIX	0xAA55	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0063	
LEN	success : 5, failed : 2	
RET	Result Code	
DATA	3bytes	Success: Template ID(2bytes) + AutoLearn result (1byte)

Table 3-23 CMD_SEARCH command

E.g. : Identify RamBuffer0 with ID range of 1-2000 templates and return result

Host command : 55 AA 00 00 63 00 06 00 00 00 01 00 D0 07 00 00 00 00 00 00 00 00 00 00 40 02

Target response : AA 55 01 00 63 00 05 00 00 00 08 00 01 00 00 00 00 00 00 00 00 00 00 00 71 01

2.5.15 Verify an RamBuffer with an appointed ID in DB (CMD_VERIFY)

- **[Function]**

Verify (1:1 match) fingerprint template in an appointed RamBuffer with an appointed ID in DB and return verify result.

- **[Sequence]**

1. If the appointed template ID is invalid , return ERR_INVALID_TMPL_NO.
2. If the appointed Ram Buffer ID is invalid , return ERR_INVALID_BUFFER_ID .
3. If there is no template data in the appointed ID , return ERR_TMPL_EMPTY.
4. Match the appointed template in Ram Buffer to the template that appointed ID in DB and return result.
5. If verify success , return ERR_SUCCESS , the DATA Field is template ID and AutoLearn result intelligence. Else return ERR_VERIFY

- **[Command and Response]**

PREFIX	0x55AA	
SID	Source Device ID	
DID	Destination Device ID	
CMD	0x0064	
LEN	4	
DATA	2bytes	The Template ID for match
	2bytes	Ram Buffer ID
PREFIX	0xAA55	
SID	Source Device ID	
DID	Destination Device ID	
RCM	0x0064	
LEN	success : 5, failed : 2	
RET	Result Code	
DATA	3bytes	Success: Template ID(2bytes) + AutoLearn (1 : Yes , 0 : NO)

Table 3-24 CMD_VERIFY command

E.g. : Verify (1:1 match) RamBuffer0 with template in DB (ID=8)

Host command : 55 AA 00 00 64 00 04 00 08 00 00 00 00 00 00 00 00 00 00 00 6F 01
 Target response : AA 55 01 00 64 00 05 00 00 00 08 00 01 00 00 00 00 00 00 00 00 00 72 01

2.5.16 Communication error (Incorrect Command)

- [Function]

For communication error, interference cause, module get wrong command, response to HOST.

- [Command and Response]

PREFIX	0x55AA
SID	Source Device ID
DID	Destination Device ID
RCM	0x00FF
LEN	2
RET	ERR_SUCCESS
DATA	-

Table 3-25 Incorrect Command command

2.6 Note

CMD_GENERATE command is produce Template Data in ImageBuffer .

So before call this command, please use CMD_GET_IMAGE first to save the fingerprint image to ImageBuffer.

2. Call command CMD_VERIFY , CMD_SEARCH, CMD_GENERATE, CMD_MERGE, CMD_MATCH will delete the fingerprint image saved in ImageBuffer .
3. Template saved in Ram Buffer2 will be deleted when call
CMD_SEARCH, CMD_VERIFY, CMD_GENERATE, CMD_STORE_CHAR, CMD_DEL_CHAR, CMD_GET_EMPTY_ID, CMD_GET_STATUS, GET_BROKEN_ID, CMD_GETN_ENROLL_COUNT .

So ,do not use Ram Buffer2 except enroll.

3 Appendix

3.1 Response and error code

No	Response and error code	value	Instructions
1	ERR_SUCCESS	0x00	Command dispose success
2	ERR_FAIL	0x01	Command dispose failed
3	ERR_VERIFY	0x10	1:1 match failed with appointed ID Template .
4	ERR_IDENTIFY	0x11	1:N match, but no the same Template .
5	ERR_TMPL_EMPTY	0x12	No enrolled Template in appointed ID.
6	ERR_TMPL_NOT_EMPTY	0x13	With template data in appointed ID.
7	ERR_ALL_TMPL_EMPTY	0x14	NO any enrolled template data
8	ERR_EMPTY_ID_NOEXIST	0x15	No valid for enroll Template ID .
9	ERR_INVALID_TMPL_DATA	0x17	Appointed template Data invalid.
10	ERR_DUPLICATION_ID	0x18	This fingerprint have been enrolled.
11	ERR_BAD_QUALITY	0x19	Fingerprint image lower quality .
12	ERR_MERGE_FAIL	0x1A	Template merge failed.
13	ERR_INVALID_TMPL_NO	0x1D	Template ID No. that appointed is invalid.
14	ERR_INVALID_PARAM	0x22	Use wrong parameters .
15	ERR_GEN_COUNT	0x25	Count of template merge is invalid.
16	ERR_INVALID_BUFFER_ID	0x26	Buffer ID No. is wrong
17	ERR_FP_NOT_DETECTED	0x28	No fingerprint in sensor .

3.2 Enroll and match flow chart

3.2.1 Enroll

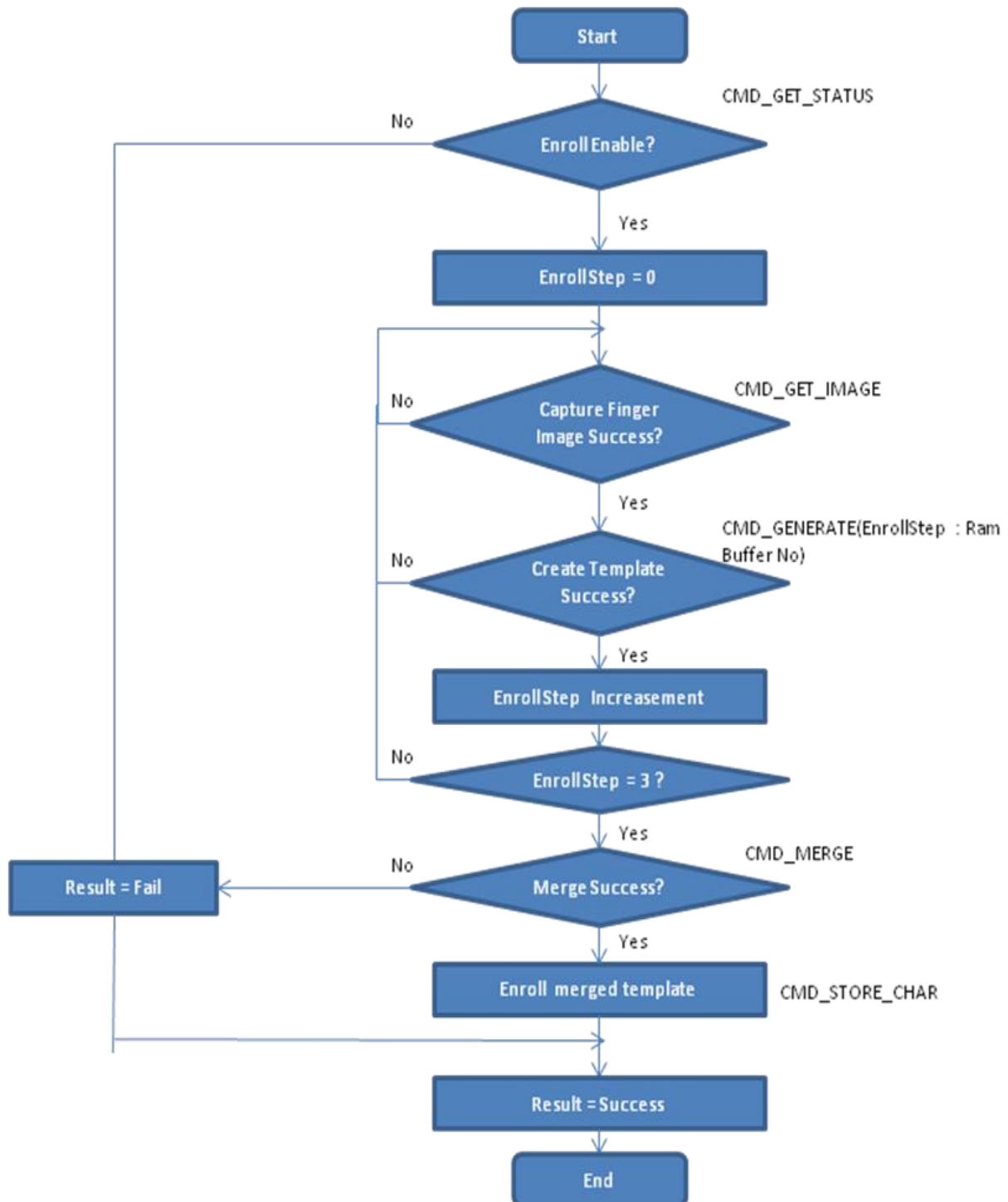


Table 6-1 enroll

3.2.2 Verify & Identify

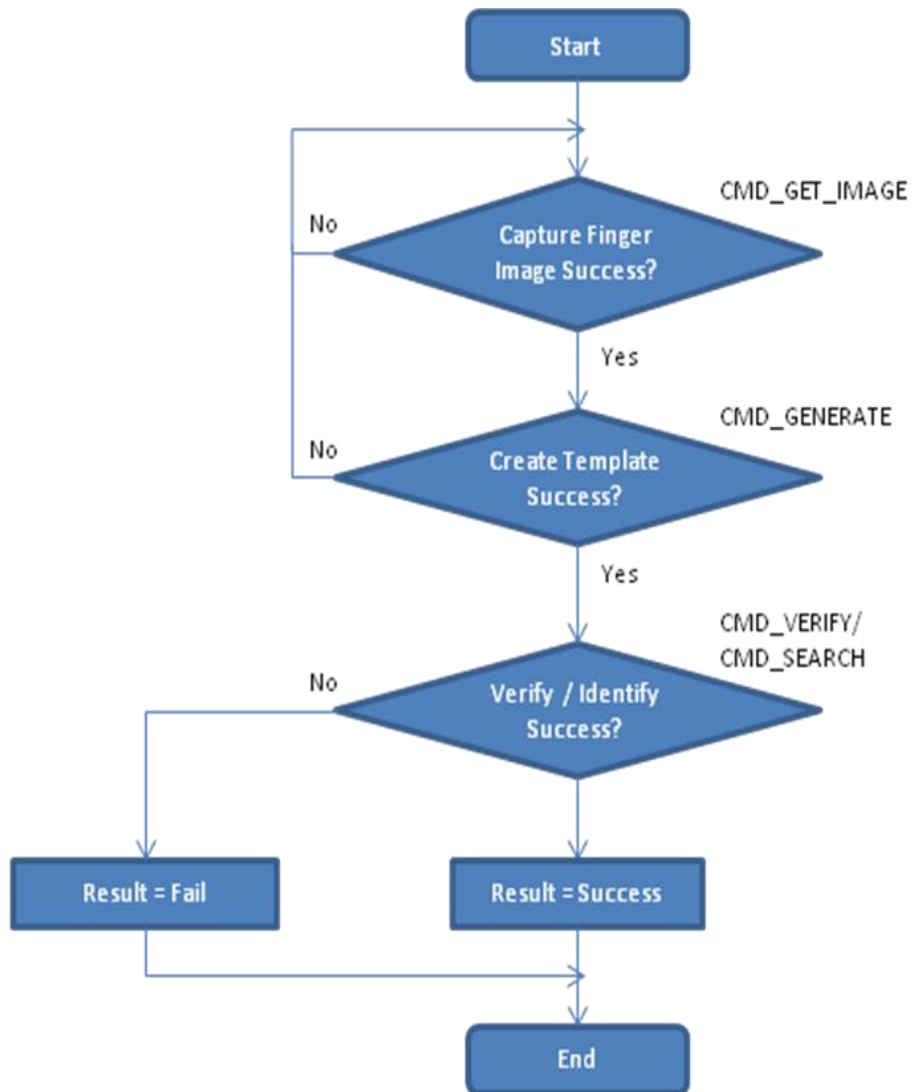
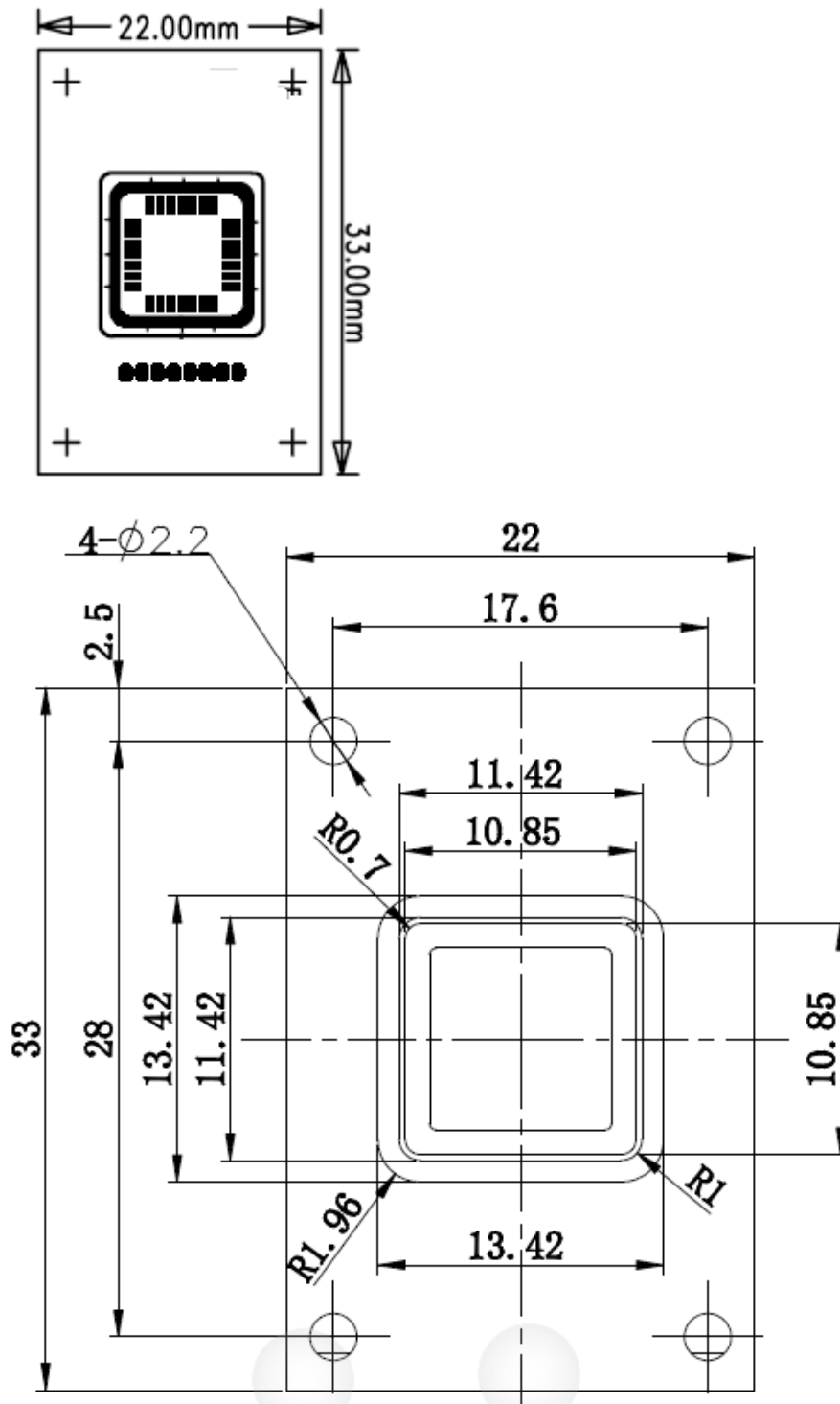


Table 6-2 verify and identify

4 Hardware

5.1 Dimension of board



4.2 Connector

The connector is 8 Pin ,pitch is 1.0mm

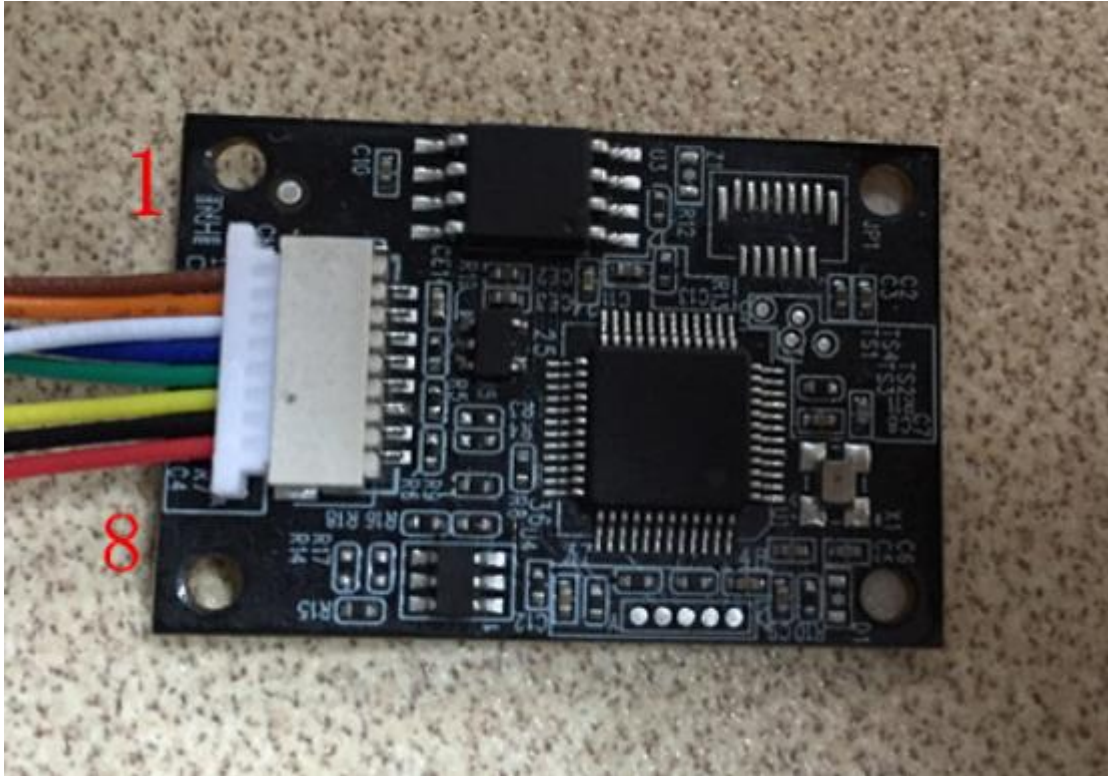


Table 5-1:Type and Communication mode

Type	Communication Port
MEA-335	UART
MEA-336	UART,USB
IMEA-536	UART,USB

Table5-2: Connector signal for MEA-335communication

Pin	Name	description
1	GPIO	Dev
2	GND	DC3.3V
3	VIN	
4	Touch out	Touch switch
5	NC	
6	RX	Receive In
7	TX	Transmit out

8	GND	
---	-----	--

Table5-2: Connector signal for MEA-336 communication

Pin	Name	description
1	3.3V	DC3.3V
2	GND	
3	VIN	HOST Provide for the Touch switch DC2. 3V-5.5V
4	Touch out	Touch switch
5	LED_Blue	
6	TX/D-	Transmit out
7	RX/D+	Receive In
8	LED_Red	

Table 5-3:Connector signal for MEA536 communication

Pin	Name	description
1	3.3V	DC3.3V
2	GND	
3	VIN	HOST Provide for the Touch switch DC2. 3V-5.5V
4	Touch out	Touch switch
5	LED_Blue	
6	RX/D-	Receive In
7	TX/D+	Transmit out
8	LED_Red	

4.3 TECHNOLOGY PARAMETERS

Item	parameters
CPU	Core: 32-bit ARM Cortex-M4, 108MHz, RAM:96KB,Flash memory:1MB
Sensor	BRK301,BRK501
Fingerprint capacity	Option:1000/1700/2000/3000 default is 1700
FAR	< 0.001 % (Security Level: 3)
FRR	< 0.1 % (Security Level: 3)
Match mode	1:N identification ,1:1 verify
Fingerprint template size	498 Bytes
Security level	1-5,default is 3
Time of Response	Fingerprint pretreatment < 0.45 s
	1:N match (full 3000 fingerprints) < 1.5s
Communication mode	USB2.0、UART (3.3V-TTL logic)
UART Communication	Parity = NONE,Stop Bit = 1,Baud rate support: 9600/19200/38400/57600/115200/230400/460800/921600 Default is 115200BPS
Power Supply	Voltage: DC3.3V Work current: <20mA Standby current:<=100uA
Working Environmen	Working temperature : -10 ℃- 60 ℃ Working Humidity: 20% - 80%
Support OS	Windows or Android