ADH Technology Co. Ltd. sales@adh-tech.com.tw

Programming Guide

GT-GT-NUCL1633

Version 2.1

Oct, 12, 2023

Revision History

Version	Date	Description	FW
V1.0	Nov 18, 2021	Created	
V1.1	Nov 29, 2021	Modify Chapter 6 example	
V1.2	Apr 29, 2022	Add UART, LED,Touch command. Modify Enrolled/Identify Flow	
V1.3	May 4th, 2022	Modify Get entry ID	
V1.4	Nov 10, 2022	Add Get S/N command	
V1.5	Apr 24, 2023	Add parameter into LED flicker	
V1.6	Aug 23, 2023	Modift FW version offset 15~17 Add parameter into UART command	V2.2.8
V2.0	Sep, 6, 2023	Change UART default baudrate to 9600bps	V2.5.x
V2.1	Oct, 12, 2023	Identify admin IDs range.	V2.5.2

Table of content

1. Introduction

1.1 Fingerprinter

The ADH-Tech GT-NUCL1633 is one chip fingerprint reader designed to integrate products with a UART interface.

The module is built-in a high-quality hybrid sensor with sufficient effective area. The active area allows stable imaging and the ability to cope with mass-market applications needing both security and convenience. The reader within low power consumption microcontroller and the fingerprint algorithm on it.

1.2 Feature

Sensor

- Ultra-thin hybrid sensor
- SPI sensor image output
- Resolution up to 508 DPI
- Works well with dry, wet, or rough fingerprints.
  The sensor has an SPI interface. Ultra high resolution makes fingerprint algorithms easy to develop and realize.

MCU/Algorithm

- 1:1 Verification, 1:N Identification
- Reading & writing fingerprint template(s) from/to the device
- High-accuracy and high-speed fingerprint identification technology
- Convenient & Safe & Facilitation—Just one touch and easy to enroll and recognize.

2. Packet Structure

2.1 Command Packet (Command)

OFFSET	ITEM	TYPE	DESCRIPTION
0	0xF5	BYTE	Command start code1
1	command	BYTE	Command type
2	P1	BYTE	Command parameter 1
3	P2	BYTE	Command parameter 2
4	P3	BYTE	Command parameter 3
5	P4	BYTE	Command parameter 4
6	Check Sum	BYTE	XOR check Sum OFFSET[1]^...^ OFFSET[5] = Check Sum
7	0xF5	BYTE	Command end code 2

2.2 Response Packet (Acknowledge)

OFFSET	ITEM	TYPE	DESCRIPTION
0	0xF5	BYTE	Response start code1
1	Command	BYTE	Often the same as the command type
2	Q1	BYTE	Response parameter 1
3	Q2	BYTE	Response parameter 2
4	Q3 (ACK)	BYTE	Response parameter 3
5	0x00	BYTE	Reserved.
6	Check Sum	BYTE	XOR check Sum OFFSET[1]^...^ OFFSET[5] = Check Sum
7	0xF5	BYTE	Response end code2

2.3 Data Packet (Data)

OFFSET	ITEM	TYPE	DESCRIPTION
0	0xF5	BYTE	Data start code1
1~Len	Data	N BYTES	N bytes Data
Len+1	Check Sum	BYTE	XOR check Sum OFFSET[1]^...^ OFFSET[Len] = Check Sum
Len+2	0xF5	BYTES	Data end code2

3. Command Summary

In a command package, the second byte for the command code may be one of the following:

Number (HEX)	Alias	Description
A0	**Open**	Initialization
A1	**Close**	Termination
A3	**Uart Control**	Set UART baud rate
B4	**LED Control**	LED On/Off/Flicker
B5	**Is Press Finger**	Check if the finger is placed on the sensor
01	**Enroll**	An enrollment function
04	**Delete ID**	Delete the fingerprint with the specified ID
05	**Delete All**	Delete all fingerprints from the database
09	**Get User Count**	Get enrolled fingerprint count
0C	**Identify**	1:N Identification of the captured fingerprint image with the database
0D	**Get Entry ID**	Get an unused ID
26	**Get firmware version**	Get firmware version
27	**Get Serial Number**	Get Module S/N
92	**Enroll Cancel**	Break the enrollment sequences

4. ACK Code

In a response package, the device will return the execution result, and it may be of the following:

ACK Parameter	Value	Description
ACK_SUCCESS	0x00	Command execute success
ACK_FAIL	0x01	Command execute fail
ACK_FULL	0x04	The database is full
ACK_NOUSER	0x05	For GetEnrollCount
ACK_USER_EXIST	0x07	The specified ID is registered
ACK_TIMEOUT	0x08	Timeout during capturing finger
ACK_WRONG_FORMAT	0x09	Template wrong format
ACK_BREAK	0x18	Abort current command
ACK_INVALID_PARAMETER	0xB0	Input invalid parameter
ACK_FINGER_IS_NOT_PRESSED	0xB1	Finger is not pressed
ACK_COMMAND_NO_SUPPORT	0xB4	Command not support
ACK_ENROLL_OVEREXPOSURE	0xB5	Finger image is overexposure
ACK_ENROLL_MOVE_MORE	0xB6	Finger moved too less.
ACK_ENROLL_MOVE_LESS	0xB7	Finger moved too much.
ACK_ENROLL_DUPLICATE	0xB8	Finger position is duplicated.
ACK_FINGER_PRESS_NOT_FULL	0xB9	Finger press is not full.
ACK_ENROLL_POOR_QUALITY	0xBA	Finger image quality is poor.

5. Command Details

5.1 Open

Check the device connection status and initalize sensor status.

Byte	1	2	3	4	5	6	7	8
CMD	0xF5	0xA0	0	0	Flag	0	CHK	0xF5
ACK	0xF5	0xA0	Hi(Len)	Low(Len)	0	0	CHK	0xF5

Flag:
1: Read FW date and module.

Hi(Len):
FW module and date length high byte.

Low(Len):
FW module and date length low byte.

Byte	1	2…Len+1	Len+2	Len+3
DATA	0xF5	Data	CHK	0xF5

Data structure (11 Bytes):

Offset	Description
0	Device ID : 0x01
1	Reserved : default 0x00.
2	FW release date [0] : Day, [1] : Month [2] : Year (Low byte), [3] : Year (High byte)
6	Reserved : default 0x00.
10	Sensor type

If the Flag is set to 0, the data package will not return.

5.2 Close

Close command does nothing. (Reserved)

Byte	1	2	3	4	5	6	7	8
CMD	0xF5	0xA1	0	0	0	0	CHK	0xF5
ACK	0xF5	0xA1	0	0	ACK	0	CHK	0xF5

ACK:
ACK_SUCCESS

5.3 UART Control

UART baud rate setting.

Byte	1	2	3	4	5	6	7	8
CMD	0xF5	0xA3	Index	Timeout	0	0	CHK	0xF5
ACK	0xF5	0xA3	0	0	ACK	0	CHK	0xF5

Baud rate will be change and respond ACK after 100msec.
The timeout duration for UART transmission (unit: milliseconds). It should be set to different values for different Baud rates (e.g. 9600: >0x0C 19200: >0x04, 115200:>0x02). Power off will restore the default baudrate.

Baud rate index:
1: 9600 (FW v2.5.x default setting)
2:19200
3:115200 (FWv2.2.x default setting)
4:230400
5:460800
6:921600

Time out:
0: Auto

ACK:
ACK_SUCCESS

※FW v2.2.x default setting is 115200bps, FW v2.5.x is 9600bps.

5.4 LED Control

Control LED On / Off / Flicker.

Byte	1	2	3	4	5	6	7	8
CMD	0xF5	0xB4	switch	0	0	0	CHK	0xF5
ACK	0xF5	0xB4	0	0	ACK	0	CHK	0xF5

Switch:
0x00: On
0x01: Off
0xFF: Flicker

ACK:
ACK_SUCCESS

5.5 Check finger pressing status(IsPressFinger)

This command is used to check finger. This function can be replaced by reading touch signal.

Byte	1	2	3	4	5	6	7	8
CMD	0xF5	0xB5	0	0	0	0	CHK	0xF5
ACK	0xF5	0xB5	Status	0	ACK	0	CHK	0xF5

Status:
0: Is Not Pressed
1: Is Pressed

ACK:
ACK_SUCCESS
ACK_FINGER_IS_NOT_PRESSED

※ Recommend polling touch IO pin to check finger pressing status, rather than polling the IsPressFinger command. This approach can effectively conserve power and resources.

5.6 Enroll

Before enrollment, you can call “Get Entry ID 0x0D” to get an unregistered ID (1~N) and set it to the USER_ID (2nd~3rd bytes). Start a registration, and you shall repeatedly call the Enroll command till the enroll result (2nd byte) of the response packet has become from 0x01 to 0x03. You shall use the Enroll command (8+1) times this way. The first enroll command is setting the user ID.
If there is no finger pressed on the sensor for each sampling time in 8 seconds, the enrollment will be canceled. The enrollment flow can be interrupted by each command except **Enroll** and **IsPressFinger**.

Byte	1	2	3	4	5	6	7	8
CMD	0xF5	0x01	USER ID_H	USER ID_L	0	Reserved 0x00	CHK	0xF5
ACK	0xF5	0x01	USER ID_H	USER ID_L	ACK	0	CHK	0xF5

USER ID_H :
USER ID high byte.

USER ID_L :
USER ID low byte

Note: User ID value is 1~N

Reserved:
Value Must be 0.

ACK :
ACK_SUCCESS
ACK_FAIL
ACK_USER_EXIST
ACK_INVALID_PARAMETER
ACK_ ENROLL_OVEREXPOSURE
ACK_ENROLL_MOVE_MORE
ACK_ENROLL_MOVE_LESS
ACK_ENROLL_DUPLICATE
ACK_FINGER_PRESS_NOT_FULL
ACK_ENROLL_POOR_QUALITY

Byte	1	2	3	4	5	6	7	8
CMD	0xF5	0x01	0	0	0	0	CHK	0xF5
ACK	0xF5	ENROLL RESULT	ENROLL PROGRESS	Ignore	ACK	Ignore	CHK	0xF5

.

ENROLL RESULT:
0x01: Continue
0x03: Final enroll

ENROLL PROGRESS:
Percentage, 1 - 8.

ACK :
ACK_SUCCESS
ACK_FAIL

Ignore:
Byte 4,6: Ignore

Host shall keep resending the Enroll command until **ENROLL RESULT** is 0x03. The enrollment is completed while 0x03 is received. Ignore "Enroll Progress" byte when ACK_FAIL was received. The following takes three sampling times, for example:

Example: ID=1

The 1st command: F5 01 00 01 00 00 xx F5 (set ID)
Return : F5 01 00 00 00 00 xx F5

The 2nd command: F5 01 00 00 00 00 xx F5 (press finger)
Return : F5 01 01 00 00 00 xx F5

The 3rd command: F5 01 00 00 00 00 xx F5 (press finger)
Return : F5 01 02 00 00 00 xx F5

……..

The last command: F5 01 00 00 00 00 xx F5 (enroll success)
Return : F5 03 08 00 00 00 xx F5

5.7 Delete one fingerprint(DeleteID)

Byte	1	2	3	4	5	6	7	8
CMD	0xF5	0x04	USER ID_H	USER ID_L	0	0	CHK	0xF5
ACK	0xF5	0x04	0	0	ACK	0	CHK	0xF5

USER ID_H :
USER ID high byte.

USER ID_L :
USER ID low byte

Note: User ID value is 1~N

ACK :
ACK_SUCCESS
ACK_NOUSER
ACK_INVALID_PARAMETER

Example: Delete ID=1

Command: F5 04 00 01 00 05 F5

Return : F5 04 00 00 00 04 F5

5.8 Delete all fingerprints(DeleteAll)

Byte	1	2	3	4	5	6	7	8
CMD	0xF5	0x05	0	0	0	0	CHK	0xF5
ACK	0xF5	0x05	0	0	ACK	0	CHK	0xF5

ACK :
ACK_SUCCESS

Example: Delete all ID

Command: F5 05 00 00 00 05 F5

Return : F5 05 00 00 00 05 F5

5.9 Get enrolled fingerprint count(GetEnrollCount)

Byte	1	2	3	4	5	6	7	8
CMD	0xF5	0x09	0	0	0	0	CHK	0xF5
ACK	0xF5	0x09	USERCOUNT_H	USERCOUNT_L	ACK	0	CHK	0xF5

ACK :
ACK_SUCCESS
ACK_NOUSER

Example: Get user count

Command: F5 09 00 00 00 00 09 F5

Return: F5 09 00 02 00 00 0B F5 (Two ID exists)

5.10 1:N Identification(Identify)

This function supports breaks when capturing a finger. This command will recognize and respond to the user ID. If the user can not be found, the ID shall return 0. If no users are registered, it will respond with ACK_NOUSER.
To prevent administrator enrolls more than two IDs, the host can set the identification range to restrict the range of admin IDs.

Byte	1	2	3	4	5	6	7	8
CMD	0xF5	0x0C	Range	0	0	0	CHK	0xF5
ACK	0xF5	0x0C	USER ID_H	USER ID_L	ACK	Ignore	CHK	0xF5

Range (v2.5.2):
0: Default. Identify all IDs.
1~5: Identify admin IDs range for the administrator mode. (e.g. 3: from ID1~ID3)

USER ID_H :
USER ID high byte.

USER ID_L :
USER ID low byte

ACK:
ACK_SUCCESS
ACK_NOUSER

Ignore:
Byte 6: Ignore

Example:
Command: F5 0C 00 00 00 00 0C F5

Identify success: ID=2
Return: F5 0C 00 02 00 00 CHK F5

Identify Not Found: ID=0
Return: F5 0C 00 00 00 00 0C F5

5.11 Get entry ID(GetEntryID)

Return an unused ID (1~N).

Byte	1	2	3	4	5	6	7	8
CMD	0xF5	0x0D	0	0	0	0	CHK	0xF5
ACK	0xF5	0x0D	USER ID_H	USER ID_L	ACK	0	CHK	0xF5

USER ID_H :
USER ID high byte.

USER ID_L :
USER ID low byte

Note: Unused ID value is 1~N

ACK :
ACK_SUCCESS
ACK_FULL

5.12 Get firmware version

Get firmware information

Byte	1	2	3	4	5	6	7	8
CMD	0xF5	0x26	0	0	0	0	CHK	0xF5
ACK	0xF5	0x26	Hi(Len)	Low(Len)	ACK	0	CHK	0xF5

Hi(Len):
Firmware version length high byte

Low(Len):
Firmware version length low byte

ACK:
ACK_SUCCESS

Byte	1	2…Len+1	Len+2	Len+3
DATA	0xF5	Version Data	CHK	0xF5

The version data include sensor type, enroll type, and module version.

Version data structure (40 Bytes) :

Offset	Description
0	Sensor type
12	Year
13	Month
14	Day
15~17	Version
18	TBD
39	TBD

5.13 Get Serial Number

Get unique serisal number of modual.

Byte	1	2	3	4	5	6	7	8
CMD	0xF5	0x27	0	0	0	0	CHK	0xF5
ACK	0xF5	0x27	0	0x10	ACK	0	CHK	0xF5

ACK:
ACK_SUCCESS

Byte	1	2…17	18	19
DATA	0xF5	16 bytes serisal number	CHK	0xF5

5.14 Enroll Cancel

Interrupt the enrollment sequence and erase the template data of ID.

Byte	1	2	3	4	5	6	7	8
CMD	0xF5	0x92	USER ID_H	USER ID_L	0	0	CHK	0xF5
ACK	0xF5	0x92	USER ID_H	USER ID_L	ACK	0	CHK	0xF5

USER ID_H :
USER ID high byte.

USER ID_L :
USER ID low byte

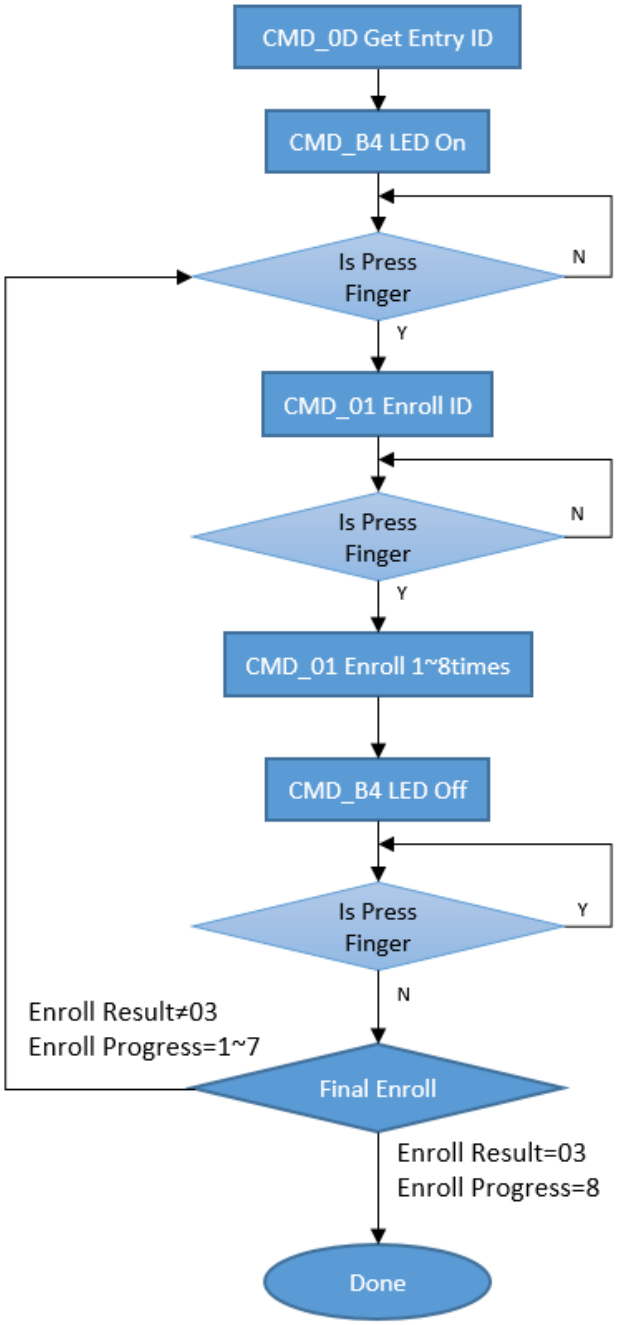
Note: User ID value is 1~N

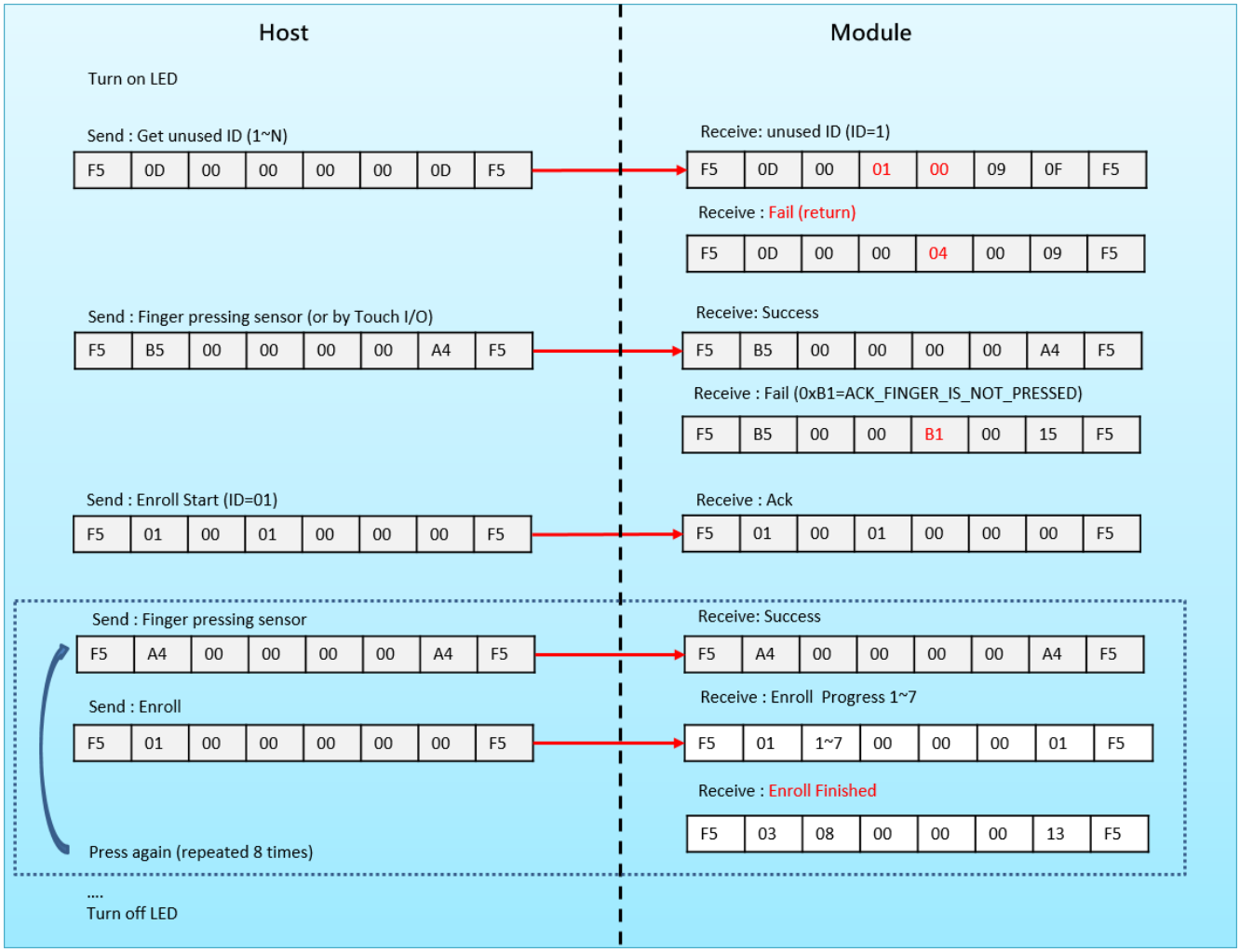
Reserved:
Value Must be 0.

ACK :
ACK_SUCCESS
ACK_FAIL

6. Protocol: Flowchart

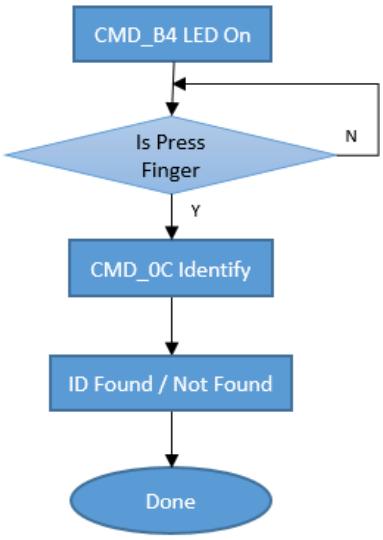
6.1 Enrollment

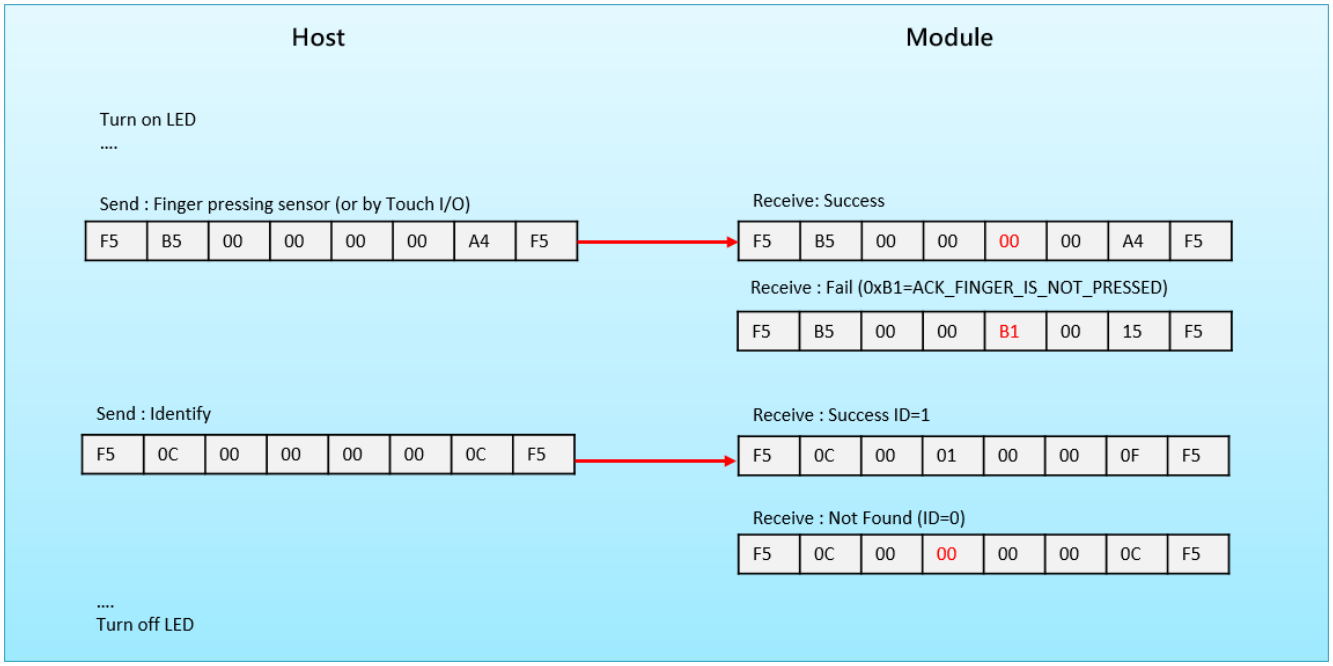




※ Recommend polling touch IO pin (TP_HI) to check finger pressing status, rather than polling the IsPressFinger command. This approach can effectively conserve power and resources.

6.2 Identification





※ Recommend polling touch IO pin (TP_HI) to check finger pressing status, rather than polling the IsPressFinger command. This approach can effectively conserve power and resources.

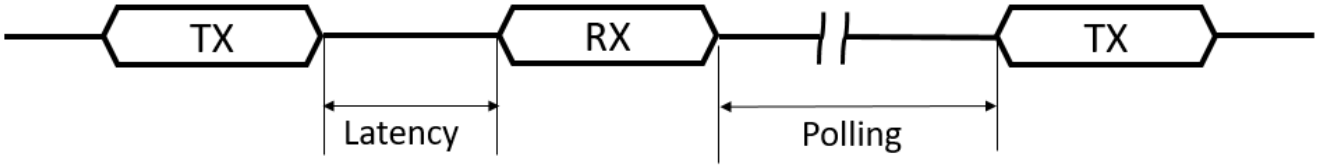
'

7. Appendix:

7.1 Uart Rx/Tx Timing

The overall duration of UART transmission comprises four stages: Transmit (Tx) time, latency, Receive (Rx) time and Polling time.

※FW v2.2.x default setting is 115200bps, FW v2.5.x default setting is 9600bps.



1. Tx: The host sends a command to the GT-NUCL1633 module. UART time out setting must be longer than transmit time (Tx).
2. Latency: This represents the delay between sending command data and receiving respond from GT-NUCL1633 module.
3. Rx: The host receives a response from the GT-NUCL1633 module.
4. Polling: Controlled by the host.

Example: 0xB4 LED Control

Baud rates	1. Tx	2. Latency	3. Rx	Total (1+2+3)
115200	689usec	1.59msec	655usec	**2.986msec**
19200	3.89msec	1.41msec	3.89msec	**9.19msec**
9600	7.79msec	2.94msec	7.79msec	**18.52msec**